Code analysis report

Business Requirements

Document Title: Request for COBOL Code Submission for Analysis

Introduction

In order to conduct a comprehensive and precise analysis of the COBOL program, it is imperative to have access to the actual source code. This document outlines the necessity for stakeholders to submit the COBOL program code that requires analysis. The aim is to ensure a thorough understanding and evaluation of the code to meet the business needs effectively.

Purpose

The primary purpose of this request is to facilitate an accurate and detailed analysis of the COBOL program. By reviewing the actual code, we can provide insights and recommendations that are tailored to the specific requirements and functionalities of the program in question.

Requirement

Stakeholders are requested to provide the COBOL program code they wish to have analyzed. The submission of the complete and unaltered source code is essential for the following reasons:

1. **Accuracy**: Access to the original code ensures that the analysis is based on the actual implementation rather than assumptions or approximations.

2. **Comprehensiveness**: A detailed examination of the code allows for a thorough understanding of its structure, logic, and functionality, enabling a more comprehensive assessment.

3. **Tailored Recommendations**: With the specific code in hand, any insights or recommendations can be directly applicable to the program, enhancing the relevance and effectiveness of the analysis.

Conclusion

For an accurate and effective analysis of the COBOL program, it is crucial for stakeholders to provide the actual source code. This will enable a detailed evaluation and facilitate the

development of tailored solutions that align with the business objectives. We appreciate your cooperation in this matter and look forward to receiving the necessary materials for analysis.

Generated Test Cases

Certainly! Here are three detailed test scenarios based on the given business requirements:

Test Case 1

1. **Test Case Description**
   Validate that the system prompts the user to input COBOL code when no code is provided.

2. **Input Values**
   - No COBOL code is provided (empty input).

3. **Expected Output**
   - The system displays a message: "Please provide the COBOL program code for analysis."

4. **Reasoning**
   The requirement specifies that an accurate analysis requires the actual COBOL code. If no code is provided, the system should prompt the user to input the necessary data to proceed with the analysis.

Test Case 2

1. **Test Case Description**
   Verify that the system accepts and processes valid COBOL code input.

2. **Input Values**
   - A valid COBOL program code snippet:
   ```
   IDENTIFICATION DIVISION.
   PROGRAM-ID. HelloWorld.
   PROCEDURE DIVISION.
   DISPLAY 'Hello, World!'.
   STOP RUN.
   ```

3. **Expected Output**
   - The system accepts the input and proceeds with the analysis without any error

messages.

4. **Reasoning**
   The requirement implies that the system should be able to analyze the given COBOL code. Providing a valid COBOL program should allow the system to proceed with the analysis, demonstrating its capability to handle correct inputs.

Test Case 3

1. **Test Case Description**
   Check the system's response to invalid COBOL code input.

2. **Input Values**
   - An invalid COBOL program code snippet:
     ```
     IDENTIFICATION DIVISION.
     PROGRAM-ID. MissingDivision.
     DISPLAY 'This will fail'.
     STOP RUN.
     ```

3. **Expected Output**
   - The system displays an error message: "Invalid COBOL code. Please check the syntax and structure."

4. **Reasoning**
   The requirement indicates the need for accurate analysis, which is only possible with valid COBOL code. When invalid code is provided, the system should identify the issue and inform the user, ensuring that only correct and analyzable code is processed.