Interface Design Document (IDD)

1. Introduction and Background

1.1 Purpose

The purpose of this Interface Design Document (IDD) is to define and detail the interface between [System A] and [System B]. This document outlines the design, requirements, and specifications necessary to ensure successful integration and data exchange between these systems.

1.2 Scope

This document covers the design and implementation of the interface, including HL7 message specifications, data mapping, architecture, and testing. It serves as a guideline for developers, testers, and administrators involved in the integration process.

1.3 Audience

This document is intended for:

- System architects

- Software developers

- Quality assurance engineers

- IT support teams

- Project managers

1.4 Definitions

- HL7 : Health Level Seven, a set of international standards for the transfer of clinical and administrative data between software applications.

- Interface : A defined method of communication between two systems.

2. System Overview

2.1 System A

- Description : Briefly describe System A, including its purpose, key functionalities, and data it manages.

- Technologies : List technologies used (e.g., database, programming languages).

2.2 System B

- Description : Briefly describe System B, including its purpose, key functionalities, and data it manages.

- Technologies : List technologies used.

2.3 Integration Objectives

- Describe the main goals of the integration, such as data synchronization, process automation, or reporting enhancements.

3. Requirements

3.1 Functional Requirements

- Requirement 1 : Detail functional requirement, e.g., "System A must send patient data to System B on creation or update."

- Requirement 2 : Detail functional requirement, e.g., "System B must acknowledge receipt of the data with an acknowledgment message."

3.2 Non-Functional Requirements

- Performance : Response time and throughput requirements.

- Security : Data encryption, user authentication, and authorization.

- Compliance : Adherence to relevant standards and regulations (e.g., HIPAA).

3.3 Constraints

- Technical Constraints : Limitations due to technology or architecture.

- Operational Constraints : Constraints related to operations, such as batch processing times.

4. HL7 Message Specifications

4.1 Message Types

- Message Type 1 : E.g., ADT (Admission, Discharge, Transfer) - Description of the message type and its purpose.

- Message Type 2 : E.g., ORM (Order Message) - Description of the message type and its purpose.

4.2 Message Structure

- Segments : List of segments used (e.g., MSH, PID, OBR).

- Fields : Description of key fields in each segment.

4.3 Message Format

- Encoding : Character encoding (e.g., UTF-8).

- Delimiter : Field delimiter, component separator, and repetition separator.

5. Data Mapping Table

| Source Field | Source Segment | Target Field | Target Segment | Transformation/Rules |

|---------------------|-----------------|---------------------|----------------|-------------------------------|

| PatientID | PID | PatientID | PID | Direct mapping |

| PatientName | PID | PatientName | PID | Concatenate First & Last Name |

| DateOfBirth | PID | DOB | PID | Format YYYY-MM-DD |

| OrderNumber | ORC | OrderNumber | ORC | Extract and map directly |

6. Interface Architecture

6.1 Overview

Provide a high-level architecture diagram showing the interface components, data flow, and interaction points between System A and System B.

6.2 Components

- Data Sender : Component responsible for sending data from System A.

- Data Receiver : Component responsible for receiving data in System B.

- Message Broker : If applicable, a component that handles message routing and transformation.

- Database : Components where data is stored temporarily or permanently.

6.3 Communication Protocols

- Protocol : Define protocols used (e.g., HTTP, MLLP).

- Ports and Endpoints : List the ports and endpoints for data exchange.

7. Message Flow and Error Handling

7.1 Message Flow

- Step 1 : Describe the first step in the message flow, e.g., "System A generates an HL7 message upon patient registration."

- Step 2 : Describe subsequent steps, including any transformations or routing.

7.2 Error Handling

- Error Detection : Methods for detecting errors (e.g., validation checks, status codes).

- Error Reporting : How errors are reported and logged.

- Error Resolution : Steps for resolving common errors, such as invalid data formats or communication failures.

8. Testing and Validation Plan

8.1 Test Strategy

- Unit Testing : Test individual components of the interface.

- Integration Testing : Test the interaction between System A and System B.

- End-to-End Testing : Test the complete data flow from System A to System B.

8.2 Test Cases

- Test Case 1 : Description, expected result, and actual result.

- Test Case 2 : Description, expected result, and actual result.

8.3 Validation

- Data Validation : Ensure data integrity and correctness.

- Compliance Validation : Verify adherence to HL7 standards and other relevant regulations.

9. Deployment and Maintenance Instructions

9.1 Deployment Plan

- Pre-Deployment : Checklist of tasks before deployment (e.g., backup, configuration).

- Deployment Steps : Detailed steps for deploying the interface.

9.2 Maintenance

- Routine Maintenance : Regular tasks to keep the interface running smoothly.

- Troubleshooting : Common issues and their resolutions.

- Updates : Procedure for updating the interface.

10. Training Materials

10.1 User Guide

- Overview : Introduction to the interface.

- Usage Instructions : Step-by-step instructions for end-users.

10.2 Technical Training

- Architecture Overview : Detailed explanation of the interface architecture.

- Development Guidelines : Best practices and coding standards for developers.