# LumiraDX Technical Test – High Level Test Plan

## Executive summary:

Our team has been tasked with building a simple back-end for a blog. This is built using RESTful API to implement CRUD operations.

## In scope:

The following are in scope for testing:

* Swagger API documentation.
* GET /blog/categories/ - Returns list of blog categories
* POST /blog/categories/ - Creates a new blog category
* DELETE /blog/categories/{id} - Deletes blog category
* GET /blog/categories/{id} - Returns a category with a list of posts
* PUT /blog/categories/{id} - Updates a blog category

## Out of scope:

The following end points are out of scope for testing, however, will be aid the testing of the above in scope end points:

* GET /blog/posts/ - Returns list of blog posts
* POST /blog/posts/ - Creates a new blog post
* GET /blog/posts/archive/{year}/ - Returns list of blog posts from a specified year
* GET /blog/posts/archive/{year}/{month}/ - Returns list of blog posts from a specified month
* GET /blog/posts/archive/{year}/{month}/{day}/ - Returns list of blog posts from a specified day
* DELETE /blog/posts/{id} - Deletes blog post
* GET /blog/posts/{id} - Returns a blog post
* PUT /blog/posts/{id} - Updates a blog post

## Defect definitions:

Defects found in the project will be raised in the issues part of GitHub.

Defects reports will have the following sections:

* **Title** – Descriptive title of bug found.
* **Description** – Summary of bug found.
* **Priority** – This will be rated: High, Medium, Low, and Cosmetic.
* **Steps to reproduce** – Steps to help reproduce the bug.
* **Screenshots** (If applicable)

## Risks and assumptions:

* Due to working commitments on other projects, the QAs time on this is limited.
* There is a risk due to the time limitations; the QA will not be able to complete the automated tests.
* This project assumes that all the required data in the database has been pre-populated for testing.
* This project assumes that testing carried out on MacOS will be identical to Windows and Linux systems.

## Manual tests approach:

QA will make use of Postman to test the end points manually. Postman will allow the QA to organise his testing and group the endpoints together in a more readable and understandable manner.

## Automated test approach:

QA will make use of the Pytest framework using Pycharm to create some automated test scenarios.

As most validation will be completed for the blog categories end point, the automated tests aims to create an end-to-end flow.