System Testing

Introduction

System testing is a crucial phase in the software development lifecycle (SDLC) that evaluates the entire application as a cohesive unit. It verifies if the system meets its functional and non-functional requirements, ensuring it integrates seamlessly with any external dependencies and functions as intended in a real-world environment.

Purpose of Testing

The primary goals of system testing in this document processing application are:

To identify issues that might impact the overall functionality of the application.

To ensure the application extracts specified fields from uploaded documents accurately and reliably.

To verify system integration with any external components like file storage services or document processing APIs.

To evaluate system performance, including factors like processing speed and memory usage.

To assess system usability and user experience for a seamless document upload and information retrieval process.

Types of Testing

Here's a breakdown of various types of system testing applicable to this application, along with a sample test case table for each:

1. Unit Testing:

Unit testing focuses on isolating and testing individual software units (functions, modules) to confirm they operate as designed. While not directly applicable to the entire application in this context, unit testing might be relevant for any custom functionalities developed within the project.

2. Integration Testing:

Integration testing verifies how different modules or components within the application interact and exchange data. This might involve testing the integration of the document upload functionality with the document processing engine or the information retrieval component.

3. System Testing:

System testing evaluates the entire application as a system, ensuring it fulfills all its intended functionalities and behaves as expected in a simulated real-world environment.

4. User Testing:

User testing involves real users interacting with the application to evaluate its usability, user interface (UI), and overall user experience. This helps identify any usability issues or areas for improvement in the user interaction flow.