Testing APIs withPostman

Please complete this task within given timeframe in the email. No worries if you cannot complete all parts, as we just want to see your current knowledge and ability to learn. All mandatory questions have a weight in percent (Postman Overview: 7%; Basic Postman: 32%; Manual Testing: 41%; Intermediate Postman: 20%), please allocate the time accordingly.

Refer to the Swagger file sent to you in the email (COCUS Challenge - Swagger) for all details about the API you will be testing (Import the Swagger file in https://editor.swagger.io - do NOT use the default “Petstore”.

* Required fields

1. Have you used Postman before? *

Mark only one oval.

☒ Yes

○ Tried it

○ No

2. Have you used Swagger before? *

Mark only one oval.

☒ Yes

○ No

Understanding Swagger

A quick section to ensure understanding of Swagger.

3. Get endpoint parameters for Login DE *

Using the provided swagger, please tell us all the parameters for Login DE endpoint, including if they are required, type of parameter e.g., header and allowed value(s)

Header parameters: device-os (required)(allowed values: android/ios), app-locale (required)(allowed values: de_DE/en_GB), x-api-key (required), correlation-id (not required)

Body parameters: loginBody (required) - json with two keys -> username and password

Postman overview

A quick overview of Postman components.
In this section you need to write a short description for each of the component
* What is it/How it works
* How is it useful

4. Workspace (1%) *

5. Collection (1%) *

6. Environment (1%) *

7. Sandbox and Test Scripts (1%) *

8. Collection Runner (1%) *

9. Monitor (1%) *

10. Newman (1%) *

Basic Postman

In this section we will cover the basics of API testing though Postman. Each question is a follow up to a previous one, so it is recommended to complete them in the order provided. However, if you struggle, you may skip the step and complete them through other methods.

Before starting, please ensure you have Postman installed (Download URL: https://www.getpostman.com/apps).

12. Create necessary Environments (10%)*

Please create and upload environments based on the following: Kurt Fritze is a DE customer and logs in to their app using their username (test) and password (test), their booking reference is ADE9452. Sam Smith is a UK customer using an android device and their booking reference is AR58930. Once complete, upload the environments. Please name the environments with the following format: Your_Name _Market (e. John_Doe_UK).

Files submitted: Helder_Moreira_UK.json and Helder_Moreira_DE.json

13. Create a Collection (20%) *

Using the provided swagger (attached in the email), create a single collection with all available endpoints. Once complete and you can call each endpoint, upload the collection. If you have made changes to environments, or created globals, please upload them here as well. (Hint: Login endpoints return "login-token" header on success request). Please name your collections with Your_Name.

Files submitted: Helder_Moreira.json

14. Did you use the import functionality to create the collection?*

If not, give it a try :)

Mark only one oval.

☒ Yes

○ No

15. Reading response results in postman (2%) *

Please provide screenshots of status, time taken, size, headers, and body of a SINGLE RESPONSE in Postman

Files submitted: response.png and response_headers.png

Manual Testing

Ability to test and analyze is very important in this role, please ensure to provide detailed answers to questions in this section.

16. Analyze and improve testing requirements for the following scenario (6%) *

A new API has been created to standardize Login endpoint for all markets. The request is now POST only with b parameters: bookingRef, surname, username, and password. A new header parameter "appName" has also been added to ensure correct processing of the data. Tester tries to login with each market and closes the testing tic What other tests could the tester do to ensure that API is working correctly?

Test invalid parameter values to ensure the API returns error; Test not filling optional parameters, and expect the API to succeed;

Test not filling required parameters and expect the API to return error;

17. Test the APIs (35%) *

Test all the APIs using Exploratory Testing / Session Based test techniques. This is one of the more important questions in the Test so spend plenty of time on it! Document all your findings including your test ideas, unresolved questions, positive/negative results, headers, payloads, values etc. Upload all your documentation here. Your ability to organize the findings in a clear and structured way will also be judged.

Files submitted: All files inside folder COCUS_Assignment_QA_Automation\Postman\17.TestAPI\

Intermediate Postman

Everything from "Basic Postman" section applies here, however we will be looking at the Postman Sandbox, which allows to automate testing.

18. Validate headers for all requests (20%) *

All requests in this API have to return Content-Type (application/json) and Connection (keep-alive) headers. If you think some other header validations need to be done for some endpoints, please add it. Once complete, upload collection.

Files submitted:

19. Do you want to try the advanced section? *

Mark only one oval.

○ Yes *Skip to question 20*

○ No *Skip to question 22*

Advanced Postman (Optional)

This a bonus section, do not feel pressured to complete i

20. Ensure that only necessary endpoints are called for each environment in collection runner

As you may have noticed, there are different endpoints that each market needs to call and by default all the requests will be sent in order when using Automation Runner, Monitor or Newman. Ensure that endpoints that a not used by the market are skipped during those runs. Once complete, upload the collection.

Files submitted:

21. Run the collection in Newman

Using Newman, run the collection and upload the screenshot of the summary.

Files submitted:

Feedback

Please share with us your thoughts about this task and how we could improve it for the future.

22. Please write your feedback here:

____

____

____

____

____