Comprehensive Test Strategy for PDF Upload and Metadata Extraction

Purpose

This document outlines a thorough approach to testing a newly developed feature that allows users to upload PDF documents and extract associated metadata, such as file names and upload dates. The strategy ensures functionality, performance, and reliability while validating compliance with user expectations and requirements.

Scope of Testing

The outlined test strategy covers:

- **Functional Testing**: Verifying core capabilities like file uploads, metadata extraction, and error handling.
- **Non-Functional Testing**: Assessing performance, file size limitations, and system behavior under varying conditions.
- **Automation Testing**: Defining automated test scenarios for repetitive and critical paths.

Functional Test Cases

Positive Scenarios

1. **Upload a Valid PDF Document**

   - **Preconditions**: User must be authenticated.
   - **Steps**:
     1. Navigate to the upload interface.
     2. Select a valid PDF file.
     3. Click the "Upload" button.
   - **Expected Outcome**: The file uploads successfully, and metadata, including file name and upload date, is displayed correctly.

2. **Verify Metadata Accuracy**

   - **Steps**:
     1. Upload a correctly formatted PDF document.
   - **Expected Outcome**: Displayed metadata, such as the file name, upload date, and file size, matches the actual document details.

3. **Support for Bulk Uploads**

- **Steps**:
  1. Upload multiple valid PDF documents at once.
- **Expected Outcome**: Each document uploads without error, and corresponding metadata is displayed for all files.

Negative Scenarios

1. **Unsupported File Format**

   - **Steps**:
     1. Select a non-PDF file format (e.g., .jpg, .docx).
     2. Attempt to upload the file.
   - **Expected Outcome**: An error message appears, such as "Unsupported file type. Please upload a PDF."

2. **Empty File Upload**

   - **Steps**:
     1. Select an empty PDF file.
     2. Click the "Upload" button.
   - **Expected Outcome**: A message appears, such as "File is empty. Please select a valid document."

3. **No File Selected**

   - **Steps**:
     1. Click "Upload" without selecting a file.
   - **Expected Outcome**: The system prompts the user with a message like "No file selected. Please choose a file to upload."

4. **Corrupted File Upload**

   - **Steps**:
     1. Attempt to upload a corrupted PDF document.
   - **Expected Outcome**: The system displays an error indicating that the file could not be processed.

Non-Functional Test Scenarios

File Size Constraints

1. **Max File Size**

   - **Scenario**: Upload a PDF document at the maximum allowable size (e.g., 10 MB).
   - **Expected Outcome**: The file uploads successfully within acceptable performance metrics.

2. **Exceed File Size**

- **Scenario**: Attempt to upload a file exceeding the maximum size limit.
- **Expected Outcome**: An error message, such as "File exceeds maximum size limit," is displayed.

Performance and Load Testing

1. **Concurrent File Uploads**

   - **Scenario**: Simulate multiple users uploading files simultaneously.
   - **Expected Outcome**: The system processes all uploads without performance degradation.
2. **Response Time Validation**

   - **Scenario**: Measure the time taken to upload and process a document.
   - **Expected Outcome**: File uploads and metadata extraction complete within a predefined time frame (e.g., under 3 seconds).
3. **Stress Testing**

   - **Scenario**: Push the system by attempting to upload a very large number of files concurrently.
   - **Expected Outcome**: The system remains stable and processes uploads gracefully.

Security Testing

1. **File Type Validation**

   - **Scenario**: Upload files with disguised extensions (e.g., .exe renamed to .pdf).
   - **Expected Outcome**: The system detects and rejects files with invalid content.
2. **Metadata Injection**

   - **Scenario**: Attempt to upload malicious files with embedded harmful metadata.
   - **Expected Outcome**: The system detects and blocks such uploads.

Automation Test Coverage

Automatable Scenarios

1. **Core Functionalities**:

   - Uploading valid PDF files.
   - Metadata extraction and validation.
2. **Negative Scenarios**:

   - Unsupported file types.

- Empty and corrupted file uploads.
3. **Regression Testing**:

  - Validating upload functionality after system updates.

Prioritization

- **Critical**: File upload validation, metadata extraction, and error handling.
- **Moderate**: Load testing and edge cases like maximum file size.
- **Low**: Stress testing for extreme conditions.

Tools and Frameworks

- **Functional Testing**: Cypress or Playwright.
- **Performance Testing**: JMeter or Locust.

Practical Scenarios and Testing Value

Real-World Relevance

- Users upload PDFs with diverse file sizes and content regularly. The system must handle all valid scenarios while rejecting invalid ones effectively.

Added Value

- Ensures users experience seamless uploads and accurate metadata display.
- Identifies potential vulnerabilities and performance bottlenecks early.
- Provides a robust framework for continuous testing and quality assurance.

Conclusion

This test strategy delivers a comprehensive framework for validating the PDF upload and metadata extraction feature. By incorporating functional, non-functional, and automated testing, it ensures a reliable, secure, and user-friendly experience. This approach emphasizes quality and scalability, setting a strong foundation for future enhancement.