Nurovant AI ML Assessment

You will be building a Q.A.T (Question, Answer & Test) system for research documents. The Q.A.T system is very similar to the typical Q&A (Question & Answer) system, the difference here lies in the response provided by the Q.A.T system.

The workflow for a normal Q&A system entails asking a question w.r.t a specific document and a response is generated as an answer to the question. For a Q.A.T system a *"test_question"* is returned along with the answer and some other information (if necessary), the *"test_question"* is used to evaluate whether the user understood the response generated by the system.

For this assessment, you are required to build out a flask application with 3 major endpoints:

1. The **"upload/"** endpoint: This endpoint would be used to upload the research document to the application.

2. The **"query/"** endpoint: This endpoint would be used as an interactive Q.A.T session between the user and the system.

3. The **"evaluate/"** endpoint: This endpoint would be used to evaluate the response provided by the user to the "test_question".

The expected response format for the following endpoints are indicated below, respectively:

1. **"upload/"** : Feel free to return the response based on your own discretion.

2. **"query/":** The json response should be as follows:

```
{
 "answer": " …",
 "bullet_points": [ …, …, …, …]
 "test_question": "...",
 "test_question_id:"..."
 }
```

3. **"evaluate/"** :The json response should be as follows:

```
{
 "knowledge_understood": bool,
 "knowledge_confidence": int
 }
```

Response definition for **“query/”**:

1. *answer*: The answer provided by tthe system to the question asked.
2. *bullet_points*: A list of bullet points emphasizing key details in the answer to improve understanding.
3. *test_question*:Generated question to evaluate if the user understood the answer provided.
4. *test_question_id*: id for every test_question generated.

Response definition for **“evaluate/”**:

1. *knowledge_understood*: This is a boolean value indicating that the user understood the answer provided. **True** if the user understood the answer, **False** if the user did not understand the answer.
2. *knowledge_confidence*: This is an integer value (in %) indicating how confident the evaluation is.

Bonus Point:
For every **“test_question”** generated, a **“test_answer”** should be generated as well and stored in a database. This test_answer would be used to evaluate the user answer for the provided **“test_question”.**

Note:
It is not mandatory to deploy this application, just ensure the code is properly documented and all the files needed to run the application are provided in order to test the application locally. Also attach a demo video explaining and running the code base.

Deadline:
The solution to this assessment should be returned on the 3rd of October, 12:00 pm. A github repo should be provided with the necessary files and documentation detailing how to set up and run the application.

Feel free to reach out if you have any questions.