Health Claims Exchange: Standardize the claims process, creating a transparent and rule-based platform

Proposal for Code4GovTech 2023| Project idea: Health claims exchange

June 05, 2023

Portfolio

NAME	Aishwarya N
CONTACT INFORMATION	aishnana.03@gmail.com +91-8838809508
CURRENT OCCUPATION	Undergraduate Student
EDUCATIONAL DETAILS	College Name: SASTRA Deemed University Degree: B.Tech Electrical And Electronics Engineering
TECHNICAL SKILLS WITH LEVEL	JavaScript - Expert Python - Expert Django and Frameworks - Expert MySQL - Intermediate Figma UI/UX - Intermediate MATLAB - Intermediate Google Cloud - Intermediate DevOps- Novice

Abstract

How does the current process of Health insurance claims run in India:

A patient submits their KYC document to the hospital. The hospital then provides them with a claim form. This form is filled out manually and then scanned together with any supporting documentation. These scanned copies are subsequently emailed to the insurer or, if a third-party administrator (TPA) is involved, to the TPA. In some instances, they might submit these to the site for the insurer or aggregator. Once the medical insurance provider or TPA has this data, they will digitize it and check all the paperwork before sending it to the appropriate team for manual adjudication.

How are Health claims exchange different from current operations in India:

The Health claims exchange(HCX) protocol is used to exchange claims-related information across various entities such as payers, providers, beneficiaries, regulators, and observers. To guarantee that the information being transmitted is accurate and reliable, it is designed to be interoperable, machine-readable, auditable, and verifiable. The FHIR-based e-claims filed by providers via the HCX over standard protocols (APIs) will operate as a gateway for the ecosystem, serving as a platform (with routing and validation capabilities) through which the insurers and TPAs will send a vision to standardize the claims process, HCX will lower operational costs and boost trust between payers and providers. It will also lower the cost of processing claims, speed up pre-authorization or patient discharge approvals, improve patient experience, provide better visibility for tracking claims, and produce better quality data for the industry and regulators.

How can it make a difference at Nation level:

The nation's existing system for settling health insurance claims is largely manual, non-digital, and cumbersome, causing difficulties at every step. The ecosystem's present mechanism for exchanging claims lacks standardization, with the majority of data interchange taking place through manual processes. Insurers, TPAs, and providers each have very different processes, which makes processing each claim expensive.

Health Claims Exchange (HCX), the new process is projected to be substantially faster than the present approach. Everyone will benefit from a faster turnaround time for hospital beds as a result. The HCX procedure is anticipated to be a lot more precise and organized as well. Additionally, it will result in cost reductions for everyone, including insurance companies and hospitals. The government will have improved access to medical information at the demographic level. This would enable them to make better decisions regarding the deployment

of healthcare assets and infrastructure across the country, thereby making inexpensive and high-quality healthcare available to an increasing number of people. Additionally, it will reduce instances of fabricated, erroneous, or fraudulent claims.

PROJECT DETAILS:

Project Overview

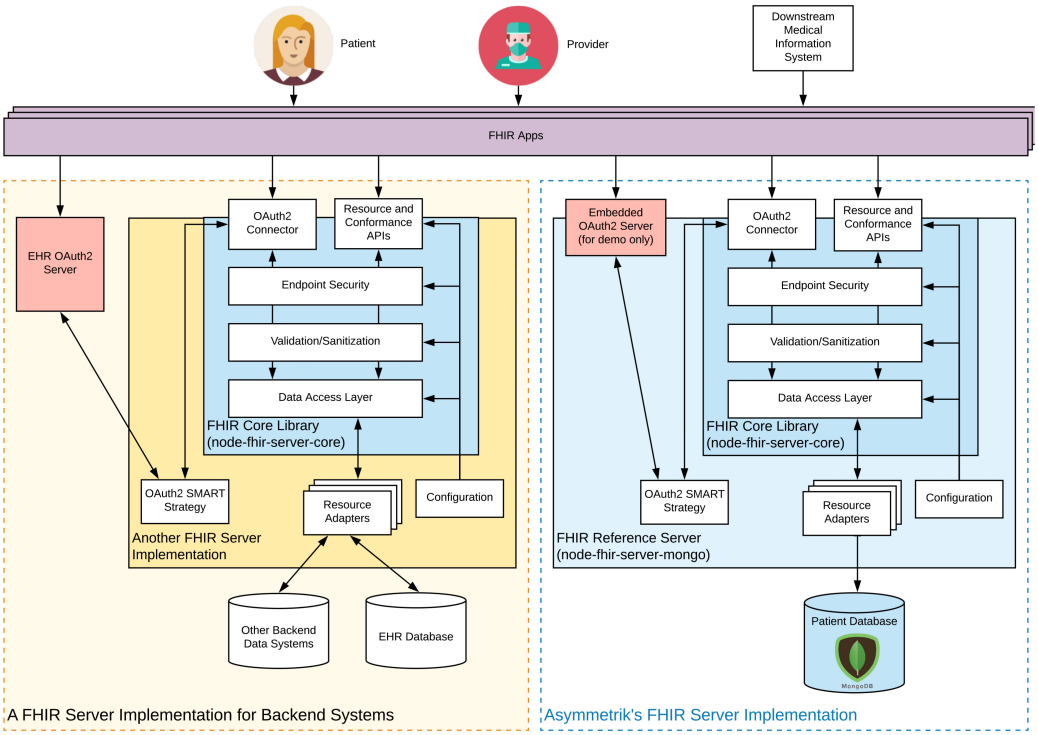
The main objective of the project is to create an open-source project, that serves as a platform for widely accepted Health Claims data exchange specifications as a public good that can be accepted. It should be open to supporting technology and vendor neutrality, and adaptable to changes and innovations.

Key design principles for open specifications

Fast Healthcare Interoperability Resources(FHIR) Architecture

FHIR is built on top of the HTTPS (HTTP Secure) protocol, analytics tools can access and interpret FHIR resources to gather real-time data. By using certain resource models, healthcare organizations would be able to collect data in real time. FHIR resources can be streamed to a data repository where they can be correlated with other informatics data. Tracking epidemics, detecting prescription drug fraud, alerting users to dangerous drug interactions, and cutting down on wait times at ERs are just a few potential use cases. Resources are the fundamental units of the FHIR architecture. Patients, doctors, prescriptions, diagnoses, observations, and procedures are examples of discrete units of healthcare information represented by resources. Each resource has a distinct identification and a predetermined set of elements that stand in for particular qualities and connections. Resources are often represented in JSON format and are developed using a data modeling technique. For performing common operations on resources, FHIR specifies a set of predefined interactions.

These interactions range from general ones like Search, History, and Operations to more focused ones like Create, Read, Update, and Delete (CRUD). Systems can readily communicate and exchange medical data by adhering to these standardized interactions.



https://www.figma.com/file/eitxgWA94ckRc6ZN0F5r8p/FHIR-Architechture?type=whiteboard&node-id=875-1066&t=5GAVUr8bcW4zeBCy-0

Data Modelling for Health Claims Exchange (HCX)

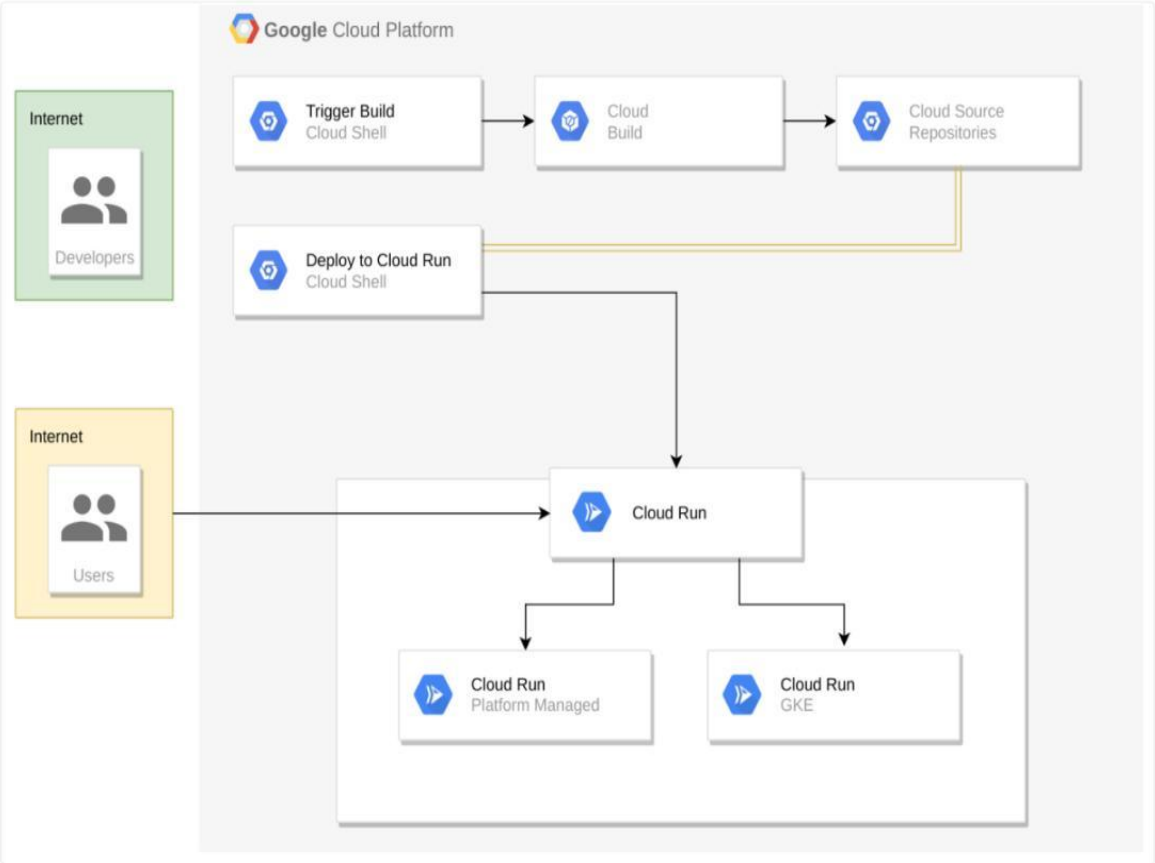
Patients, providers, payers, claims, diagnoses, procedures, prescriptions, and supporting documentation are among the essential entities involved in the health claims exchange process. Patient traits may include demographic information, insurance information, and unique identifiers. The name, contact information, and specialization of the provider are all possible provider properties. The properties of claims can include dates,

billing codes, and claim status. A claim, for instance, is connected to a patient, a provider, and a payment. Depending on the needs of the business, we have to specify the cardinality of the relationships, such as one-to-one, one-to-many, or many-to-many. Multiple diagnosis codes connected with a claim or multiple procedures inside a single claim are examples of hierarchies in health claims. To appropriately describe the claim specifics and supporting data, take into account these hierarchical linkages in your data model. Health claims rule and standards may change over time, therefore data architecture should allow for easy migration to newer versions without losing any data.

Development environment and server

To deploy and test the health claims exchange system, we need to set up a development server or cloud environment. It can be used to experiment with and improve our code on this server by simulating the production environment without affecting the real system. The google cloud console is a web-based administration interface provided by Google Cloud Platform (GCP) that enables the management of cloud resources, establishing services, monitoring use, and executing various administrative duties. An interface for configuring and managing cloud services and APIs is provided by the Google Cloud Console. We can activate and customize particular services and APIs, such as Cloud Functions, Cloud Pub/Sub, and Google Cloud Storage, with which our SDK communicates. The Google Cloud Console provides monitoring and logging options to track the usage and performance of your SDK. To obtain insights into how our SDK is being used, we can check logs, set up alerts, and monitor metrics.

https://www.figma.com/file/1iFPuk1ySO6QGXKOvL02TE/GCP?type=whiteboard&node-id=875-1066&t=KDXrlxi2cug0udgJ-4



Security and Compliance

We need to use effective encryption techniques to safeguard health claims data while it is being sent and stored. Using industry-standard encryption protocols and techniques, such as TLS/SSL, to protect data while it is in transit and at rest can be an effective way of data encryption. Strong access control procedures are in place to ensure that only authorized personnel or systems have access to health claims data. Employ robust authentication mechanisms, such as multi-factor authentication (MFA), and implement stringent access rules based on roles and permissions. We must gather and keep the minimal amount of health claims information required for the intended use. Avoid keeping needless or sensitive information that could jeopardize your privacy. To ensure that data is maintained for as long as necessary and safely disposed of when no longer needed, we

must implement data retention policies. We must Conduct frequent vulnerability assessments and penetration testing to detect and address potential security flaws in our systems. We need to keep all elements of our health claims exchange infrastructure, such as software libraries, frameworks, and operating systems, up to date with security patches and updates.

API Design

The API design must include security safeguards and complies with any privacy and security laws, such as the HIPAA (Health Insurance Portability and Accountability Act). The HIPPA goals are:

1. To limit the use of protected health information to those with a “need to know”.
2. To penalize those who do not comply with confidentiality regulations.

We can create standard data formats for API queries and answers. Use well-defined data models that are in line with the criteria to establish health claims. Based on industry standards, take into account using JSON formats for data serialization.

JSON Object

```
{
  "PatientId": "12345",
  "FirstName": "Akshay",
  "LastName": "Kumar",
  "DateOfBirth": "1980-01-01"
}
```

JSON Arrays for multiple inputs

```
"Diagnoses": [
```

```
  {
   "Code": "AA12",
   "Description": "Diagnosis 1"
  },
  {
   "Code": "BB12",
   "Description": " Diagnosis 2"
  }
]
```

Nested Objects

```
"Address": {
  "Street": "123 Main St",
  "City": "Chennai",
  "State": "Tamilnadu",
  "PostalCode": "603202"
}
```

Date and Time Format

```
"DateOfBirth": "1980-01-01",
"AdmissionDate": "2023-01-12T10:30:IST"
```

Null values

```
{
  "Attribute1": "value1",
  "Attribute2": null
```

}

Handling Code Systems and Value Sets

{

"Code": "12345",

"System": "http://example.com/codesystem/icd-10"

}

Implement pagination and filtering options in API when working with large datasets to let users retrieve data in digestible pieces. To improve performance and usability, we must offer settings for defining page size, offset, sorting, and filtering criteria. To restrict access to our API, we must use secure authentication and permission procedures. Support industry-standard authentication techniques like OAuth 2.0 or API keys, and make sure that only authorized users or systems have access to sensitive health claims data.

Error Handling and Logging

A consistent format for error responses to provide clear and standardized information to clients is crucial.

1. INVALID_REQUEST_FORMAT: The client's request was not sent in the proper JSON format or structure, resulting in the error code

2. MISSING_REQUIRED_FIELD: The JSON request lacks one or more necessary fields.

3. INVALID_FIELD_VALUE: The JSON request's error code indicates that a particular field's value is incorrect or does not adhere to the requirements.

4. RESOURCE_NOT_FOUND: The system was unable to locate the requested resource.

5. UNAUTHORIZED_ACCESS: The client lacks the necessary authorization or authentication to access the requested resource, resulting in this error.

6. DUPLICATE_RESOURCE: A resource that already exists in the system was created.

7. SERVER_ERROR: While processing the request, an unexpected problem was encountered on the server.

8. SERVICE_UNAVAILABLE: The requested service or endpoint is momentarily unavailable and is marked as SERVICE_UNAVAILABLE.

9. VALIDATION_ERROR: The business logic or validation rules for the JSON request were violated.

10. RATE_LIMIT_EXCEEDED: The client's rate of access to the API has been exceeded.

11. CONFLICTING_REQUEST: The request violates a business rule or conflicts with an already-existing resource.

12. FORBIDDEN_OPERATION: The system does not support or authorize the requested operation.

13. UNPROCESSABLE_ENTITY: The JSON request contains incomplete or erroneous data from a semantic perspective.

14. INSUFFICIENT_PERMISSIONS: The client's permissions are insufficient to carry out the requested operation, as indicated by the error code.

15. INVALID_AUTH_TOKEN: The request's given authentication token is either faulty or out of date.

Put in place a logging system to record mistakes and pertinent contextual data for investigation and analysis. Record important facts including error messages, timestamps, request specifics, user identities, and stack traces. A centralized logging system or a specific log file should be used to store log information.

Testing and Validation

- Unit testing: Use unit testing to check the functionality of distinct parts, classes, or functions. Unit tests ensure that each component behaves as intended by concentrating on testing certain functions and use cases.

- Integration Testing: Test the interaction and interoperability between the various modules, services, or systems that make up the Health Claims Exchange by performing integration tests. Any problems that develop when several components are integrated are helped to find via integration testing.

- Data Validation: Verify the accuracy of the data being transferred between parties in the Health Claims Exchange system. Ascertain that the incoming and existing data conform to the prescribed data models, structures, and coding standards. Validate the data for accuracy and completeness to avoid data integrity issues.

- Security Testing: Perform security testing to identify vulnerabilities and assure the system's robustness. Injection attacks, cross-site scripting (XSS), and cross-site request forgery (CSRF) are a few examples of frequent security flaws that should be tested for. Perform penetration testing to imitate real-world attacks and assess the system's resistance to harmful actions.

- Compliance Testing: Ensure that the Health Claims Exchange system complies with applicable compliance standards and regulations, such as HIPAA (Health Insurance Portability and Accountability Act) for data protection and security. To protect sensitive health information during exchange and storage, make sure the proper safeguards are in place.

- Performance Testing: Analyse the Health Claims Exchange system's performance and scalability under various load circumstances. Examine the system's ability to handle a high

amount of requests, record response times, and find any bottlenecks or performance issues.

- Interoperability Testing: Verify the system's interoperability with other systems, such as payer systems or EHRs (Electronic Health Records), by conducting an interoperability test. Test interoperability and compatibility with various standards and protocols to guarantee seamless data transfer between systems.

- Validation of Documentation: Check the accuracy and completeness of system documentation, such as API specifications, data models, and user guides. Ensure that the documentation is up to date and reflects the real behavior of the Health Claims Exchange system.

Collaborative tools

The communication, coordination, and information exchange among the parties involved in the Health Claims Exchange is significantly facilitated by collaborative tools.

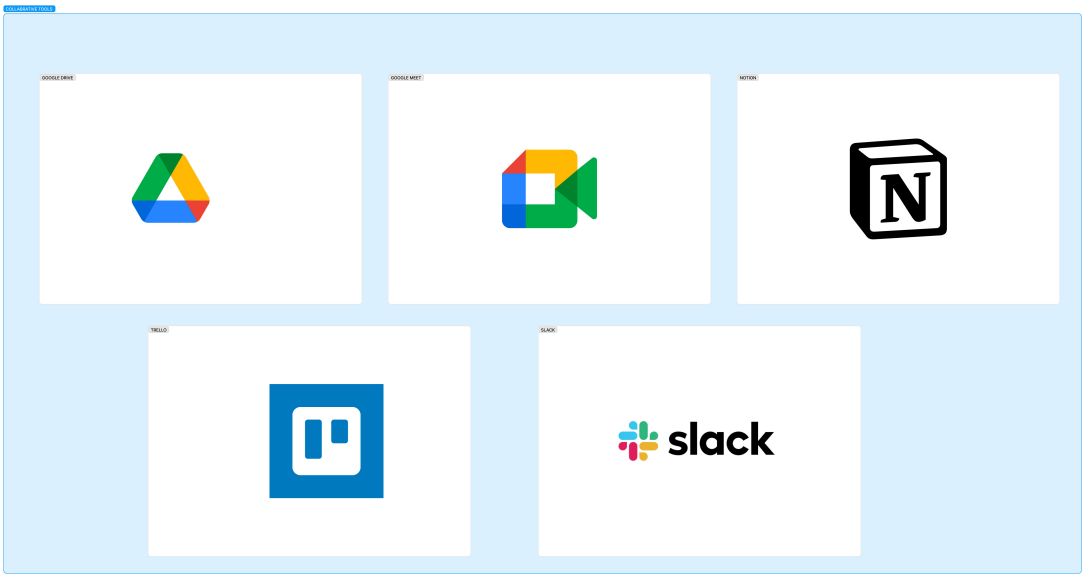
Communication and Messaging tools: Slack is a popular team communication software that enables real-time chat, file sharing, and collaboration via channels and direct communications.

Project Management tools: Trello is a board, list, and card-based visual project management application that allows teams to track progress, assign tasks, and communicate on projects.

Document Collaboration tools: Google Drive is a platform for team collaboration and cloud storage that enables the storing, sharing, and collaborative editing of documents, spreadsheets, and presentations.

Tools for online conferences and video meetings: Google Meet is a Google Workspace-integrated video conferencing service that enables teams to hold online meetings, share screens, and work together in real-time.

Tools for Knowledge Management: Notion is a flexible all-in-one workspace that incorporates wikis, project management tools, notes, and documents to promote collaboration and information exchange.



https://www.figma.com/file/GpkEkRgwQt1ugKMtdMHEcT/Collab?type=whiteboard&node-id=5703-60&t=aGFt9noFGddAOelO-4

Business Associate Agreements

Business Associate Agreements (BAAs) are crucial contracts that outline the duties and obligations of covered entities (such as healthcare providers, health plans, etc.) and their business associates (such as vendors, contractors, and service providers) about handling protected health information (PHI) in the context of exchanging health claims. The Health Insurance Portability and Accountability Act (HIPAA) Privacy Rule mandates BAAs.

1. Explicitly define terms relating to the exchange of health claims, such as covered entity, business associate, PHI, and other phrases.
2. Specify the business associate's permissible uses and disclosures of PHI. This includes specifying the reason for

sharing PHI as well as any restrictions or constraints on its use.
3. Outline the administrative, physical, and technical safeguards that the business associate must implement to secure PHI. This could include encryption, access controls, incident response, and breach reporting regulations.
4. Declare that the business associate will abide by the HIPAA Privacy Rule as well as any other applicable laws and regulations controlling the use and disclosure of PHI.
5. Indicate whether any services relating to the health claims exchange may be subcontracted by the business associate and, if so, specify that the business associate shall have written agreements with its subcontractors containing comparable privacy and security obligations.
6. Indicate the BAA's start and end dates as well as its duration and conditions.

To create and review a BAA that satisfies the unique criteria and conditions of your health claims exchange project, it is essential to work with legal specialists knowledgeable in healthcare and HIPAA compliance. Depending on the type of services rendered, the relationship between the parties, and the relevant rules and regulations in your jurisdiction, each BAA may be different.

Tech Ops Policy Specifications

Onboarding Policies

Onboarding rules for health claims exchange relate to the guidelines and processes that govern the process of bringing new participants, such as healthcare providers, payers, or vendors, onto the health claims exchange platform. These regulations guarantee that participants comprehend and follow the exchange's requirements, standards, and protocols, enabling efficient and secure data sharing. Developing onboarding policies should take the following factors into account:

✓ Participant Qualifications:
Clearly state the prerequisites for participation in the health claims exchange, such as organization type, authorization, accreditation, or adherence to rules.

✓ Application Methodology:
Describe the procedures and supporting materials that participants must provide to be considered for onboarding, such as application forms, legal contracts, and other paperwork.

✓ Sharing of Data and Privacy:
Share the health claims exchange's policies and processes for data access, privacy, and sharing.

✓ Technical prerequisites
Specify the technical requirements, benchmarks, and procedures that users must adhere to join the health claims exchange platform. Specify the criteria for interoperability and the data formats, communications protocols (such as HL7 and FHIR), security protocols (such as encryption and secure connections), and data formats.

✓ Education and Training:
Provide participants with training materials or programs that explain the health claims exchange platform's capabilities, functionality, and best practices for data exchange.
Make sure that participants are aware of their roles and duties, the procedures for submitting data, how to handle errors, and how to troubleshoot.

✓ Continuous Inspection and Compliance:
Describe the continuing checks and balances that participants undergo to ensure that they are still adhering to the health claims exchange rules and onboarding guidelines.
Indicate the penalties for non-compliance as well as the steps to take in the event of a dispute.

✓ Governance and Assistance:
Define the composition, functions, and authority of the governing body in charge of the health claims exchange.
Create a support system to help participants with registration, technical problems, and policy-related questions.

✓ Updates to Policy:
Define the procedure for assessing, communicating, and modifying the onboarding rules as necessary to take into account changing legal requirements or technology developments.

Deboarding policies
Health claims exchange deboarding policies specify the steps and requirements for removing or terminating participants, such as healthcare providers, payers, or suppliers, from the platform. When participants no longer need access to the exchange, these regulations guarantee a secure and easy transfer. Developing deboarding policies should take the following factors into account:

✓ Termination Standards:
Define the requirements for terminating or removing a participant from the health claims exchange. This could involve things like breaking contractual obligations, breaking rules, violating privacy or security regulations, or stopping operations.

✓ Communication and Notification
Create a procedure for alerting participants when they are dismissed or kicked out of the health claims exchange. Establish the means of communication, the time frame, and the details that must be included in the termination notification.

✓ Retention and deletion of data:
Specify the guidelines for data retention and deletion after participant termination. Specify the participant's responsibilities

for safely erasing or returning any PHI or data they have collected from the health claims exchange.

✓ Transition Strategy:
The steps and obligations for the participant to transfer their data, services, or liabilities to another authorized business or the covered entity, as applicable, should be outlined in a transition plan. Set a deadline and specifications for finishing the transition process.

✓ Access suspension:
Give specific instructions on how to deny a participant access to the health claims exchange platform and any related systems or resources. Upon termination, make sure that all user accounts, passwords, and rights are immediately terminated or revoked.

✓ Compliance and auditing:
Verify that the participant who has been terminated has complied with their obligations for data disposal, non-disclosure, and compliance with security and privacy regulations by conducting audits or assessments. Record the audit results and keep them with the records for future use.

✓ Review and updates to policy:
Review and update the deboarding procedures frequently to take into account new legal requirements, industry best practices, and the lessons discovered from past deboarding incidents.

Access control policies
For a health claims exchange, access control measures are crucial to preserving the availability, confidentiality, and integrity of sensitive medical data. These policies specify the conditions under which access to the exchange platform and its resources may be granted or denied. Developing access control policies should take the following factors into account:

✓ RBAC (Role-Based Access Control):
RBAC should be used to assign access rights and permissions based on predefined roles within the health claims exchange. Define user roles for administrators, medical professionals, payers, auditors, and other pertinent groups of people. To secure appropriate access to resources and functionalities, it is important to clearly define the duties and privileges attached to each position.

✓ Data and function access controls:
Depending on the user's duties and responsibilities, implement granular access restrictions to limit user access to particular data items or functionalities inside the health claims exchange. To implement fine-grained access controls based on contextual variables such as user attributes, data classifications, and business rules, employ attribute-based access control (ABAC) or other advanced access control mechanisms.

✓ Access to a secure network:
To prevent unauthorized external access to the health claims exchange platform, use secure network access controls including firewalls, intrusion detection systems, and virtual private networks (VPNs). Make sure that the platform's remote access is protected by encrypted connections and robust authentication procedures.

Exchange operation policies

The principles practices, and standards for the regular operations of the exchange platform are specified in the exchange operation policies for a health claims exchange. These guidelines guarantee efficient operation, data integrity, security, and compliance with legal requirements. The following are some crucial factors to take into account while creating exchange operation policies for a health claims exchange:

✓ Submission of Data and Validation:
Establish the steps and specifications that healthcare providers, payers, and other participants must follow to send their claim data to the exchange. To verify the accuracy, integrity, and completeness of the given data, establish data validation criteria and quality checks. Indicate the format, protocols, and standards (such as HL7 and FHIR) for data submission and integration.

✓ Data Processing and Adjudication:
Describe the steps involved in processing and deciding claims in the context of the health claims exchange. Define the criteria for approving claims, conducting claim verifications, and calculating reimbursements. Set a deadline for processing claims and offer tools for monitoring and addressing any delays or problems.

✓ Planning for business continuity and emergencies:
Create a business continuity plan to ensure the health claims exchange keeps running in the case of disruptions, catastrophes, or disasters. To reduce downtime and data loss, set up backup and recovery processes, redundant systems, and emergency procedures.

✓ Resolution of Disputes:
Create a procedure for discussing and resolving disagreements between participants or with the operators of the health claims exchange. Define means for resolving disputes over claim adjudication, data discrepancies, or operational concerns, including escalation procedures and mediation processes.

Domain Specifications

Domain Data Model
The organization and connections among the data entities involved in the exchange process are depicted in a domain data model for a health claims exchange. It outlines the fundamental

entities, traits, and associations required to collect and exchange health claims data.

Claim:
The claim represents a medical claim that a doctor has submitted to get paid.
The following attributes apply to claims: claim ID, patient ID, provider ID, claim date, claim total, status, etc.

Patient:
The patient represents a patient receiving medical attention.
Patient ID, name, date of birth, gender, contact information, insurance information, etc. are examples of attributes.

Provider:
Provider represents the hospital, clinic, or company submitting the claim.
Details like the provider's ID, name, address, phone number, and specialty attributes.

Payer:
The payer represents an insurance provider or payer in charge of paying the claim's reimbursement.
Indicators: payer identification, name, address, contact information, insurance information, etc.

Diagnosis:
It demonstrates the medical diagnosis connected to a claim.
Description, date, provider ID, diagnostic code, and other attributes.

Procedure:
It represents a medical operation or service rendered in connection with the claim.
Procedure code, description, date, provider ID, linked diagnosis, and other attributes.

Added Supporting Files:
It represents any further paperwork or attachments related to the claim, such as receipts, invoices, or medical records.
attributes such as the linked claim ID, document ID, type, file name, and description.

Payment:
The payment represents the terms of a claim's settlement or reimbursement.
There are several attributes, including payment ID, claim ID, payer ID, payment date, amount, and status.

Coverage:
Coverage represents the specifics of an individual patient's insurance coverage or policy.
the following attributes: benefits, restrictions, start and end dates of coverage, patient and payer IDs, and coverage ID.

The audit trail
It represents a record of events, modifications, or actions affecting a claim.
attributes including action type, timestamp, description, claim ID, user ID, and log ID.

Metadata specifications
A health claims exchange metadata specification provides a standardized framework for describing and organizing the data components and features connected with health claims. These requirements facilitate system and stakeholder interoperability, data interchange, and consistency.

Metadata for Claims:
Claim ID: An identifier that is assigned to each health claim.
The date on which the claim was submitted or created.
Claim Status: The claim's current status (e.g., pending, approved, or refused).
Total Amount: The total amount sought in reimbursement.

Last Updated Date: The date and time the claim was most last updated or changed.

Metadata on Patients:
A unique identity provided to each patient is known as a patient ID.
Patient Name: The name of the person receiving medical attention.
Date of Birth: The patient's birthday.
Gender: The patient's gender (e.g., male, female, or other).
Contact Information: The patient's contact information (e.g., address, phone number, email).

Metadata for Providers:
A unique identity provided to each healthcare practitioner or organization is known as a provider ID.
Name of Healthcare Provider: The name of the healthcare provider or organization.
Address: The provider's physical address.
Contact Information: The provider's contact information (e.g., phone number, email).

Metadata for Payers:
Payer ID: An identifier that is unique to each insurance company or payer.
Payer Name: The insurance company or payer's name.
The location of the payer's office.
Contact Information: The payer's contact information (e.g., phone number, email).

Metadata about Diagnosis:
Diagnosis Code: This is the code that represents the medical diagnosis that is associated with the claim.
Diagnosis Description: A description or text that provides more information about the diagnosis.

Metadata about Procedures:

Operation Code: This is the code that represents the medical operation or service that was done as part of the claim.
A procedure description is a description or text that provides more information about the procedure.

Metadata for Documentation Support:
A unique identification issued to each supporting document or file is known as a document ID.
Document Type: The document's type or category (for example, medical report, invoice, or receipt).
Document Description: A description or text that provides further information about the document.

Metadata for Payment:
Payment ID: An identifier that is unique to each payment or reimbursement.
money Date: The date on which the money was received.
Amount: The monetary value of the payment or refund.
Payment Status: The payment's status (e.g., paid, pending, denied).

Proposed Governance Approach for Specifications at iSPIRT

iSPIRT is a non-profit Indian think tank dedicated to the growth and development of the Indian software product business. When it comes to specification governance approaches at iSPIRT, it is critical to build a structure that assures transparency, inclusion, and accountability.

Governance body

Establish a steering committee or governance group that will be in charge of managing the development of the specifications, ensuring that the governance approach is followed, and resolving any conflicts that may occur.
Representatives from iSPIRT, industry leaders, subject matter experts, and other key stakeholders should form the governance

body. To maintain continuity and accountability, define the duties, responsibilities, and tenure of governance body members.

Milestones

Milestone 1: Create the overall system design for the SDK, taking scalability, interoperability, and security into account.
Milestone 2: Consider the relevant data pieces, entities, and relationships while creating the data model and schema for health claim exchange.
Milestone 3: Enable integration with already-existing healthcare systems, including payer systems, electronic medical record (EMR) systems, and government health information exchanges.
Milestone 4: To aid developers in understanding and utilizing the SDK, create extensive documentation, tutorials, and API reference guides.
Milestone 5: To obtain input and identify areas for improvement, do beta testing with a select group of users, healthcare providers, and payers.
Milestone 6: Release the SDK to the developer community and make it widely available for use.

Timeline and Implementations

Date	Tasks
June 24	Candidate Announcement
June 26-29	DGP Bootcamp and Cohort bonding
June 30	Inaugural event
July 01	Improve my knowledge and understanding of project deliverables. Set up regular meetings, and have a great one-on-one connection while soliciting feedback.
July 02-07	Getting started with codebases, and requirements for the project. Discussion over APIs and finalizing the number of APIs.

	Discussion with mentors and implementation of ideas.
July 08-09	Discussion with mentors about the FHIR architecture. Setting up codebase structure.
July 10 - 22	**Milestone 1** Determine the unique integration needs of the healthcare systems with which you want to integrate. Establish processes for authentication and authorization to ensure secure access to the integrated systems. Setting up the basic routing for the components. Deploying the project in GitHub. Conversion of the codebase into a progressive overall system for SDK.
July 23 - 31	**Milestone 2** Determine the critical data elements and entities involved in health claim exchange. Claims, patients, providers, procedures, diagnoses, payments, and supporting documents may be included. To ensure data integrity and eliminate redundancy, use normalization procedures. Add the manifest JSON to the codebases. Developing components to receive data from APIs. Creating a Google Console Project.
August 01-09	**Milestone 3** Determine the data formats, communication protocols, and APIs each system supports. Determine the best communication channels to use when connecting with each healthcare system. APIs, web services, message queues, and other interoperability frameworks may be used.

	Injection of JSON data and testing Dynamic pages. Inspecting all APIs.
August 10 - 17	**Milestone 4** Create detailed documentation and integration instructions for developers and end users. Provide users with technical support and help during the integration process.
August 18-20	**Buffer Time** Work on pending milestones(if any). Else start with Milestone 5
PHASE 1 EVALUATION August 21-22	**Getting all the milestones approved by the mentors.**
August 23-27	**Milestone 5** Determine the beta testing phase's precise goals and objectives. Determine a varied set of beta testers who will represent the target users, such as healthcare providers, payers, and other important stakeholders. Maintain a feedback loop to guarantee that the system adapts in response to changing user demands and expectations.
August 28 - 30	**Milestones 6** Make thorough documentation that includes installation instructions, set-up guidelines, API references, usage samples, and any other pertinent information. Publish the SDK package to a central repository or package management that is commonly used by developers. Monitor market trends and changing developer demands to keep the SDK current and up to date.
September	Post-program dissemination and CCBP

Future Developments

Collaboration and cooperation with system owners, IT teams, and stakeholders are required for integration with current healthcare systems. Clear communication, adherence to standards, and thorough testing are required for successful integration. Integrating into existing healthcare systems, such as payer systems, electronic medical record (EMR) systems, and government health information exchanges, necessitates careful design and execution. Collaborate with domain experts to achieve interoperability and consistency in health claim data interchange by leveraging existing healthcare data models and standards.

It should be noted that the aforementioned milestones are high-level and should be tailored to the specific context and requirements of the health claim exchange project in India. Throughout the SDK development process, regular communication and engagement with stakeholders such as healthcare providers, payers, and regulatory authorities will be critical.

1. Create a dedicated SDK website or portal where developers may view and download the SDK as well as associated materials.
2. Package the SDK in an easily distributable and installable format, such as a compressed archive (e.g., ZIP) or a package manager-specific format.
3. Make sure the package is appropriately versioned to enable future updates and compatibility.
4. Include code snippets, step-by-step instructions, and explanations to assist developers in understanding how to use the SDK in their projects.
5. Set up lines of contact for developers to ask questions, provide feedback, and seek assistance.

6. Promoting the SDK in social media platforms, developer forums, industry-specific groups, and relevant conferences or events to our advantage.
7. To address frequent issues, consider providing online documentation, FAQs, troubleshooting tips, and code samples.
8. Iterate on the SDK continuously depending on feedback to improve its functionality, usability, and performance.

Availability

When do your classes and exam finish?
My examinations for the third year have just finished, thus I have no exams scheduled for the next four months. My classes will begin in offline mode on the 31st of July. They can often extend till 6 p.m. at most, but that won't be an issue because I usually work late. I get up early in the morning and accomplish most of my coding-related work then.
Managing classes in addition to the program and one-on-one mentor interactions would be no problem.

Do you have any plans for full-time or part-time employment or internship this summer?
Yes. I have plans for implant training this summer. I have applied in various sectors. I hope the implant training will be planned for 2 weeks.

How many hours a week can you devote to a summer project?
For the summer project, I will be able to consistently put in an average of 60 hours per week.
I am enthusiastic about this initiative and the prospects it provides for me. I'd like to see this through to completion and contribute to this fantastic effort.

What do you want to gain from C4GT?

- Work experience with a recognized organization, to get myself versed with the work in the real world.
- This project will provide me an opportunity to implement my learnings on a practical scale.
- This project can benefit me from enhanced scrutiny and peer review because of a huge community of contributors. This can help me lead to better code quality, faster bug detection, and faster issue resolution, resulting in more reliable software.
- Working on this project offers me the ability to alter and customize the software to meet their demands.
- I can gain important expertise by contributing to these open-source projects. It enables me to improve my technical abilities, learn from more experienced contributors, and become acquainted with best practices in software development.

Why Me?

I am Reliable and responsible, can bounce back in any circumstance, and can come up with optimal solutions. My enthusiasm for the project can fuel my dedication and motivation to contribute efficiently to the project. My passion and experience in developing unique systems from the ground up are what motivate me and will continue to do so throughout the duration of this project. I believe that it is our responsibility to make the world a better place, to find answers, and to solve problems to make human life more efficient. I want to contribute to this progress and hope that something great emerges at the end.

I believe that all of the grounds stated above are sufficient to support my approval as a contributor to the C4GT 2023 program under the Project Idea: Health Claims Exchange.

Thanking Note

Thank you so much to all of the mentors and the entire community for allowing me to work on such an interesting project. I will do all in my power to add value to this community. Thank you for listening to all of my concerns while developing this proposal, and thank you in advance for the mentorship and direction that all of you bring to help us grow and improve our talents. I'm looking forward to the exciting road ahead and am eager to learn new things.