System Requirements Specification Index

For

Django Rest-API
Blog Application

Version 1.0

IIHT Pvt. Ltd.
fullstack@iiht.com

TABLE OF CONTENTS

REST API FOR BLOG APPLICATION
System Requirements Specification

1 Project Abstract

Blog Application is Django Rest API application with SQLite Database, where it allows to manage the blogs and can post comments on the blog.

Following is the requirement specifications:

	Blog Application
Modules	
1	Blogs
2	Comments
Blog Module Functionalities	
1	Create a Blog
2	Update a Blog
3	Delete a Blog
4	Get the Blog by Id
Comment Module Functionalities	
1	Post a comment on the Blog

2 Assumptions, Dependencies, Risks / Constraints

2.1 Blog Constraints:

- While fetching the Blog by Id, if Id does not exist then operation should throw custom exception.
- While Updating the Blog by Id, if Id does not exist then operation should throw custom exception.
- While deleting the Blog by Id, if Id does not exist then operation should throw custom exception.

2.2 Comment Constraints

- If you want to post a comment on a blog, if blog id does not exist, then operation should throw a custom exception.

3 Business Validations

- Blog title max 100 characters.
- Blog content is max 200 characters.
- Comment blog id is not null.
- Comment is max 200 characters.

4 REST ENDPOINTS

Rest End-points to be exposed in the controller along with method details for the same to be created.

Class Name	Method Name	Purpose Of Method
CommentsView	post(self, request,format=None)	To post a comment on the Blog.
BlogsView	get(self,request,pk=None,format=None)	To get the blog by Id.
	post(self, request,format=None)	To add a blog.
	patch(self,request,pk,format=None)	To updateablog.
	delete(self,request,pk,format=None)	To delete ablog.

5 TEMPLATE CODE STRUCTURE

Resources (Models)

Class	Description	Status
BlogsModel	o A model class for Blogs. o It will map to the BlogsModel table.	Already implemented.
CommentsModel	o A model class for Comments. o It will map to the CommentsModel table.	Already implemented.

Resources (Serializers)

Class	Description	Status
BlogsSerializer	A serializer for BlogsModel	Already implemented
CommentsSerializer	A serializer for CommentsModel	Already implemented

Resources (Views)

Class	Description	Status
BlogsView	A class for the get, post, patch, delete functionalities on **BlogsModel**model**.**	To be implemented
CommentsView	A class for the post functionality on **CommentsModel**model**.**	To be implemented

Resources (Exceptions)

Class	Description	Status
IdNotAvailable	Custom exception **IdNotAvailable** is raised in case specified blog id is not available.	Already implemented.

6 Execution Steps to Follow

1. All actions like build, compile, running application, running test cases will be through Command Terminal.
2. To open the command terminal the test takers, need to go to
   Application menu (Three horizontal lines at left top) -> Terminal ->New Terminal.
3. The editor Auto Saves the code.
4. If you want to exit(logout) and to continue the coding later anytime (using Save & Exit option on Assessment Landing Page) then you need to use CTRL+Shift+B command compulsorily on code IDE. This will push or save the updated contents in the internal git/repository. Else the code will not be available in the next login.
5. These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.
6. To test any Restful application, the last option on the left panel of IDE, you can find ThunderClient, which is the lightweight equivalent of POSTMAN.
7. To test any UI based application the second last option on the left panel of IDE, you can find Browser Preview, where you can launch the application.
8. Install 'djangorestframework' module before running the code. For this use the following command.

   pip install djangorestframework
9. Use the following command to run the server

   python3 manage.py runserver
10. Mandatory: Before final submission run the following commands to execute testcases

    python3 manage.py test blog.test.test_functional

    python3 manage.py test blog.test.test_exceptional

    python3 manage.py test blog.test.test_boundary

11. To test rest end points

    Click on 'Thunder Client' or use Ctrl+Shift+R ->Click on 'New Request' (at left side of IDE)

12. Once you are done with development and ready with submission, you may navigate to the previous tab and submit the workspace. It is mandatory to click on "Submit Assessment" after you are done with code.

13. You need to use CTRL+Shift+B - command compulsorily on code IDE, before final submission as well. This will push or save the updated contents in the internal git/repository, and will be used to evaluate the code quality.

-------*--------