

SOEN 345 - Software Testing, Verification, and
Quality Assurance
Assignment 7

Ali Hanni - 40157164

*Gina Cody School of Computer Science and Software Engineering
Concordia University, Montreal, QC, Canada*

Winter 2022

*** NOTE ***



Ali Hanni

alihanni95@gmail.com

[View Profile](#)

Account Settings

Notification Preferences

Active Sessions

Sign Out



Add a new account

Here are the required screenshot showing that the different parts of the assignment were correctly executed.

The screenshot shows the Postman interface for a REST client. The left sidebar lists the collection 'SOEN345-partE' with endpoints: GET getlistusers, GET getsingleuser, GET getsingleusernotfound, and POST postuser. The main panel displays the 'GET' request for 'https://reqres.in/api/users?page=2'. The 'Tests' tab is active, showing the following JavaScript code:

```
6 });
7 pm.test("Total page number equal to 2", function () {
8   var jsonData = pm.response.json();
9   pm.expect(jsonData.total_pages).to.eql(2);
10 });
11 pm.test("Total records number equal to 12", function () {
12   var jsonData = pm.response.json();
13   pm.expect(jsonData.total).to.eql(12);
14 });
```

The 'Test Results' tab shows four passed tests:

- PASS Status code is 200
- PASS Response time is less than 1000ms
- PASS Total page number equal to 2
- PASS Total records number equal to 12

Additional details at the bottom: Status: 200 OK, Time: 29 ms, Size: 1.88 KB, Save Response.

Part E - Get List Users

The screenshot shows the Postman interface for a REST client. The left sidebar lists the collection 'SOEN345-partE' with endpoints: GET getlistusers, GET getsingleuser, GET getsingleusernotfound, and POST postuser. The main panel displays the 'GET' request for 'https://reqres.in/api/users/2'. The 'Tests' tab is active, showing the following JavaScript code:

```
2 pm.response.to.have.status(200);
3 });
4 pm.test("Response time is less than 1000ms", function () {
5   pm.expect(pm.response.responseTime).to.be.below(1000);
6 });
7 pm.test("Verify email", function () {
8   var jsonData = pm.response.json();
9   pm.expect(jsonData.data.email).to.eql("janet.weaver@reqres.in");
10 });
```

The 'Test Results' tab shows three passed tests:

- PASS Status code is 200
- PASS Response time is less than 1000ms
- PASS Verify email

Additional details at the bottom: Status: 200 OK, Time: 51 ms, Size: 116 KB, Save Response.

Part E - Get Single Users

SOEN345-partE / **getsingleusernotfound**

GET <https://reqres.in/api/users/23> Send

Params Authorization Headers (7) Body Pre-request Script **Tests** Settings Cookies

```

1 pm.test("Status code is 404", function () {
2   pm.response.to.have.status(404);
3 });
4 pm.test("Response time is less than 1000ms", function () {
5   pm.expect(pm.response.responseTime).to.be.below(1000);
6 });

```

Test scripts are written in JavaScript, and are run after the response is received. [Learn more about tests scripts](#)

SNIPPETS
[Get an environment variable](#)
[Get a global variable](#)
[Get a variable](#)

Body Cookies Headers (18) **Test Results (2/2)** Status: 404 Not Found Time: 41 ms Size: 875 B Save Response

All Passed Skipped Failed

PASS Status code is 404

PASS Response time is less than 1000ms

Part E - Get Single User Not Found

SOEN345-partE / **postuser**

POST <https://reqres.in/api/users> Send

Params Authorization Headers (9) Body Pre-request Script **Tests** Settings Cookies

```

1 pm.test("Status code is 201", function () {
2   pm.response.to.have.status(201);
3 });
4 pm.test("Response time is less than 1000ms", function () {
5   pm.expect(pm.response.responseTime).to.be.below(1000);
6 });

```

Test scripts are written in JavaScript, and are run after the response is received. [Learn more about tests scripts](#)

SNIPPETS
[Get an environment variable](#)
[Get a global variable](#)
[Get a variable](#)

Body Cookies Headers (15) **Test Results (2/2)** Status: 201 Created Time: 249 ms Size: 867 B Save Response

All Passed Skipped Failed

PASS Status code is 201

PASS Response time is less than 1000ms

Part E - Post User

SOEN345-partE No Environment, a min ago View Summary Run Again New

All Tests Passed (110) Failed (0)

Iteration 1

Request	URL	Status	Time	Size
GET getlistusers	https://reqres.in/api/users?page=2 / getlistusers	200 OK	44 ms	1.932 KB
Pass Status code is 200 Pass Response time is less than 1000ms Pass Total page number equal to 2 Pass Total records number equal to 12				
GET getsingleuser	https://reqres.in/api/users/2 / getsingleuser	200 OK	36 ms	1.18 KB
Pass Status code is 200 Pass Response time is less than 1000ms Pass Verify email				
GET getsingleusernotfound	https://reqres.in/api/users/23 / getsingleusernotfound	404 Not Found	276 ms	873 B

Part E - Collection was run (10 iterations, 1000ms delay)

Question 1

Using Postman makes it easy to write code first thanks to snippets that let you write pre-written or even you own tests seamlessly. You can also customize your tests and, for example, run multiple iterations to simply test the fiability of the results. You can easily export your tests and collections as JSON files and use them in your applications.

Question 2

- Postman: Is used to create and run tests on API. Postman also lets you build your own APIs.
- Selenium: Provides playback tools mainly aimed for front-end testing.
- JMeter: Is used for load and performance testing. It is specially aimed for web applications.
- Blazemeter: Another performance testing software.

Question 3

Postman is mainly used for API testing. API testing would fall under the category of component testing. Indeed, testing an API is a kind of black box testing where a system is isolated and solely the output is considered.