Healthcare EDI Integration Project: Requirement Specification Document

1. Project Overview

Project Name: Healthcare EDI Integration System

Objective:

- Automate the exchange of healthcare EDI transactions between providers and payers.
- Ensure compliance with HIPAA, SNIP validations, and secure data transfer protocols.

Scope:

- Support key EDI transactions:
  - **837**: Healthcare Claim Submission.
  - **835**: Payment Remittance Advice.
  - **270/271**: Eligibility Inquiry and Response.
  - **834**: Benefit Enrollment and Maintenance.
  - **820**: Premium Payment.
  - **275**: Personal Health Record Transfer.
  - **832**: Health Care Fee Schedule.
  - **274**: Health Care Provider Directory.
  - **277**: Health Care Claim Acknowledgment.
  - **276/277**: Health Care Claim Status Inquiry and Response.
  - **269**: Health Care Benefit Coordination Verification.

- Integrate data mapping, validation, API development, and ESB workflows.

2. Functional Requirements

2.1 EDI Transactions

- **837**: Translate and transmit healthcare claims from providers to payers.
- **835**: Process payment remittance advice from payers to providers.
- **270/271**: Handle eligibility inquiries and responses in real time.
- **834**: Manage benefit enrollment and updates.
- **820**: Handle payroll deducted and group premium payments.
- **275**: Transfer personal health record (PHR) information between health plans.
- **832**: Transmit healthcare fee schedules to providers.
- **274**: Share provider directory information.
- **277**: Acknowledge receipt and status of healthcare claims.
- **276/277**: Request and retrieve healthcare claim statuses.
- **269**: Verify health care benefit coordination between payers.

2.2 API Features

- **Claim Submission API**:
  - Endpoint: `POST /api/claims`
  - Accept claims in JSON format and convert to X12 837.
  - Return acknowledgment (997/999) status.
- **Eligibility API**:
  - Endpoint: `POST /api/eligibility`

    - Accept eligibility requests in JSON and respond with 271 data.
- **Enrollment API**:
    - Endpoint: `POST /api/enrollment`
    - Handle benefit enrollment data in JSON and convert to X12 834.
- **Payment API**:
    - Endpoint: `POST /api/premiums`
    - Transmit premium payments using X12 820.
- **PHR Transfer API**:
    - Endpoint: `POST /api/phr-transfer`
    - Manage personal health record transfers using X12 275.

2.3 Validation

- Implement SNIP Levels 1–7:
    - Level 1: Basic EDI Syntax.
    - Level 2: HIPAA-Specific Syntax.
    - Level 3: Balancing (e.g., totals in 835 files).
    - Level 4: Situational Rules.
    - Level 5: Code Set Validation (e.g., ICD-10, CPT).
    - Level 6: Product-Specific Requirements.
    - Level 7: Custom Trading Partner Rules.

2.4 Data Mapping

- Use **ITX/ITXA** for mapping data:
    - Convert JSON/XML to X12 formats (e.g., 837, 835, 834).
    - Translate X12 data back to JSON/XML for internal systems.

2.5 Error Handling

- Log all errors (e.g., validation failures, transmission errors).
- Provide retry mechanisms for failed transactions.

2.6 Batch and Individual Transaction Simulation

- **Batch Processing**:
  - Simulate processing multiple claims or remittances in a single file.
  - Example: A batch of 837 claims submitted together via `POST /api/batch-claims`.
  - Validate the entire batch and log errors for individual transactions within it.
- **Individual Transactions**:
  - Simulate real-time submission and processing of single claims or inquiries.
  - Example: A single 837 claim submitted via `POST /api/claims` with immediate acknowledgment.

3. Technical Requirements

3.1 Communication Protocols

- **AS2**: Secure transmission to trading partners.
- **SFTP**: Batch file exchanges.
- **HTTPS**: Secure API communications.

3.2 Security Standards

- Ensure **HIPAA compliance**:

    - Use encryption (TLS/SSL).
    - Implement OAuth2 for API authentication.
- Maintain audit trails for all transactions.

3.3 Tools and Technologies

- **Data Mapping**: ITX/ITXA.
- **Programming Languages**: Java, SQL, Python.
- **API Gateway**: AWS API Gateway or Kong.
- **ESB System**: MuleSoft or Apache Camel.
- **Database**: PostgreSQL for transaction logs and data storage.

4. Use Case Scenarios

Use Case 1: Submit Healthcare Claim (837)

- **Actor**: Healthcare provider.
- **Flow**:
    1. Provider submits a claim using the API (`POST /api/claims`).
    2. System validates the claim data (SNIP Levels 1–7).
    3. ITX maps JSON claim data to X12 837 format.
    4. Claim is transmitted securely via AS2 to the payer.
    5. Acknowledgment (997/999) is logged and returned to the provider.

Use Case 2: Retrieve Payment Remittance (835)

- **Actor**: Provider billing system.

- **Flow**:
    1. Provider queries remittance data using the API (`GET /api/remittances`).
    2. System retrieves X12 835 files from payers.
    3. ITX converts 835 data to JSON format.
    4. Data is returned to the provider system.

Use Case 3: Benefit Enrollment (834)

- **Actor**: Employer or health plan administrator.
- **Flow**:
    1. Employer submits benefit enrollment details using the API (`POST /api/enrollment`).
    2. System validates the enrollment data and converts it to X12 834 format.
    3. Enrollment information is transmitted securely to the payer via AS2.
    4. Acknowledgments and errors are logged and reported back to the employer.

Use Case 4: Eligibility Inquiry and Response (270/271)

- **Actor**: Healthcare provider.
- **Flow**:
    1. Provider submits an eligibility inquiry using the API (`POST /api/eligibility`).
    2. System validates the request and converts it to X12 270 format.
    3. Inquiry is securely transmitted to the payer via AS2.
    4. Payer responds with X12 271 eligibility details.
    5. ITX maps the 271 data to JSON format and returns it to the provider.

Use Case 5: Premium Payment (820)

- **Actor**: Employer or government agency.
- **Flow**:
  1. Employer submits premium payment details using the API (`POST /api/premiums`).
  2. System validates and converts the data to X12 820 format.
  3. Premium payments are transmitted securely to the payer.
  4. Acknowledgments are logged and returned to the employer.

Use Case 6: Personal Health Record Transfer (275)

- **Actor**: Health plan or employer.
- **Flow**:
  1. Health plan requests a personal health record transfer via the API (`POST /api/phr-transfer`).
  2. System validates and converts the data to X12 275 format.
  3. Personal health records are securely transmitted to the requested entity.
  4. Acknowledgments are logged and shared with the requestor.

Use Case 7: Fee Schedule (832)

- **Actor**: Health plan or payer.
- **Flow**:
  1. Health plan submits a fee schedule using the API (`POST /api/fees`).
  2. System validates and converts the data to X12 832 format.
  3. Fee schedules are transmitted to providers securely.
  4. Errors and acknowledgments are logged and reported.

Use Case 8: Health Care Claim Status (276/277)

- **Actor**: Healthcare provider.
- **Flow**:
  1. Provider submits a claim status request using the API (`POST /api/claim-status`).
  2. System validates and converts the data to X12 276 format.
  3. Claim status request is securely transmitted to the payer.
  4. Payer responds with X12 277 claim status details.
  5. ITX maps the 277 data to JSON format and returns it to the provider.