

CSE 3101: Internet Computing II

Lab Session 7

Paul Crawford

Semester 1, Week 12 (12th & 13th November, 2018)

1. Aims

- 1.1. Further understanding of Web applications, including client-side Web Forms (with server-side processing upon submission), server-side PHP functions, and Web Service architectures.
- 1.2. Further increased facility with the ongoing sets of concepts & techniques needed for eventual completion of the Final Project.
- 1.3. **POLICY CHANGE (REMINDER)**: Please note that henceforth **all Lab Sessions will be marked** and count as a portion of the overall Lab grade (10%). Completed labwork must be zipped into an archive (filename given later below at the end of § 2.1), and submitted to the Tutor (<pcrawford@mac.com>) by midnight Friday each week, with the Subject line: ‘CSE3101 - Lab <N> - <YourGroupName>’.

2. Tasks

2.1. Web Forms, and Service API ‘Layer’: Uploading Files; Etc.

- 2.1.1. Please try to ensure that your xAMP[P] stack includes a modern version of PHP at least \geq v5.4.0 (e.g., in XAMPP, by navigating to ‘localhost/dashboard’, and then clicking ‘PHPInfo’ in the top bar). This will help to avoid any “missing-feature” errors such as those related to the relatively recent ‘`http_response_code()`’ function. {It might end up being necessary for the Tutor or Lecturer to make formal upgrade requests to the Technical staff (?).}
- 2.1.2. Download the Lab 07 Support folder archive from the tutor’s Nyanza Software website (<www.nyanzasoftware.com>) > ‘Teaching’, § ‘University of Guyana’ > ‘CSE3101: Internet Computing II’. Then, **unzip (extract)** it to obtain the ‘Lab 07 Support’ folder.
- 2.1.3. Set up the Lab 07 test database via the PhpMyAdmin console, by first creating an empty database named ‘lab07_db’, and then **importing** the ‘lab07_db.sql’ file (located in the folder extracted above).
- 2.1.4. Copy the ‘form_with_upload’ subfolder (located in the folder extracted above) into your Apache Web Root directory (e.g., ‘C:\xampp\htdocs’).
- 2.1.5. Copy the ‘client_files’ subfolder (located in the folder extracted above) into any convenient User location on the hard drive. {Note: These files were intended to test the

file-upload feature, but this may be postponed to the next Lab session.}

- 2.1.6. Exercise the Web Forms & Service API functionality (this time, all URIs should be tested directly via any Web browser, **not** a specialised web service such as Postman), and save the various **result files** as follows:

- 2.1.6.1. Navigate to ‘localhost/form_with_upload/**form_maint/staff_maint.php**’ to create the **first** test Staff Member, by submitting to that same URI the following POST data:

First Name : Test
Last Name : Tester
Age : 25

Make sure that you see the result message (at the bottom of the form): ‘Staff Member was created.’

- 2.1.6.2. Navigate to ‘localhost/form_with_upload/**page/staff.php**’ {to retrieve all the existing Staff Members, indirectly via an internal JSON API layer}, and save the resulting HTML page (there should be just 1 entry) to a file named ‘Staff_ReadViaAPI_AfterU_First.html’.

- 2.1.6.3. Click the ‘Information’ link in the corresponding [HTML page](#) table entry {to retrieve details for that first test Staff Member, indirectly via an internal JSON API layer}, and save the resulting HTML details page to a file named ‘Staff_ReadViaAPI_AfterU_First_Details.html’.

- 2.1.6.4. (Re-)Navigate to ‘localhost/form_with_upload/**form_maint/staff_maint.php**’ to create the **second** test Staff Member, by submitting to that same URI the following POST data:

First Name : Test2
Last Name : Tester2
Age : 34

Make sure that you see the result message (at the bottom of the form): ‘Staff Member was created.’

- 2.1.6.5. (Re-)Navigate to ‘localhost/form_with_upload/**page/staff.php**’ {to retrieve all the existing Staff Members, indirectly via an internal JSON API layer}, and save the resulting HTML page (there should now be 2 entries) to a file named ‘Staff_ReadViaAPI_AfterU_Second.html’.

- 2.1.6.6. Click the ‘Information’ link in the corresponding [HTML page](#) table entry {to retrieve details for that **second** test Staff Member (note that they will actually be the **top row** again, due to the internal descending sort by creation date/time), indirectly via an internal JSON API layer}, and save the resulting HTML details

page to a file named 'Staff_ReadViaAPI_AfterU_Second_Details.html'.

- 2.1.6.7. From the above **4 result files** — i.e.: 'Staff_ReadViaAPI_AfterU_First.html', 'Staff_ReadViaAPI_AfterU_First_Details.html', 'Staff_ReadViaAPI_AfterU_Second.html', and 'Staff_ReadViaAPI_AfterU_Second_Details.html' — create a zip archive named 'CSE3101_Lab07_<YourGroupName>.zip' and submit it as described in § 1.3.

2.2. Overall Considerations

- 2.2.1. For additional information, practice and code samples, you could also explore any of various useful articles, tutorials & references available online. For instance, the **Wikipedia** site (<en.wikipedia.org>) has several detailed articles on Web Services, Web APIs, the REST architecture, etc.