

Erica M. Ravanera

BSI/T 3E

Test Procedure Documentation

Objective:

The objective of the performance test is to evaluate the system's ability to retrieve, organize, and present backend course information according to predefined criteria. This includes assessing the efficiency and accuracy of data retrieval, sorting, and extraction processes.

Test Steps:

a. Database Tool Installation:

Obtain the MongoDB Database tool from the provided link, ensuring compatibility with the system's architecture. Execute the installation process following the step-by-step instructions provided in the MongoDB documentation. Pay close attention to system requirements and any prerequisites to ensure a smooth installation experience.

b. Backend API Endpoint Setup:

Establish a backend API endpoint to handle the course data using the MongoDB Database tool's import functionality. Utilize the provided command to import the course data into the designated MongoDB database and collection.

c. Test Scenarios:

Conduct various test scenarios to validate the functionality of the backend API endpoint:

- Retrieve all backend courses and arrange them alphabetically to ensure proper sorting.
- Extract the name and specialization of each course, verifying the accuracy of data extraction.
- Retrieve all BSIS and BSIT courses from the curriculum, confirming the correct retrieval of specified courses.

d. Data Validation:

Implement rigorous data validation procedures at each step of the testing process:

- Verify the integrity of data by cross-referencing with the original dataset and ensuring consistency in retrieved information.

- Conduct thorough checks to validate the alphabetical ordering of courses and the correctness of extracted course names and specializations.
- Perform targeted validations to confirm the accurate retrieval of BSIS and BSIT courses, ensuring compliance with specified criteria.

Challenges Faced:

a. MongoDB Installation Issues:

Encounter challenges during MongoDB installation, such as compatibility issues with the system environment or dependencies. Troubleshoot and resolve these issues by referring to the MongoDB documentation and seeking assistance from relevant resources or support channels.

b. Data Validation Complexity:

Experience complexity in validating data integrity, particularly when handling special characters or formatting discrepancies in course names. Address this challenge by refining validation algorithms, enhancing error-handling mechanisms, and conducting thorough testing to cover edge cases.

Solutions Implemented:

a. Documentation Review:

Conduct a comprehensive review of the MongoDB installation guide to identify and address potential installation issues effectively. Ensure adherence to all specified steps and verify system compatibility to mitigate installation challenges.

b. Improved Validation Approach:

Enhance the data validation approach by implementing advanced validation techniques, including regular expression matching, character encoding handling, and robust error detection mechanisms. Continuously refine and optimize validation procedures to improve accuracy and reliability.