# LumiraDX Technical Test – Test completion report

## Executive summary:

This document highlights any findings, defects found and recommendations for the LumiraDX technical test.

## Defects:

A total of 13 defects were found during testing. The breakdown is listed as:

|  |  |  |  |
| --- | --- | --- | --- |
| High | Medium | Low | Cosmetic |
| 3 | 6 | 3 | 1 |

### High priority defects

|  |  |  |  |
| --- | --- | --- | --- |
| Issue ID | Description | Recommendation | Link |
| 1 | Swagger API doc does not open on https:port 8889 | After some investigation into the code, it seems port 8889 was not defined: <https://github.com/amaccormack-lumira/rest_api_demo/blob/master/rest_api_demo/settings.py#L6>  Perhaps a new port needs to be defined. | <https://github.com/iamandrewli/lumiradx_tech_test/issues/1> |
| 2 | New category name missing first letter | The cause appears to be due to the following line: <https://github.com/amaccormack-lumira/rest_api_demo/blob/master/rest_api_demo/api/blog/business.py#L32>  This slices the string to read from position 1 until the end of the string. | <https://github.com/iamandrewli/lumiradx_tech_test/issues/2> |
| 7 | Posting with existing ID causes 500 | Exceptions could be coded to handle the error more gracefully. | <https://github.com/iamandrewli/lumiradx_tech_test/issues/7> |

### Medium priority defects

|  |  |  |  |
| --- | --- | --- | --- |
| Issue ID | Description | Recommendation | Link |
| 4 | Using value greater than max int causes 500 | Exceptions could be coded to handle the error more gracefully. | <https://github.com/iamandrewli/lumiradx_tech_test/issues/4> |
| 6 | Missing error responses | Code in the missing error responses always add into the swagger documentation. | <https://github.com/iamandrewli/lumiradx_tech_test/issues/6> |
| 8 | null message received after creation of category | Add a more useful message to help the user. | <https://github.com/iamandrewli/lumiradx_tech_test/issues/8> |
| 10 | Blank name value allowed | Validations to be coded to guard against this. | <https://github.com/iamandrewli/lumiradx_tech_test/issues/10> |
| 11 | Blank name value allowed | Blank name value allowed | <https://github.com/iamandrewli/lumiradx_tech_test/issues/11> |
| 13 | Duplicate category names allowed | Validations to be coded to guard against this. | https://github.com/iamandrewli/lumiradx\_tech\_test/issues/13 |

### Low and cosmetic priority defects

|  |  |  |  |
| --- | --- | --- | --- |
| Issue ID | Description | Recommendation | Link |
| 3 | description missing | Update swagger api documentation. | <https://github.com/iamandrewli/lumiradx_tech_test/issues/3> |
| 5 | 404 - getting standard html page rather than custom message | Code the application in a way to handle this error. | <https://github.com/iamandrewli/lumiradx_tech_test/issues/5> |
| 9 | Missing description | Update swagger api documentation. | <https://github.com/iamandrewli/lumiradx_tech_test/issues/9> |
| 12 | Swagger API doc versions are different | Keep the versioning consistent. | <https://github.com/iamandrewli/lumiradx_tech_test/issues/12> |

### Manual test results

|  |  |  |  |
| --- | --- | --- | --- |
| Test ID | Test Scenario | Result | Defect |
| 1 | GET: All categories | PASS |  |
| 2 | GET: Category ID with associated blog posts | PASS |  |
| 3 | GET: Category ID with no associated blog posts | PASS |  |
| 4 | GET: Category ID with non-existing ID | PASS |  |
| 5 | GET: Category ID with the maximum int value | PASS |  |
| 6 | GET: Category ID over the maximum int value | FAIL | https://github.com/iamandrewli/lumiradx\_tech\_test/issues/4 |
| 6 | GET: Category ID with string value as ID | PASS with caveat | https://github.com/iamandrewli/lumiradx\_tech\_test/issues/5 |
| 7 | GET: Category ID with an ID value of 0 | PASS |  |
| 8 | GET: Category ID with an ID value of -1 | PASS with caveat | https://github.com/iamandrewli/lumiradx\_tech\_test/issues/5 |
| 9 | GET: Category ID | PASS |  |
| 10 | POST: With no values | PASS |  |
| 11 | POST: With no ID and with a category name in body | FAIL | https://github.com/iamandrewli/lumiradx\_tech\_test/issues/2 |
| 12 | POST: With new ID and with category name in body | FAIL | https://github.com/iamandrewli/lumiradx\_tech\_test/issues/2  https://github.com/iamandrewli/lumiradx\_tech\_test/issues/8 |
| 13 | POST: With existing ID and new category name in body | FAIL | https://github.com/iamandrewli/lumiradx\_tech\_test/issues/7 |
| 14 | POST: With new ID and with an existing category name in body | FAIL | https://github.com/iamandrewli/lumiradx\_tech\_test/issues/2 |
| 15 | POST: With string value in ID field and int value in category name | PASS |  |
| 16 | POST: With ID field populated only | PASS |  |
| 17 | POST: With existing category name only | FAIL | https://github.com/iamandrewli/lumiradx\_tech\_test/issues/13 |
| 18 | POST: With blank value in the name | FAIL | https://github.com/iamandrewli/lumiradx\_tech\_test/issues/10 |
| 19 | PUT: Update existing category name | PASS |  |
| 20 | PUT: Update non existing category | PASS |  |
| 21 | PUT: Update with blank string on name field | PASS |  |
| 22 | PUT: Update category where blog posts exists | PASS |  |
| 23 | DELETE: Existing category | PASS |  |
| 24 | DELETE: Attempt to delete category that doesn’t exist | PASS |  |
| 25 | DELETE: string | PASS |  |
| 26 | DELETE: use wildcard to attempt to delete all | PASS |  |
| 27 | DELETE: Blog post ID | PASS |  |

## Automated test results

|  |  |  |  |
| --- | --- | --- | --- |
| TEST ID | Test Scenario | Result | Defect |
| 28 | * Create new blog post and link with existing category. * GET category can retrieve newly created blog post. | PASS |  |
| 29 | * DELETE newly created blog post. * GET category and check that newly created blog post was deleted. | PASS |  |
| 30 | * Create new blog post and link with existing category * GET category to retrieve blog post * PUT to update the blog post category * GET category to check that blog post no longer exist in original category and now moved to new category | PASS |  |

## Future considerations:

The POST categories endpoint is recommended to go back into development to work on the defects found to ensure a more robust system.

Should QA have more time; it would be beneficial to increase the automation integration test suite to ensure a good regression test suite.

Future improvement to the automated integration test suite to set up and tear down tests so that the test suite can be executed multiple times.