End-to-End Testing in NLP Systems

1. Introduction to End-to-End Testing
End-to-End (E2E) testing is a testing methodology that validates the complete flow of an application from start to finish. In NLP systems, it ensures that text input provided by users passes through preprocessing, model inference, API handling, and storage layers correctly.

2. Importance of E2E Testing in NLP
NLP systems involve multiple components such as tokenizers, ML models, APIs, and databases. E2E testing verifies that these components work together seamlessly and helps detect real-world issues early.

3. NLP System Architecture
A typical NLP system includes:
- User Interface
- Backend API (Flask/FastAPI)
- NLP Model (BERT, GPT, Ollama, etc.)
- Database (MongoDB)
- Response Layer

4. NLP E2E Testing Workflow
User Input → API Request → Preprocessing → Model Inference → Post-processing → Database → Response

5. Input Validation Testing
- Empty text
- Special characters
- Long inputs
- Multilingual text

6. Model Inference Validation
- Model loading
- Prediction format
- Confidence scores
- Error handling

7. API-Level Testing
- HTTP status codes
- Response time
- JSON schema validation

8. Database Validation
- Record insertion
- Timestamp validation
- Model metadata storage

9. E2E Testing Tools for NLP
- Pytest
- Postman
- Selenium (UI)
- Docker
- CI/CD pipelines

What Do We Test in NLP E2E?

1. **Input Handling**
    - Empty input
    - Long text
    - Special characters
    - Multiline text
2. **NLP Processing**
    - Tokenization works
    - Model loads correctly
    - Prediction format is correct
3. **API Behaviours**
    - HTTP status codes
    - Response time
    - JSON structure
4. **Output Validation**
    - Correct label (Bug / Feature / Neutral)
    - Confidence score exists
    - Generated text is meaningful