Facultad de Ingeniería
Carrera de Ingeniería de Sistemas

Software integrador de información médica utilizando Health Level Seven y FHIR.

Trabajo de Titulación previo a la obtención del Título de Ingeniero de Sistemas

Autor:

Kevin Alexander Maxi Jara
Cl: 0106604416
Correo electrónico: kevin.maxi.jara@gmail.com

Director:

Dr. Carlos Villie Morocho Zurita.
Cl: 0300930328

Cuenca-Ecuador
19-septiembre-2022

RESUMEN

Fast Healthcare Interoperability Resources (FHIR) y *Health Level 7* (HL7) son dos estándares que ayudan el intercambio de datos clínicos y que en los últimos años han ganado bastante protagonismo en la industria médica, ambos son estándares de información médica cuya propuesta es alcanzar la interoperabilidad entre los diferentes sistemas de proveedores de servicios médicos. FHIR y HL7 permiten reducir los esfuerzos de la industria médica a la hora de interoperar con otros entes médicos, puesto que brinda una sola estructura de datos que pretende ser universal, además FHIR se construyó con el enfoque de transferencia de estado representacional o REST que es la interfaz para conectar varios sistemas a través de HTTP.

Sin embargo, en la industria médica latinoamericana y en específico en Ecuador, los esfuerzos para alcanzar la interoperabilidad de los sistemas médicos se han visto opacados por muchos aspectos, el más evidente es la inversión que conllevaría construir nuevos sistemas médicos que hagan uso de estos estándares y dejar obsoletos a los anteriores sistemas, o al menos a gran parte de su extracción de datos. Sin embargo, este proceso es necesario, para poder cumplir con la universalidad propuesta por el estado ecuatoriano donde se define la necesidad de extender la cobertura de los beneficios del sistema de salud, a toda la población. En este trabajo de titulación se presenta el diseño e implementación de un middleware de información médica usando estándares HL7 y FHIR, que permite la interoperabilidad de sistemas de información médica.

Palabras claves: Cuidado integrado. E-Health. FHIR. HAPI FHIR. HL7. Interoperabilidad. Integración. Middleware. Servicios web.

ABSTRACT

Fast Healthcare Interoperability Resources (FHIR) and Health Level 7 (HL7) in recent years have gained considerable prominence in the medical industry, both are medical information standards whose purpose is to achieve interoperability between the different systems of medical service providers. FHIR and HL7 reduce the efforts of the medical industry when it comes to interoperating with other medical entities, since they provide a single data structure that aims to be universal, and FHIR is built with the REST approach.

However, in the Latin American medical industry and specifically in Ecuador, the efforts to achieve medical interoperability have been overshadowed by many aspects, the most evident being the investment that would entail building new medical systems that make use of these standards and render obsolete to previous systems, or at least to much of their data mining. However, this process is necessary in order to comply with the universality proposed by the Ecuadorian state, where it is defined as extending the coverage of the benefits of the health system to the entire population. For this reason, in this degree work, the design and implementation of a medical information middleware is presented using the HL7 and FHIR standards, which allows the interoperability of medical information systems.

Keywords: E-Health. FHIR. HAPI FHIR. HL7. Integrated care. Integration. Interoperabillity. Middleware. Web services.

ÍNDICE

CAPÍTULO 1: INTRODUCCIÓN 14

1.1. Motivación y contexto 14
1.2. Planteamiento del problema 15
1.3. Estado del arte 16
1.4. Arquitectura de emparejamiento propuesta 18
1.5. Contexto del proyecto 19
1.6. Hipótesis y Objetivos 19
1.6.1. Hipótesis 20
1.6.2. Objetivo General 20
1.6.3. Objetivos Específicos 20
1.7. Estructura del trabajo 20

CAPÍTULO 2: MARCO CONCEPTUAL 22

2.1. Historial Clínico Electrónico 22
2.2. Sistemas de información médica 22
2.2.1. Límites de un Sistema de información Médica 23
2.3. Interoperabilidad 23
2.4. El reto de la interoperabilidad 24
2.4.1. Capas de la interoperabilidad 24
2.4.2. ¿Por qué los estándares son necesarios? 24
2.4.3. Complejidad 25
2.4.4. Traducciones 26
2.5. Health Level Seven (HL7) 26
2.5.1. HL7 versión 3 27
2.6. Fast Healthcare Interoperability Resources 28
2.6.1. Proceso de desarrollo FHIR 28
2.6.2. Artefactos FHIR 29
2.6.3. Beneficios de FHIR 30
2.6.4. API's 30
2.6.5. Interfaces RESTful 31
2.6.6. Repositorio 33
2.6.7. Recursos 34

2.7. CIE10 35
2.8. Expediente único para la Historia Clínica Ecuatoriana 35

CAPÍTULO 3: METODOLOGÍA 37

3.1. Procedimiento para la conversión 37
3.2. Requerimientos y diagramas del software 39
3.2.1. Obtención y análisis de requisitos 39
3.2.1.1. Captura de requisitos 39
3.2.1.2. Requisitos obtenidos 41
3.2.1.3. Elaboración de los requisitos 42
3.2.1.4. Definición de los requerimientos prioritarios 43
3.3. Arquitectura REST y comunicaciones 43
3.4. Variables a consideración para establecer la interoperabilidad 44
3.4.1. Formulario 001 44
3.4.2. Formulario 002: 45
3.4.2.1. Anamnesis – Consulta externa 46
3.4.2.2. Consulta externa - Evolución 47
3.4.3. Formulario 003 47
3.4.3.1. Anamnesis 47
3.4.3.2. Examen Físico 48
3.4.4. Formulario 005 50
3.4.4.1. Evolución y prescripciones 50
3.5. Implementación del proveedor/extractor de datos 50
3.5.1. Seguridad implementada 51
3.5.2. Manejo de la extracción de datos 51
3.6. Construcción de los servicios para FHIR y HL7 54
3.6.1. Controladores REST 56
3.6.2. Módulo CIE 10 61
3.7. Implementación del sistema 61
3.7.1. Modelo lógico obtenido 67
3.7.2. Herramienta de gestión de usuarios 69

CAPÍTULO 4: ANÁLISIS Y PRESENTACIÓN DE RESULTADOS 72

4.1. Variables 73

4.1.1. Variables independientes 73
4.1.2. Variables dependientes 73
4.1.3. Grupo de control: 74
4.2. Métricas 74
4.2.1. Tasa de transformación de datos: 74
4.3. Material Experimental 74
4.3.1. Casos de prueba 75
4.3.2. Documentos para la capacitación 79
4.3.3. Documento para la evaluación 79
4.3.4. Encuestas 79
4.4. Diseño experimental 81
4.5. Procedimiento experimental 82
4.5.1. Lectura de requerimientos de la transformación 83
4.5.2. Inserción al MDE 84
4.5.3. Inserción al middleware 84
4.5.4. Ejecutando la evaluación 85
4.5.5. Evaluación de las conclusiones 85
4.6. Definición y selección de la muestra 86
4.7. Pruebas de funcionalidad 86
4.8. Resultados de las pruebas de funcionalidad 90
4.9. Análisis de los resultados 93
4.9.1. Prueba de normalidad de Shapiro-Wilk 93
4.10. Análisis de rendimiento de usuario 98
4.11. Amenazas a la validez 99

CAPÍTULO 5: CONCLUSIONES Y TRABAJO FUTURO 101

5.1. Conclusiones 101
5.2. Líneas de trabajo futuro 102

REFERENCIAS 104

Anexo 1 – Especificación de Requerimientos: 106
Anexo 2 – Determinación de recursos FHIR: 128
Anexo 3 – Ejercicio de capacitación para el middleware (Desarrolladores): 136
Anexo 4 – Ejercicio de capacitación para el middleware (Administradores) 145

Anexo 5 – Ejercicio de evaluación para el middleware (desarolladores)..... 155
Anexo 6 – Ejercicio de evaluación para el middleware (Administradores) 168

Cláusula de licencia y autorización para publicación en el Repositorio Institucional

Kevin Alexander Maxi Jara en calidad de autor y titular de los derechos morales y patrimoniales del trabajo de titulación "Software integrador de información médica utilizando Health Level Seven y FHIR.", de conformidad con el Art. 114 del CÓDIGO ÓRGANICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN reconozco a favor de la Universidad de Cuenca una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos.

Asimismo, autorizo a la Universidad de Cuenca para que utilice la publicación de este trabajo de titulación en el repositorio institucional, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Cuenca, 19 de septiembre de 2022

Kevin Alexander Maxi Jara

C.I.: 0106604416

Claúsula de Propiedad Intelectual

Kevin Alexander Maxi Jara, autor del trabajo de titulación "Software integrador de información médica utilizando Health Level Seven y FHIR.", certifico que todas las ideas, opiniones y contenidos expuestos en la presente investigación son de exclusiva responsabilidad de su autor.

Cuenca, 19 de septiembre de 2022

Kevin Alexander Maxi Jara

C.I.: 0106604416

AGRADECIMIENTOS

Agradezco a mi director, Ing. Villie Morocho Zurita, por la confianza brindada al permitirme contribuir en el desarrollo de este proyecto como tema de titulación al cual le hemos puesto toda nuestra dedicación.

A la Rama Estudiantil IEEE por brindarnos siempre un espacio para estudiar, hacer nuestros trabajos y ayudarnos en los mismos. Además de permitirnos conocer a grandes personas durante este tiempo.

A nuestros compañeros por darse un tiempo para ayudarnos con la evaluación de nuestra tesis.

A mis amigos Daniel Peralta y María Caridad Cáceres, con quien he empezado este camino de tesis juntos y me han acompañado durante todo el trayecto.

A mis familiares, amigos y docentes, por ser parte de nuestra vida, personal y universitaria. Por alentarme a continuar cada día para dar lo mejor de mí.

Y finalmente, a la Facultad de Ingeniería y la Universidad de Cuenca por prepararme como buen profesional, y brindarme los mejores años y recuerdos.

DEDICATORIA

A mis padres, Lorena Jara y Diego Maxi, por brindarme todas las posibilidades para llegar hasta este punto. Han sido guía y ejemplo de esfuerzo y me han apoyado cada día para poder conseguir este mérito.

A mis abuelitos Beatriz Valladarez y Oswaldo Jara quienes han visto por mi desde que he sido un niño y de quienes he aprendido a seguir creciendo día a día

A mis hermanos, Bryam Maxi, Paulette Ojeda, Adriel Ojeda, Camila Maxi y Emilio Maxi, quienes son mi inspiración para seguir mejorando día con día.

A mi novia Jackeline Pañi quien ha sido mi apoyo y fortaleza durante el proceso de este trabajo y en mi vida.

A mis mejores amigos Daniel y Luis, por ayudarme a crecer personal y profesionalmente. A mis amigos de la facultad con quienes compartí las aulas y las arduas labores de estudio, quienes en diferentes maneras fueron inspiración y ayuda tanto en mi vida profesional como en mi vida.

A la Universidad de Cuenca por ser mí segundo hogar. A mis compañeros y docentes, porque gracias a sus enseñanzas he terminado esta etapa y puedo continuar con mi vida profesional.

ACRÓNIMOS

A

API
Application Program Interface
ASSA
Agenda de Salud Sostenible de las Américas

C

CMD
Conjunto mínimo de datos

E

EHR
Electronic Health Record

F

FHIR
Fast Healthcare Interoperability Resources

H

HCE
Historia clínica electrónica
HL7
Health Level Seven

J

JSON
JavaScript Object Notation

M

MDE
Model Driven Engineering

P

PSM
Proveedor de servicios médicos

R

REST
Representational State Transfer
RDF
Resource Description Framework

X

XML
Extensible Markup Language

Yo, Kevin Alexander Maxi Jara en calidad de autor y titular de los derechos morales, patrimoniales e intelectuales del trabajo de titulación “Software integrador de información médica utilizando Health Level Seven y FHIR.”, manifiesto que la idea presentada en este trabajo de investigación fue planteada por la empresa INNTRATEC S.A.S. A petición de la empresa esta idea está sujeta a cláusulas de confidencialidad, por lo que se mantendrá como confidencial, salvo que la otra parte, proporcione el permiso expreso y por escrito para revelarla.

CAPÍTULO 1: INTRODUCCIÓN

En la actualidad uno de los requisitos claves para mejorar la calidad de la atención médica es la interoperabilidad (Sharma & Aggarwal, 2019). Con el rápido incremento de la diversidad de modelos y la escala de los datos relacionados a la salud, cada vez se generan más problemas para cumplir con esta característica.

La transformación digital en el sector de atención médica ha provocado la transición desde los papeles de evolución clínica a una amplia adopción de “Historias Clínicas Electrónicas” (HCE) o “Electronic Health Record” (EHR) que pondrá una gran cantidad de datos a disposición de varios actores.

Entre estos actores, se incluyen ejecutivos, médicos, investigadores, entre otros, para diversos fines, incluida: la administración, la gestión, la práctica clínica y la investigación (Boussadi & Zapletal, 2017).

Sin embargo, las bases de datos que contienen este tipo de información son difíciles de consultar. Esto, se debe en gran medida a la heterogeneidad en el diseño e implementación de las bases de datos y sus componentes.

La falta de estandarización es evidente, lo que complica de manera significativa la interoperabilidad entre sistemas (Boussadi & Zapletal, 2017).

1.1. Motivación y contexto

Los problemas de interoperabilidad derivan en: perdida de historial clínico previo de pacientes, lo que evita la movilidad y los restringe a un solo proveedor de servicios médicos.

Provoca además una eficiencia pobre de los recursos (escasos ya en nuestro medio) y de manera indirecta se afecta la universalidad de la salud descrita en el acuerdo ministerial Acuerdo Ministerial NO. 00001190 (Ministerio de Salud Pública, 2011). En dicho acuerdo se establece cómo necesario: “extender la cobertura de los beneficios del sistema de salud, a toda la población”.

Otros países como, por ejemplo: India, han mejorado la seguridad pública drásticamente con la implementación de estándares de interoperabilidad adecuados. Donde la información precisa del paciente puede ser accesible en cualquier momento y

desde cualquier lugar. El uso adecuado de los estándares disponibles pueden mejorar el significado semántico y sintáctico de la información correcta (Sharma & Aggarwal, 2019).

Entre los estándares para información médica existentes está el marco *Fast Healthcare Interoperable Resources* (FHIR[1], por sus siglas; estándar HL7[2] que ha sido adoptado rápidamente por la comunidad sanitaria). Este estándar parece ser el candidato probable para superar dicho desafío. Además, también es compatible con un modelo de información enriquecido que permite lograr la interoperabilidad semántica de los datos clínicos. Las tres opciones de modelos son XML, JSON y RDF (Leroux et al., 2017)

Por lo antes mencionado, en el acuerdo ministerial NO. 00001190 (Ministerio de Salud Pública, 2011) se detalla la aprobación de uso de HL7 y su implementación en todas las instituciones del sistema nacional de salud. Además, estos estándares son necesarios para cumplir con el objetivo 6 de la Agenda de Salud Sostenible de las Américas 2018-2030 (ASSA, por sus siglas) (Organización Panamericana de la Salud, 2017) para permitir la integración técnica de los subsistemas de información.

En lugar del enfoque tradicional centrado en documentos, HL7 FHIR, adopta un enfoque modular y representa los datos de atención médica atómicos/granulares como entidades modulares independientes denominadas *Resources*.

Estos recursos FHIR se administran mediante *Application Program Interfac*es (API's) y servicios web RESTful, la columna vertebral de las aplicaciones web modernas (Saripalle et al., 2019).

1.2. Planteamiento del problema

El principal objetivo de este trabajo de titulación es proporcionar un software de tipo **middleware** que permita obtener y servir los diferentes datos de los pacientes (sin importar su forma interna), desde diferentes proveedores de servicios médicos, utilizando el estándar FHIR y HL7. A esto se ha sumado una capa de seguridad proporcionada por la tesis: "Propuesta de Identidad Digital para Historial Clínico

[1] FHIR v4.0.1 (hl7.org)
[2] Health Level Seven International

Unificado utilizando tecnología Blockchain" (Cáceres & Peralta, 2021). Habilitando así la interoperabilidad entre los diferentes sistemas de información médica que formen parte del consorcio.

El presente trabajo propone, además, la elaboración de un esquema FHIR y HL7 para representar los siguientes documentos descritos en el expediente único para la historia clínica de la República del Ecuador (Ministerio de Salud Pública del Ecuador, 2006):

- Anamnesis
- Consulta Externa – Anamnesis y EF
- Consulta Externa – Evolución
- Evolución y prescripciones
- Examen Físico

1.3. Estado del arte

Se han analizado los últimos artículos que presentan como integrar los diferentes modelos de datos sistemas heterogéneos de información médica. Entre estas iniciativas se encuentra, HL7 PHR. que es un estándar que aborda las necesidades funcionales de los desarrolladores y usuarios del sistema de registros médicos personales.

Por otro lado, Saripalle et al. (2019) exploran y realizan un análisis crítico del uso de FHIR para lograr la interoperabilidad del Historial Clinico (HC) del paciente, para diseñar y realizar un prototipo móvil interoperable del HC del paciente que se corresponda al modelo funcional HL7 PHR y permite la comunicación bidireccional con OpenEMR[3].

Por otro lado, Kilintzis et al. (2019) presentan una metodología para la implementación de un marco de gestión de datos para la telemedicina, que apoya los servicios de atención integral para pacientes crónicos y multimórbidos. Este marco hace uso de una ontología construida sobre recursos FHIR, para proporcionar

[3] OpenEMR (open-emr.org)

almacenamiento y representación de datos de registros médicos electrónicos enriquecidos semánticamente.

Siguiendo los principios de Datos Vinculados, cabe señalar que, esta solución fue parte del proyecto de la UE WELCOME para la gestión de datos en donde se implementó con éxito un sistema de telemonitorización para pacientes con Enfermedad pulmonar obstructiva crónica (EPOC) y comorbilidades. Kamel y Nagy (2018) ilustran cómo se puede utilizar FHIR para ofrecer al servicio de radiología una mejor integración clínica y un sistema centrado en el usuario.

Asimismo, en América Latina también se han realizado varios esfuerzos, como el caso de Argentina, López et al. (2020) presenta como FHIR se está utilizando para desarrollar e implementar una plataforma web de telemedicina para consultas electrónicas.

Sin embargo, estas constantes investigaciones en torno a la salud, se han enfocado en solucionar el problema de la integración de datos de los diferentes sistemas de información médica, centrándose en soluciones que requieren la creación de la estructura de datos basada en el estándar FHIR, y no ha buscado la forma de integrar información existente en formatos propios (sin estándar) para cada región, país e incluso cada prestador de servicios médicos.

Es común encontrar en América Latina este tipo de problemas donde la adaptación a estos estándares internacionales es relativamente nueva, tal es el caso de Ecuador, donde los esfuerzos para lograr la interoperabilidad de los sistemas de información médica ha sido un problema que puede tener más de veinte años y que no ha logrado encontrar una solución adecuada.

Kiourtis et al. (Kiourtis et al., 2019) proponen resolver este problema mapeando datos de la industria médica a FHIR a través de una alineación de ontologías. Además, los autores se enfocan en un mecanismo que puede aprovechar al máximo el entorno de uso intensivo de datos sin perder la complejidad de la salud. No obstante, estos resultados no son del todo precisos debido a que se propone el procesamiento automático y sin supervisión humana, por lo que, los autores lo recomiendan para escenarios de flujo de datos en tiempo real ya que es mejor y más eficiente en comparación con otros. Finalmente, Mandirola et al. [11] concluye la siguiente declaración:

La falta de políticas sobre estándares aumenta los costos y los errores en los procesos de interoperabilidad del sistema. Por tanto, antes de definir los sistemas a utilizar, es necesario definir los marcos y políticas que permitirán la interoperabilidad de los sistemas que se implementen. Esto incluye estándares de mensajería, terminología, identificadores, tablas maestras y motores de integración.

Además, en países como Ecuador es común encontrar que la información de las bases de datos ha sido anonimizada y en algunos casos específicos incluso se ha codificado, o se ha encriptado de alguna forma la estructura de la base de datos. La necesidad de una conversión previa es evidente, dado que es imposible utilizar alineamiento de ontologías sobre datos que implica un trabajo complejo para encontrar la relación semántica.

Por ello, la propuesta de este trabajo es desarrollar un marco en el que los datos son transformados a través de los expertos de cada uno de los sistemas, donde además se definen las políticas de los mismos, para establecer la máxima precisión de la transformación y permita utilizar para intercambiar datos entre distintas organizaciones médicas. Esto además permitirá una alineación de ontologías mucho más adecuada para integrar investigaciones posteriores.

1.4. Arquitectura de emparejamiento propuesta

La arquitectura propuesta se encuentra en la figura 1. El sistema consta de dos partes principales: la primera es la que se aloja en un servidor y se encarga de manejar los datos y las conversiones, y la segunda que es el extractor y proveedor de datos.

En la parte del servidor se tienen dos controladores uno HL7 y otro FHIR ambos se encargan de la transformación e interpretación de los diferentes datos almacenados en las bases de datos de las aplicaciones externas. Los datos se intercambian a través de endpoints REST y XML.

Los datos clínicos para la prueba y validación del software se trabajaron en coordinación con la empresa INNTRATEC S.A.S pues la solución se propone para uso en dicha empresa.

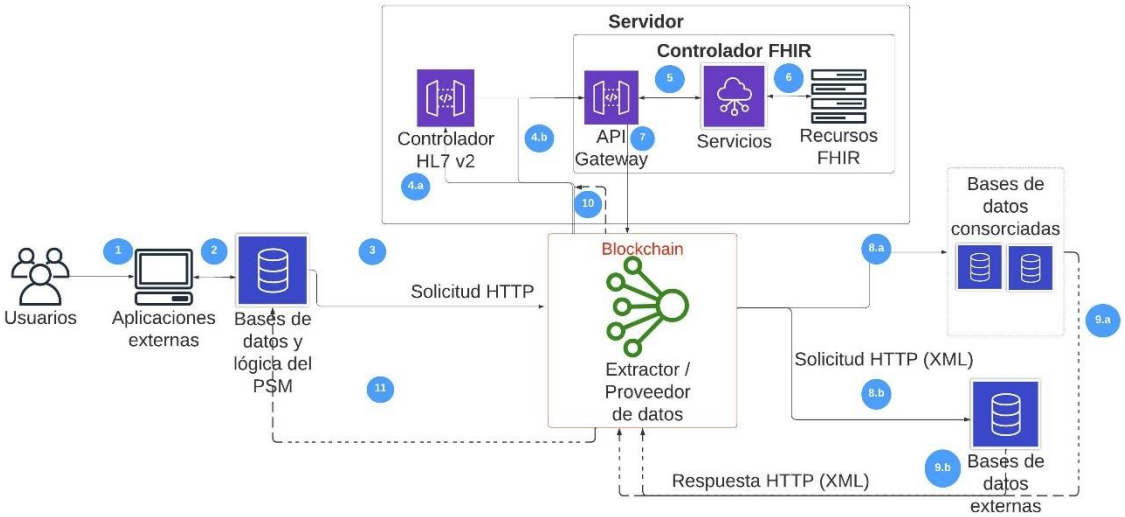


Figura 1.1: Arquitectura del Sistema. Fuente: Elaboración propia.

Cabe recalcar que este trabajo de titulación dio como resultado un artículo científico que fue parte del International Conference on Smart Technologies, Systems and Applications SmartTech-IC 2021 y que fue publicado como parte del libro: Smart Technologies, Systems and Applications en Springer Link (Maxi & Morocho, 2022).

1.5. Contexto del proyecto

Este trabajo de titulación se ha desarrollado en el contexto del proyecto titulado: "*Software integrador de información médica utilizando Health Level Seven y FHIR*" planteado por la empresa INNTRATEC SAS.

1.6. Hipótesis y Objetivos

En esta sección, se definen las hipótesis y los objetivos del presente trabajo de titulación.

1.6.1. Hipótesis

H0: Un middleware informático basado en tecnologías web y en estándares internacionales HL7 no permite la interoperabilidad correcta de diferentes sistemas de información médica.

H1: Un middleware informático basado en tecnologías web y en estándares internacionales HL7 permite la interoperabilidad correcta de diferentes sistemas de información médica.

1.6.2. Objetivo General

Diseñar e implementar un middleware de información médica usando estándares HL7 y FHIR, que permita la interoperabilidad de sistemas de información médica.

1.6.3. Objetivos Específicos

1. Analizar los requerimientos del software para la correcta construcción del software.
2. Definir la arquitectura FHIR, HL7 y diagramar el software propuesto.
3. Implementar el software usando una metodología ágil.

1.7. Estructura del trabajo

El presente trabajo de titulación y sus contenidos se estructuran de la siguiente forma:

Capítulo 1 – Introducción: se presenta la problemática y los objetivos del presente trabajo de titulación.

Capítulo 2 – Metodología para la conversión: se describe la metodología que se seguirá para lograr la construcción del middleware propuesto.

Capítulo 3 – Marco Conceptual: se definen ciertos términos necesarios para la completa comprensión del sistema desarrollado, así como su importancia. Este capítulo se centra mucho en definir lo establecido por FHIR.

Capítulo 4 – Metodología: se describe el análisis y diseño de software, la arquitectura del middleware, las variables que se tendrán a consideración para alcanzar la interoperabilidad, es decir, los datos que se usarán de los formularios médicos ecuatorianos; y finalmente la implementación y construcción de los servicios para FHIR y HL7.

Capítulo 5 – Análisis y presentación de resultados: presenta los resultados obtenidos al construir el sistema y aplicar un diseño experimental donde se realiza un cuasi-experimento para validar, responder, y aceptar/rechazar la hipótesis planteada en la introducción y las hipótesis planteadas en este mismo diseño.

Capítulo 6 – Conclusiones y trabajo futuro: se describen las conclusiones obtenidas al realizar este trabajo de titulación, así como, las recomendaciones para trabajos futuros.

CAPÍTULO 2: MARCO CONCEPTUAL

En este apartado se resumen las ideas obtenidas para lograr los objetivos planteados en el capítulo 1. Además, se definen ciertos conceptos que se irán utilizando a lo largo del documento, para definir la importancia y mejorar el entendimiento del software elaborado.

El middleware elaborado está enfocado a intercambiar información médica entre sistemas por lo que, es de vital importancia definir varios términos que son muy comunes en este entorno.

2.1. Historial Clínico Electrónico

Uno de los puntos centrales de esta tesis, se encuentra en el historial clínico electrónico, dado que, es la principal fuente de información de los pacientes en el sistema de salud ecuatoriano.

Según la literatura, el significado de Historial Clínico Electrónico (EHR, por sus siglas en inglés) es inestable

Un EHR se utiliza principalmente con el propósito de establecer objetivos y planificar la atención al paciente, también ayuda a documentar la prestación de la atención y evaluar los resultados de la atención. Incluye información sobre las necesidades del paciente durante los episodios de atención brindados por diferentes profesionales de la salud (Häyrinen et al., 2008).

Los EHR se clasificaron sobre la base de la definición de la Organización Internacional de Normalización (ISO, 2010). Según esta definición, EHR significa un depósito de datos de pacientes en forma digital, almacenados e intercambiados de forma segura y accesible por múltiples usuarios autorizados. Sin embargo, la estructura y el contenido de las EHR ha variado a lo largo del tiempo. (Häyrinen et al., 2008).

2.2. Sistemas de información médica

Los historiales clínicos electrónicos o EHR idealmente son manejados por sistemas de información médica. Con el creciente énfasis en proporcionar la información correcta a la persona adecuada en cualquier lugar y en cualquier momento en el mundo

interconectado globalmente de hoy, la industria de la salud del mundo se ha estado moviendo hacia sistemas de registro de salud electrónico (EHR). (Thakkar & Davis, 2006)

El estado general de los sistemas de EHR y los efectos de su uso son temas de creciente interés para los investigadores. (Thakkar & Davis, 2006)

Los sistemas de salud deben ser capaces de medir e informar lo que más les importa a las personas, como la atención competente, la experiencia del usuario, los resultados de salud y la confianza en el sistema (Kruk et. al*., 2018*).

Sin embargo, en países como Ecuador, históricamente se han generado sistemas que no necesariamente han seguido estándar alguno. Estos están incomunicados entre sí por la baja interoperabilidad existente.

2.2.1. Límites de un Sistema de información Médica

Se han propuesto muchos límites para separar el sistema de salud de los elementos externos. Algunos componentes como, los servicios de salud individuales prestados en las clínicas, están incluidos en todas las definiciones de estos límites. Pero hay más controversia con otros componentes. (Murray & Frenk, 2000)

Tal definición, ha sido construida comenzando con el concepto de "acción de salud". Una acción de salud se define como cualquier conjunto de actividades cuya intención principal es mejorar o mantener la salud. Y un sistema de salud incluye los recursos, actores e instituciones relacionados con el financiamiento, regulación y provisión de acciones de salud. Este criterio de intención principal conduce a una definición amplia del sistema de salud (Murray & Frenk, 2000).

2.3. Interoperabilidad

En este punto, es importante hablar acerca de la interoperabilidad que, según la RAE, se refiere a la habilidad de dos o más sistemas, o de sus componentes, para utilizarse de forma conjunta e intercambiable (Real Academia Española, s.f., definición 1).

La interoperabilidad es la única manera sostenible de permitir que las entidades que actúan en varios lugares, con diferentes conocimientos y perspectivas, y que utilizan

distintos sistemas de información de diferentes proveedores, colaboren y brinden atención médica de calidad (Kiourtis et al., 2019).

2.4. El reto de la interoperabilidad

En consecuencia, uno de los más grandes retos en este campo, es establecer esta interoperabilidad en los sistemas de información médica y por lo que, es inevitable realizarse la siguiente pregunta: ¿Por qué la interoperabilidad es difícil? Por este motivo, se definen ciertos aspectos que permitirán responder la interrogante en cuestión.

2.4.1. Capas de la interoperabilidad

Muchos grandes proyectos de TI de salud no logran sus objetivos, especialmente cuando se trata de interoperabilidad. Los beneficios de la atención médica conjunta dependen de una interoperabilidad segura y confiable para proporcionar la información correcta cuando y donde se necesite. Según Benson & Grieve (2021) se puede pensar en la interoperabilidad como si tuviera cuatro capas: institucional, datos, humana y tecnológica.

En la interoperabilidad del cuidado de la salud, cada una de estas cuatro capas es importante. Por eso, los gobiernos y los proveedores deben trabajar juntos para lograr buenos resultados, especialmente a nivel institucional, donde las barreras a la interoperabilidad se ven seriamente exacerbadas por la falta de incentivos apropiados, el bloqueo tecnológico y las preocupaciones por la privacidad.

El arte consiste en permitir la diversidad al tiempo que se garantiza que los sistemas funcionen juntos de la forma que más importa (Benson & Grieve, 2021).

2.4.2. ¿Por qué los estándares son necesarios?

Parte del problema con los estándares no es que haya tantos para elegir, sino que no se ha logrado incentivar adecuadamente el uso de los que se tiene. A menudo, el problema es que no existe un regulador con el poder de hacer que el despliegue se lleve a cabo de forma ordenada. Esto es contraproducente ya que los estándares que no se implementan son una pérdida de tiempo y esfuerzo (Benson & Grieve, 2021). Una visión alternativa es que los estándares disponibles han sido demasiado complejos y costosos de implementar y mantener. Esta visión ha llevado al desarrollo de FHIR (Fast

Health Interoperability Resources) que busca evitar este problema en específico (Benson & Grieve, 2021).

Es importante mencionar que, el volumen de transacciones en el sector de la salud es asombroso. Por ejemplo, en 2007, un solo sistema EHR en un gran hospital (la Clínica Mayo en Rochester, Minnesota) procesó más de 660 millones de mensajes HL7 v2 al año, alrededor de dos millones de mensajes al día (Benson & Grieve, 2021).

2.4.3. Complejidad

El número de enlaces necesarios para conectar *n* sistemas diferentes aumenta según la fórmula:

$$Número\ de\ enlaces = \frac{n(n-1)}{2} = \binom{n}{2}$$

La vinculación de dos nodos requiere una única interfaz, que puede ser ejemplificada fácilmente por un par de personas sentadas alrededor de una mesa. La vinculación de 6 nodos requiere 15 interfaces; vincular 100 nodos requiere 4950 interfaces. Esta es una explosión combinatoria. Sin embargo, en el caso de los 6 nodos, con el uso de un solo estándar se necesitarían únicamente 6 interfaces para interconectarse entre sí, como se puede observar en la figura 2.1. (Benson & Grieve, 2021).

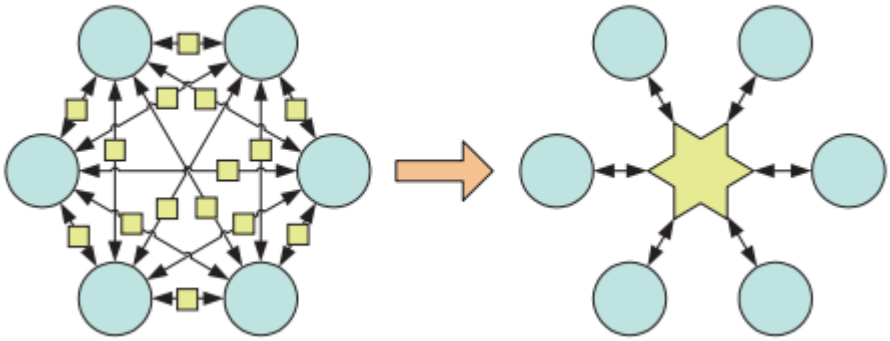


Figura 2.1. Por Benson and Grieve, 2021. Los beneficios de un estándar. Fuente: Principles of Health Interoperability

2.4.4. Traducciones

Cada tipo de sistema informático almacena datos internamente de una manera diferente. Esto significa que, para comunicarse, primero es necesario una correspondencia de las estructuras de base de datos y posteriormente, los datos deben traducirse de un idioma o formato interno a otro. (Benson & Grieve, 2021).

La conexión implica la traducción a un formato estándar que sea entendido por cada parte. Esto requiere dos traducciones, primero del idioma nativo del remitente al formato y luego del formato al idioma nativo del destinatario. Los estándares HL7, incluido FHIR, proporcionan una lengua franca común para hacer precisamente esto. (Benson & Grieve, 2021).

2.5. Health Level Seven (HL7)

Dentro de los esfuerzos que se han hecho por parte de la industria para habilitar la interoperabilidad de sistemas de información médica, se encuentra Health Level 7 (HL7) que es una organización de establecimiento de estándares acreditada por el Instituto Nacional Estadounidense de Estándares (ANSI, por sus siglas en inglés). HL7 ha desarrollado protocolos de comunicación ampliamente utilizados en los Estados Unidos, con un creciente reconocimiento e implementaciones internacionales. HL7 es una organización respaldada por proveedores, su misión es proporcionar estándares para el intercambio, la gestión y la integración de datos que respaldan la atención clínica del paciente y la gestión, prestación y evaluación de los servicios de atención médica (Dolin et al., 2015)

Esto abarca el ciclo de vida completo de una especificación de estándares: desarrollo, adopción, reconocimiento de mercado, utilización y cumplimiento. Las especificaciones HL7 están unificadas por modelos de referencia compartidos de los dominios técnico y sanitario. El estándar de mensajería HL7 actualmente se encuentra en la versión 3 (Dolin et al., 2015).

Mucha gente conoce HL7 como una organización que crea estándares de mensajería para el cuidado de la salud. Health Level 7 también ha desarrollado estándares para la representación de documentos clínicos (como resúmenes de alta y notas de evolución). Estos estándares de documentos conforman la Arquitectura de documentos clínicos (CDA, por sus siglas en inglés) de HL7. La arquitectura de

documentos clínicos HL7, versión 1.0, se convirtió en un estándar HL7 aprobado por ANSI en noviembre de 2000 (Dolin et al., 2015).

2.5.1. HL7 versión 3

La versión 3 de HL7 utiliza una metodología de desarrollo orientada a objetos y un modelo de información de referencia (RIM, por sus siglas en inglés) para crear mensajes. RIM es un diagrama de estilo UML[4] basado en seis tipos básicos de clases y reglas que gobiernan cómo se relacionan entre sí. Estas reglas, así como otras restricciones sobre los atributos permitidos para cada clase, conforman el modelo de información de la versión 3 de HL7. Las restricciones de cardinalidad y opcionalidad también existen en las relaciones entre clases y en los atributos (Ryan, 2006).

Las seis clases principales de RIM son:

- *Act* - una acción de interés.
- *Entity*: una clase o instancia de una cosa específica capaz de participar en *Acts*.
- *Role*: una *Entity*, en un *Role* particular, puede participar en un *Act*.
- *Participation*: una asociación entre un *Role* y un *Act*.
- *ActRelationship*: una asociación entre un par de *Acts*.
- *RoleLink*: una conexión entre dos “Roles” que expresan una dependencia (Ryan, 2006)

Tres de estas clases, *Act, Entity* y *Role*, están representadas además por un conjunto de clases especializadas o subtipos. P.ej. Las especializaciones de la clase de Acto incluyen: Observación, Procedimiento y Administración de sustancias.

Como puede verse por la naturaleza de las clases de RIM, el modelo de información HL7 adopta una visión centrada en el acto (*Act*), con procesos e información en el cuidado de la salud representados principalmente en términos de los actos (*Acts*) (Ryan, 2006).

[4] Unified Modeling Language – http://www.uml.org

2.6. Fast Healthcare Interoperability Resources

Como una evolución natural a HL7, tenemos a HL7-FHIR Fast Healthcare Interoperability Resources (FHIR) que es un marco estándar creado por HL7. FHIR define un conjunto de recursos de datos de salud y una API REST en la parte superior para acceder a ellos. La especificación FHIR está disponible abiertamente bajo la licencia *creative commons*. Su contenido está disponible gratuitamente para implementadores, arquitectos y médicos. Los principales bloques de construcción de FHIR son los llamados Recursos o "Resources". Los recursos son unidades semánticas discretas modulares y extensibles. Estos recursos representan conceptos clínicos como, Paciente, Diagnóstico o Medicamento, entre otros. Estos recursos se instancian en XML, JSON o RDF (Khvastova et al., 2020). Se espera que una instancia de recurso tenga una URL de identificación, metadatos, un resumen XHTML legible por humanos y los datos reales que se capturan. Cada recurso define qué elementos de datos específicos forman parte de su estructura, sus nombres, cardinalidades y restricciones. Todos los recursos deben tener una representación legible por humanos. No obstante, FHIR se basa en gran medida en información codificada (Khvastova et al., 2020).

Además, FHIR admite la vinculación terminológica de sistemas de códigos y conjuntos de valores. Los sistemas de código permiten el uso de terminologías externas como LOINC[5] o SNOMED CT[6], las cuáles sirven para codificar enfermedades por ejemplo. (Khvastova et al., 2020)

Dado que, los sistemas de información clínica tienen requisitos variables, la especificación **FHIR deja espacio para la flexibilidad**. Eso significa que, los recursos se pueden adaptar a casos de uso más específicos. Esto se puede implementar estableciendo restricciones a los elementos de datos y haciendo uso del marco de extensión o *extensions* de FHIR (Khvastova et al., 2020).

2.6.1. Proceso de desarrollo FHIR

Mientras que, la mensajería HL7 v2 siguió un proceso de desarrollo ad hoc y la v3 siguió un proceso impulsado por un modelo descendente definido rígidamente

[5] https://loinc.org/
[6] https://www.snomed.org/

denominado HL7 Development Framework (HDF), FHIR utiliza un enfoque incremental e iterativo para el desarrollo del estándar. El desarrollo incremental se refiere al desarrollo de piezas pequeñas a la vez. Las piezas incrementales se implementan y prueban, y el aprendizaje se retroalimenta al proceso de diseño muy rápidamente. Esto se hace una y otra vez (Bender & Sartipi, 2013).

2.6.2. Artefactos FHIR

FHIR tiene como objetivo definir las entidades claves involucradas en el intercambio de información sanitaria como recursos. Cada recurso es una entidad identificable distinta. La especificación FHIR según Bender & Sartipi en 2013 describe los siguientes atributos de recursos:

- Los recursos deben tener un límite claro, que coincida con uno o más alcances de transacciones lógicas.
- Los recursos deben diferir entre sí en el significado, no solo en el uso (por ejemplo, las diferentes formas de usar un informe de laboratorio deben no dar lugar a recursos diferentes).
- Los recursos deben tener una identidad natural.
- Los recursos deben ser muy comunes y usarse en muchas transacciones comerciales diferentes.
- Los recursos no deben ser lo suficientemente específicos o detallados como para excluir el soporte para una amplia gama de transacciones comerciales.
- Los recursos deben ser mutuamente excluyentes.
- Los recursos deben usar otros recursos, pero deben ser más que simples composiciones de otros recursos; cada recurso debe presentar un contenido novedoso.
- Los recursos deben organizarse en un marco lógico basado en los elementos comunes del recurso y con lo que se vincula.
- Los recursos deben ser lo suficientemente grandes para proporcionar un contexto significativo; Los recursos que contienen solo unos pocos atributos probablemente sean demasiado pequeños para proporcionar un valor comercial significativo.

Al momento de escribir esta tesis se pueden encontrar un total de 145 recursos (Bender & Sartipi, 2013).

2.6.3. Beneficios de FHIR

La aplicación del estándar FHIR incluye ciertos beneficios, entre estos se incluyen:

- **Fuerte enfoque en la implementación:** rápido y fácil de implementar (muchos desarrolladores han tenido interfaces simples trabajando en un solo día).
- **Varias bibliotecas de implementación**, con muchos ejemplos para impulsar el desarrollo.
- La especificación es de **uso gratuito** sin restricciones.
- **Interoperabilidad lista para usar:** los recursos básicos se pueden usar tal cual, pero también se pueden adaptar según sea necesario para cumplir con los requisitos locales utilizando Perfiles, Extensiones y Terminologías.
- **Ruta de desarrollo evolutiva desde HL7 Versión 2 y CDA:** los estándares pueden coexistir y aprovecharse entre sí.
- **Base sólida en estándares web:** XML, JSON, HTTP, OAuth, etc.
- **Soporte para arquitecturas RESTful**, intercambio fluido de información usando mensajes o documentos y arquitecturas basadas en servicios.
- **Especificaciones concisas** y de fácil comprensión.
- Un **formato de serialización legible** por humanos para facilitar su uso por parte de los desarrolladores.
- **Análisis basado en ontología con mapeo formal de corrección** (Benson & Grieve, 2021).

2.6.4. API's

Una parte importante de FHIR es la especificación de las API's. Para describir de mejor manera este concepto es importante recordar que las especificaciones de interoperabilidad se pueden agrupar en varios enfoques diferentes:

- **Mensajes:** define un conjunto de información fija que se puede intercambiar entre aplicaciones cuando ocurre un evento específico.
- **Servicios:** define un conjunto de operaciones funcionales que un sistema expone para que otros lo utilicen, junto con las expectativas de comportamiento en torno a ellas.
- **Documentos:** define un conjunto de paquetes de información fijos que se pueden intercambiar o almacenar para su uso posterior (Benson & Grieve, 2021)

Dicho esto, la interoperabilidad operativa requiere que los sistemas acuerden qué se intercambia, cuándo se intercambia y por qué. Por lo tanto, todos estos enfoques deben terminar en el mismo lugar. La diferencia entre ellos depende de qué partes de la solución general están estandarizadas y cuáles se dejan a la discreción de los implementadores.

Por ejemplo, un sistema de mensajería típicamente vinculará las descripciones de la información a un conjunto particular de tecnologías, por lo que, los mensajes se pueden intercambiar, pero deja que los implementadores del sistema decidan qué tipo de servicio ofrecer (Benson & Grieve, 2021).

Sin embargo, todo el tiempo, ha habido otra forma de interoperabilidad, llamada Interfaz de programación de aplicaciones (API, por sus siglas en inglés). Una API es un conjunto de servicios ofrecidos por repositorio que puede ser utilizado por otro programa para lograr sus propios objetivos. Estos son los servicios de interoperabilidad sobre los que se basan los sistemas operativos.

Históricamente, las API estaban limitadas por la tecnología a proporcionar servicios dentro de un solo proceso en una computadora, pero luego esto se extendió para permitir la invocación de operaciones a través de internet.

2.6.5. Interfaces RESTful

FHIR es una especificación RESTful; FHIR aspira a seguir los principios RESTful tanto como sea posible. Este es un reconocimiento de la influencia que las interfaces RESTful han tenido en la industria. Han demostrado que, es posible construir grandes ecosistemas basados en la integración rápidamente, además que los desarrolladores

pueden integrarse fácilmente con los servicios RESTful y que estos servicios escalan muy bien (Benson & Grieve, 2021).

Los principios básicos de REST son:

1. **Interfaz uniforme:** los recursos individuales se identifican mediante URL y se pueden representar de varias formas diferentes (por ejemplo, XML, JSON). Los clientes manipulan el recurso a través de las representaciones utilizando mensajes autodescriptivos. Los hipermedias (hipervínculos) y el hipertexto actúan como motor para la transferencia de estado.
2. **Interacciones sin estado**: Ninguno de los contextos del cliente se almacena en el lado del servidor entre las solicitudes, por lo que, toda la información necesaria para atender la solicitud se contiene en la URL, los encabezados o el cuerpo.
3. **Se pueden almacenar en caché:** las respuestas se pueden almacenar en caché y las respuestas deben definirse como almacenables en caché o no.
4. **El cliente y el servidor** están separados entre sí, por lo que, el cliente no se preocupa por el almacenamiento de datos y el servidor no se preocupa de la interfaz de usuario.
5. **Sistema en capas:** en cualquier momento, un cliente no puede saber si está conectado al servidor final o a un intermediario. Los intermediarios pueden ayudar a hacer cumplir las políticas de seguridad, habilitar el equilibrio de carga, etc. (Benson & Grieve, 2021).

FHIR es una interfaz RESTful, sin embargo, la especificación no adopta REST como religión. En cambio, la religión de FHIR es hacer lo que funciona en la práctica y preferiblemente lo que hacen la mayoría de las otras interfaces RESTful (Benson & Grieve, 2021).

Hay dos diferencias clave entre FHIR y el tipo de interfaces RESTful que ofrecen los proveedores de cloud típicos:

1. La mayoría de los servicios RESTful basados en la nube son proporcionados por un solo proveedor para cumplir con un propósito comercial específico en un solo servidor.

   Por otro lado, FHIR es una especificación general para el intercambio de datos entre múltiples partes. Una consecuencia de esto es que, la especificación FHIR es más amplia y flexible.

2. La información sobre el cuidado de la salud nunca se limitará al intercambio a través de las API RESTful; hay muchas razones para utilizar otros tipos de intercambio, y la especificación FHIR se extiende para atender también los enfoques de mensajería y documentos (Benson & Grieve, 2021).

2.6.6. Repositorio

Como especificación RESTful, FHIR se organiza en torno al concepto de repositorio, que es una lista de recursos de un tipo particular. Esto se representa explícitamente en las URL definidas por FHIR (Benson & Grieve, 2021). Esta URL sigue la siguiente estructura:

[Base-address]/[Type]/[id]

En las tablas 1 y 2 se presenta las descripciones de la estructura básica de una URL en FHIR y de los servicios base que incluye este estándar, respectivamente.

Tabla 1. Estructura de la URL FHIR

Parte	**Descripción**	**Ejemplo**
Base-adress	Identifica un servicio del sistema FHIR. Es decir, un servidor que hace que la información esté disponible de conformidad con la especificación FHIR.	http://server.example.com/fhir
Type	Identifica un servicio de tipo FHIR que administra una colección de recursos del mismo tipo. El tipo debe ser uno de los definidos en la especificación FHIR	Patient
Id	Identifica un servicio de instancia FHIR que administra una instancia de un recurso dentro de la colección	23455

Tabla 2. Servicios base de FHIR

Servicio	**Descripción**

Servicio de instancia	Permite a un cliente recuperar el contenido actual de un recurso, actualizar el contenido a un nuevo estado, eliminar el recurso o ver su historial de modificaciones.
Servicio de tipo	Permite a un cliente buscar en los recursos existentes, crear una nueva instancia de un recurso u obtener el historial de todos los cambios en el recurso del tipo en específico.
Servicio de sistema	Permite a un cliente determinar qué funcionalidad proporciona el sistema, realizar lotes y transacciones en múltiples tipos de recursos y obtener el historial de todos los cambios en todos los recursos.

Los servidores eligen qué funciones poner a disposición, para qué tipos, en función de los casos de uso que admiten. Los servidores deben devolver una declaración de conformidad que permita al cliente determinar qué funcionalidad proporciona. Un servidor que devuelve una declaración de conformidad que dice que no se proporciona ninguna otra funcionalidad es totalmente conforme (pero no necesariamente útil) (Benson & Grieve, 2021).

La API FHIR proporciona un enfoque centrado en los registros para el intercambio de datos. En lugar de pedirle al servidor que realice alguna operación, el cliente le dice al servidor cuál debe ser el contenido del registro. Estos servicios a menudo se denominan servicios CRUD (por sus siglas en inglés, Crear, Leer, Actualizar y Eliminar). Se debe tener en cuenta que en los sistemas sanitarios muy pocos registros se eliminan realmente. (Benson & Grieve, 2021).

2.6.7. Recursos

La mayor parte del contenido significativo de la especificación FHIR está en la definición de recursos. FHIR define 145 tipos de recursos (en R4; este número crece lentamente entre versiones), que representan tipos de contenido muy diferentes (Benson & Grieve, 2021). En este proyecto no se hará uso del total de recursos expuestos por FHIR.

2.7. CIE10

El CIE10 corresponde a la versión en español del ICD10 (International Classification of Diseases), cuyas siglas significan: Clasificación Internacional de Enfermedades. Esta clasificación fue elaborada por el Centro Nacional de Estadísticas de Salud (NCHS), del Gobierno Federal de EEUU. En el año 2011 inicia la traducción de esta clasificación en conjunto con 17 sociedades médicas en EEUU. (Sanidad & Bienestar Social, 2020).

El formato que sigue CIE10 está compuesto de dos partes, el índice alfabético y la lista tabular. En el índice alfabético se organiza a los términos alfabéticamente y se les asigna un código. Por otro lado, la lista tabular, los elementos son códigos alfanuméricos divididos en capítulos basados en sistemas corporales o entidades nosológicas (Sanidad & Bienestar Social, 2020). Por ejemplo, un absceso con sangrado estaría representado por el código K57.81.

2.8. Expediente único para la Historia Clínica Ecuatoriana

En el año 2003 la Dirección Nacional de Planificación y Financiamiento del Ministerio de Salud Pública de Ecuador propone el rediseño de los formularios de la Historia Clínica. De este modo, en 2004 el Ministerio de Salud Pública entrega de manera formal la propuesta de modificación ante el Directorio del Consejo Nacional de Salud. Luego de varios talleres, pilotajes y validaciones en 2006 se llegan a diferentes acuerdos que establecen 14 formularios básicos:

- Consulta externa
- Anamnesis y Examen Físico
- Signos vitales
- Evolución y prescripciones
- Epicrisis
- Interconsulta
- Emergencia
- Laboratorio Clínico

- Referencia
- Imagenología
- Histopatología
- Odontología
- Administración de medicamentos
- Trabajo social

Estos formularios son elevados para el conocimiento y para ser usadas por todas las instituciones del Sistema Nacional de Salud del Ecuador. (Ministerio de Salud Pública del Ecuador, 2006)

CAPÍTULO 3: METODOLOGÍA

Para la elaboración de este proyecto de tesis se siguió una metodología de programación iterativo a lo largo del enfoque incremental. Esta incluye ciertos conceptos concernientes al desarrollo de sistemas con estándares FHIR.

En este capítulo se describe cada una de las acciones realizadas para desarrollar el middleware propuesto.

3.1. Procedimiento para la conversión

Para esto se debe tener en cuenta el proceso requerido Business Process Model and Notation (BPMN), así como, su Conjunto Mínimo de Datos (CMD) que contiene la información mínima para establecer la interoperación en los sistemas médicos que se estudian.

La razón por la que, se decidió usar esta metodología sobre otras, es que mantiene su centro en el estándar FHIR así como con el FHIR development process (HL7.org, 2021) y mencionada en (Sharma & Aggarwal, 2019), que se establece en la página oficial de descripción del estándar.

Según esta metodología, es necesario:

1. **Establecer el tipo formato de intercambio**. Entre los posibles se encuentran: XML, JSON y RDF
2. **Establecer las terminologías**, ya sea Code Systems y Value Sets, a usar. En FHIR las descripciones de ciertos atributos, como las enfermedades o roles familiares, entre otros, debe estar descritos en terminologías, las cuales deben ser descritas en sitio web público.
3. **Establecer los mapeos que se usarán**, es decir, los mapeos necesarios para la conversión a otros estándares como por ejemplo HL7 V2.
4. **Establecer el RESTful API sobre HTTP**. Dado que, FHIR es un estándar, se basa en la estandarización de las estructuras de recursos y las interfaces. Esto puede considerarse una violación de los principios REST, pero es clave para garantizar una interoperabilidad consistente entre diversos sistemas. Cada "tipo de recurso" tiene definido el mismo conjunto de interacciones que se pueden utilizar para administrar los recursos de una manera muy granular (HL7, 2021d).

5. **Recopilar todos los recursos FHIR** necesarios para habilitar la interoperación.
6. **Establecer las URLs para la búsqueda.** La búsqueda de recursos es fundamental para la mecánica de FHIR. Las operaciones de búsqueda atraviesan un conjunto existente de recursos filtrados por parámetros proporcionados a la operación de búsqueda (HL7, 2021e).
7. **Establecer las URLs para las operaciones.** La API RESTful define un conjunto de interacciones comunes (leer, actualizar buscar, etc.) realizadas en un repositorio de recursos tipificados (HL7, 2021b).
8. **Establecer el uso asíncrono.** El patrón de solicitud asincrónica, basado en RFC 7240, se adapta a este caso de uso y es aplicable para todas las interacciones definidas y para las operaciones, aunque para muchos de estos usos no aporta ningún beneficio. Los servidores pueden elegir en qué interacciones se debe admitir el patrón (si es que lo hacen), y los servidores pueden optar por admitir solo algunas de las operaciones que utilizan el patrón asincrónico (HL7, 2021a).
9. **Diseñar las comunicaciones con el resto de sistemas**

   En este paso se diseñan las comunicaciones con los sistemas con los que se conectará el *framework* para la extracción y envío de datos.
10. **Codificación y pruebas**

    Finalmente, todos los esquemas y datos obtenidos en los pasos anteriores están codificados.
11. **Establecer el profiling**. La especificación FHIR describe un conjunto de recursos básicos, marcos y API que se utilizan en muchos contextos diferentes en el cuidado de la salud. Sin embargo, existe una amplia variabilidad entre las jurisdicciones y en todo el ecosistema de atención médica en torno a las prácticas, los requisitos, las regulaciones, la educación y las acciones que son factibles y / o beneficiosas. Por esta razón, la especificación FHIR es una "especificación de plataforma": crea una plataforma o base común sobre la cual se implementan una variedad de soluciones diferentes (HL7, 2021c).

    En la figura 3.1 se puede observar el flujo del procedimiento para la conversión.

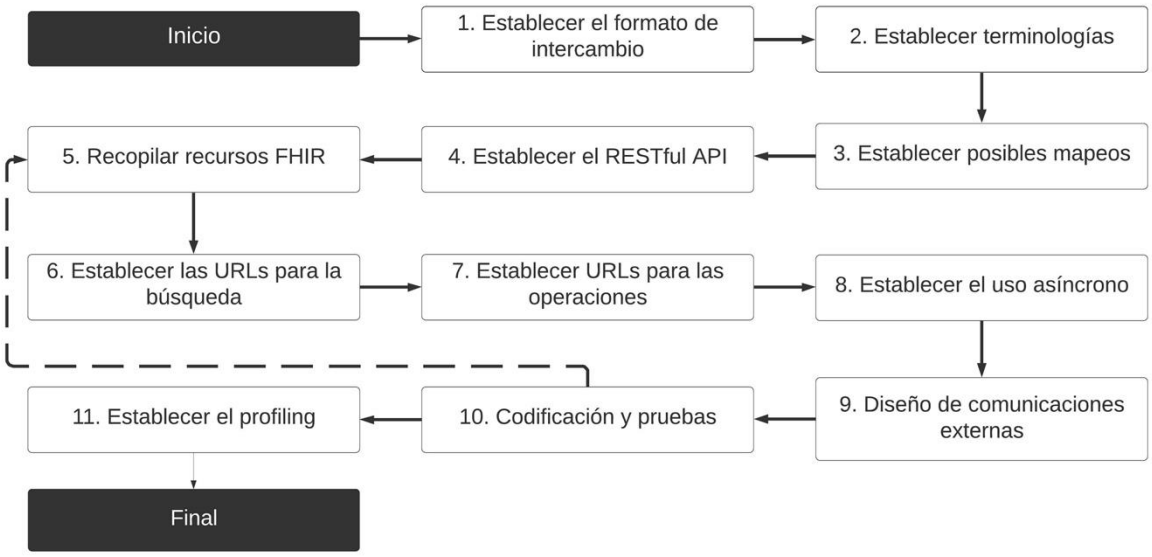


Figura 3.1. Metodología aplicada. Fuente: Elaboración propia

3.2. Requerimientos y diagramas del software

Para un mejor entendimiento de lo desarrollado, se dividió el middleware en varios diagramas UML para representar el funcionamiento interno y externo del mismo. Además, se presenta la arquitectura general del sistema.

Por otro lado, una aplicación efectiva y eficiente debe cumplir con los requerimientos de los usuarios que harán uso de ella, por lo que, en esta sección también se detalla los requerimientos obtenidos al aplicar ingeniería de requisitos con los potenciales usuarios del middleware.

3.2.1. Obtención y análisis de requisitos

Para obtener los requerimientos del middleware se realizó el siguiente estudio técnico:

3.2.1.1. Captura de requisitos

Se realizaron tres entrevistas con tres tipos de usuarios del centro de especialidades médicas de la Universidad de Cuenca: personal de medicina, personal técnico del sistema de información médica y al administrador del proveedor de servicios

médicos. Esto con el objetivo de iniciar el funcionamiento del sistema desarrollado. A continuación, se detallan dichas entrevistas

Entrevista: Personal médico
1. Puesto que ocupa: 2. ¿En qué considera que el PSM donde labora se especializa? 3. ¿Cómo considera que se podría mejorar la admisión de nuevos pacientes al PSM? 4. ¿Qué datos se requieren para poder atender a un paciente nuevo en el PSM? 5. ¿Existen datos adicionales que podrían mejorar el seguimiento de un paciente? Si es así ¿Cuáles serían estos?

Entrevista: Personal técnico del sistema
1. Puesto que ocupa: 2. ¿En qué considera que el PSM donde labora se especializa? 3. ¿Cómo considera que se podría mejorar la admisión de nuevos pacientes al PSM? 4. ¿Qué datos se requieren para poder atender a un paciente nuevo en el PSM? 5. ¿Existen datos adicionales que podrían mejorar el seguimiento de un paciente? Si es así ¿Cuáles serían estos? 6. ¿Qué datos recopila actualmente de los pacientes dentro de su sistema? 7. ¿Cómo cree que podría realizarse un intercambio de datos con otros sistemas de información médica? 8. ¿En qué tecnologías está construido el sistema? 9. ¿Qué tipo de metaformatos usa para la comunicación (XML, JSON, RDF, otros)? 10. ¿En su experiencia, cuál considera que sería el mejor metaformato para este dominio? 11. ¿En su experiencia, cuál considera que sería el metaformato más fácil de entender? 12. ¿En su experiencia, cuál considera que sería el metaformato más usado por los otros sistemas de información médica? 13. ¿Cómo se maneja el acceso a los datos del sistema?

14. ¿Cómo se incluye a un nuevo usuario dentro del sistema? 15. ¿Los usuarios pueden acceder a su historial médico a través del sistema?

Entrevista: Administrador del PSM
1. Puesto que ocupa: 2. ¿En qué considera que el PSM donde labora se especializa? 3. ¿Cómo considera que se podría mejorar la admisión de nuevos pacientes al PSM? 4. ¿Qué datos se requieren para poder atender a un paciente nuevo en el PSM? 5. ¿Existen datos adicionales que podrían mejorar el seguimiento de un paciente? Si es así ¿Cuáles serían estos? 6. ¿Qué datos recopila actualmente de los pacientes dentro de su sistema? 7. ¿Cómo cree que podría realizarse un intercambio de datos con otros sistemas de información médica? 8. ¿Qué datos desearía poder obtener de otros sistemas de información médica? 9. ¿Cómo desearía que otros proveedores de servicios médicos se integren al middleware? 10. ¿Cómo se maneja el acceso a los datos del sistema? 11. ¿Cómo se manejaría el acceso a los datos del middleware? 12. ¿Cómo se incluye a un nuevo usuario dentro del sistema? 13. ¿Cómo se incluiría a un nuevo usuario dentro del middleware? 14. ¿Los usuarios pueden acceder a su historial médico a través del sistema?

3.2.1.2. Requisitos obtenidos

A nivel general se obtuvieron los siguientes requerimientos:

1. Desarrollar una herramienta capaz de transformar el esquema de datos de un proveedor de servicios médicos en uno diferente.
2. Manejar la autenticación y autorización de los usuarios que accedan a dicha herramienta.
3. Establecer una interfaz RESTful para la comunicación con los diferentes proveedores de servicios médicos.

4. Extraer los datos de las diferentes bases de datos de los proveedores de servicios médicos.
5. Diseñar herramienta que permita automatizar la inclusión de nuevos proveedores de servicios médicos al middleware.

3.2.1.3. Elaboración de los requisitos

En esta sección, los requisitos obtenidos se definen de una manera más específica.

1. El middleware será capaz de transformar los datos obtenidos de los siguientes formularios médicos:
    a. Anamnesis
    b. Consulta Externa – Anamnesis y EF
    c. Consulta Externa – Evolución
    d. Evolución y prescripciones
    e. Examen Físico
2. El middleware obtendrá los datos de los diferentes sistemas de información médica consorciados o incluidos dentro del middleware.
3. El middleware proveerá los datos a través de una interfaz RESTful a través del formato XML.
4. El middleware manejará el acceso restringido de los usuarios
5. Se permitirá crear, eliminar, actualizar y leer usuarios.
6. Se manejará un tiempo de sesión de 12 horas desde la última autenticación.
7. El formato de transformación de los datos deberá ser FHIR.
8. El middleware podrá recibir datos en formato HL7 V2 también para su posterior conversión
9. Se manejará niveles de acceso a los datos de acuerdo a las políticas de INNTRATEC S.A.
    a. Básico: en este nivel se proveerá únicamente los datos incluidos en el formulario 001 del MSP
    b. Avanzado: en este nivel se podrá acceder a la información de todos los formularios mencionados en el primer requerimiento

10. La solución provista debe contar con una herramienta que permita generar los archivos necesarios para incluir a un nuevo proveedor de servicios médicos al middleware, todo esto a nivel interno de la organización.
11. Los usuarios podrán solicitar datos de pacientes que se encuentren dentro de las bases de datos consorciadas y de otros clientes.

3.2.1.4. Definición de los requerimientos prioritarios

Dado que, los requisitos no tienen la misma prioridad, para fines del desarrollo, se listarán los que tengan la prioridad más alta para asegurar la satisfacción de los usuarios. A continuación, se listan los mismos:

1. El middleware deberá contar con autorización y autenticación
2. El middleware permitirá convertir los datos de los sistemas de información médica sin importar su estructura interna.
3. El middleware deberá transformar los datos usando el estándar FHIR.

El documento de especificación de requerimientos se encuentra en el anexo 1.

3.3. Arquitectura REST y comunicaciones

Las comunicaciones del middleware se realizan a través de una API REST, que es la encargada de recibir las solicitudes de los clientes, procesarlas y devolver un resultado. El middleware está construido de tal manera que cada uno de los recursos a representar tiene su propio controlador REST. Estos controladores son llamados a través de *requests* HTTP, siguiendo lo establecido en la sección 2.6.6. Como se indicó en la sección 2.6.5, FHIR es una especificación RESTful. Por lo que, en esta sección se define la arquitectura que esta especificación maneja. La arquitectura se presenta en la figura 3.2. Dado que, el objetivo del middleware construido es únicamente obtener los datos médicos de distintos pacientes, el único método a usar para comunicarse con el middleware es: GET, de esta forma, los clientes envían todas las solicitudes y los criterios de búsqueda a través de la URL.

Figura 3.2. Arquitectura REST del middleware. Fuente: Elaboración propia

3.4. Variables a consideración para establecer la interoperabilidad

Para establecer la interoperabilidad dentro de los sistemas de información médica ecuatoriana, primero es necesario establecer una correlación entre los datos que utiliza la industria y los datos que se usan en el estándar FHIR. Para esto, se hizo un análisis de cada uno de los datos que se requieren para los formularios médicos de: Anamnesis, consulta externa – anamnesis y EF, consulta externa – evolución, evolución y prescripciones, y examen físico.

Cada uno de estos formularios, son físicos, por lo que, muchos datos se repiten en cada uno de ellos para su diferenciación. Sin embargo, para la conversión digital muchos de estos datos no son necesarios, por lo que, son omitidos, evitando la duplicación de los mismos.

3.4.1. Formulario 001

En esta sección se presentan los datos relacionados al formulario 001 de historia clínica única del MSP.

Tabla 3. Campos requeridos en el formulario 001 del MSP. Fuente: Elaboración propia

Institución del sistema
Unidad operativa
COD. UO
Parroquia
Cantón
Provincia
Número de historia clínica
Apellido paterno
Apellido materno
Primer nombre
Segundo nombre
Número de cédula
Dirección de residencia habitual
Barrio
Parroquia
Cantón
Provincia
Zona
Número teléfono
Fecha nacimiento
Lugar de nacimiento
Nacionalidad
Grupo cultural
Edad Años cumplidos
Sexo
Estado civil
Instrucción último año aprobado
Fecha de admisión
Ocupación
Empresa donde trabaja
Tipo de seguro de salud
Referido de
En caso necesario llamar a
Parentesco - afinidad
Dirección
Número de teléfono
Admisionista

3.4.2. Formulario 002:

En esta sección se presentan los datos relacionados al formulario 002 de historia clínica única del MSP.

3.4.2.1. Anamnesis – Consulta externa

Tabla 4. Campos requeridos en el formulario 002, sección anamnesis - consulta externa del MSP. Fuente: Elaboración propia

Establecimiento
Nombre
Apellido
Sexo
Número de hoja
Historia clínica
Motivo de consulta
Vacunas
Enfermedad perinatal
Enfermedad infancia
Enfermedad adolescencia
Enfermedad alérgica
Enfermedad cardiaca
Enfermedad respiratoria
Enfermedad digestiva
Enfermedad neurológica
Enfermedad metabólica
Enfermedad hemo linfática
Enfermedad urinaria
Enfermedad traumática
Enfermedad quirúrgica
Enfermedad mental
Enfermedad de transmisión sexual
Tendencia sexual
Riesgo social
Riesgo laboral
Riesgo familiar
Actividad física
Dieta y hábitos
Religión y cultura
Otro
Menarquia edad
Menopausia edad
Ciclos
Vida sexual activa
Gesta
Partos
Abortos
Cesáreas
Hijos vivos
FUM (fecha última menstruación)
FUC (fecha última citología)

FUP (fecha ultimo parto)
Biopsia
Método de P. familiar
Terapia hormonal
Colposcopía
Mamografía

3.4.2.2. Consulta externa - Evolución

En esta sección se presentan los campos a extraer de consulta externa – evolución

Tabla 5. Campos requeridos en el formulario 002, sección consulta externa - evolución del MSP. Fuente: Elaboración propia

Fecha
Hora
Evolución
Prescripciones
Medicamentos

3.4.3. Formulario 003

En esta sección se presentan los datos relacionados al formulario 003 de historia clínica única del MSP.

3.4.3.1. Anamnesis

Tabla 6. Campos requeridos en el formulario 003, sección anamnesis del MSP. Fuente: Elaboración propia

Establecimiento
Nombre
Apellido
Sexo
Número de hoja
Historia clínica
Motivo de consulta
Vacunas
Enfermedad perinatal
Enfermedad infancia
Enfermedad adolescencia

Enfermedad alérgica
Enfermedad cardiaca
Enfermedad respiratoria
Enfermedad digestiva
Enfermedad neurológica
Enfermedad metabólica
Enfermedad hemo linfática
Enfermedad urinaria
Enfermedad traumática
Enfermedad quirúrgica
Enfermedad mental
Enfermedad de transmisión sexual
Tendencia sexual
Riesgo social
Riesgo laboral
Riesgo familiar
Actividad física
Dieta y hábitos
Religión y cultura
Otro
Menarquia edad
Menopausia edad
Ciclos
Vida sexual activa
Gesta
Partos
Abortos
Cesáreas
Hijos vivos
FUM (fecha última menstruación)
FUC (fecha última citología)
FUP (fecha ultimo parto)
Biopsia
Método de P. familiar
Terapia hormonal
Colposcopía
Mamografía

3.4.3.2. Examen Físico

Tabla 7. Campos requeridos en el formulario 003, sección examen físico del MSP. Fuente: Elaboración propia

Índice de masa corporal

Presión arterial
Frecuencia cardiaca por min
Frecuencia respiratoria por min
Temperatura Bucal (grados centígrados)
Temperatura axilar (grados centígrados)
Peso (KG)
Talla (m)
Perímetro cefálico (cm)
Piel y faneras
Cabeza
Ojos
Oídos
Nariz
Boca
Oro Faringe
Cuello
Axilas - Mamas
Tórax
Abdomen
Columna vertebral
Ingle - Periné
Miembros superiores
Miembros inferiores
Órganos de los sentidos
Respiratorio
Cardio Vascular
Digestivo
Genital
Urinario
Musculo esquelético
Endocrino
Hemo linfático
Neurológico
Descripción
CIE
Presuntivo o definitivo
Fecha de control
Hora fin
Médico
Código
Firma

3.4.4. Formulario 005

En esta sección se presentan los datos relacionados al formulario 005 de historia clínica única del MSP.

3.4.4.1. Evolución y prescripciones

Tabla 8. Campos requeridos en el formulario 003, sección evolución y prescripciones del MSP. Fuente: Elaboración propia

Establecimiento
Nombre
Apellido
Sexo
Número de hoja
Historia clínica
Fecha
Hora
Evolución
Prescripciones
Medicamentos

Con estos datos extraídos ya es posible modelar los formularios 001, 002, 003 y 005 del MSP a través del estándar.

La alineación de estos campos con los recursos FHIR se entrega en el anexo 2, y por temas de confidencialidad no se lista dentro del documento de titulación.

3.5. Implementación del proveedor/extractor de datos

Para la extracción y provisión de datos se generó un conjunto de clases que permite la conexión segura tanto al middleware por parte de los clientes, así como, del middleware hacia las bases de datos externas. En la figura 3.3, se encuentra el diagrama de seguridad elaborado para esta implementación. El middleware maneja niveles de acceso que permiten obtener diferentes datos de los pacientes. Para esto el middleware cuenta con un gestor de usuarios que permite la creación, edición, lectura y eliminación de los usuarios.

Para la creación de usuarios se requieren los siguientes datos del mismo:

- Nombre de usuario
- Contraseña
- Correo electrónico
- Estado
- Rol

El campo “estado” se usa para desactivar o activar el acceso de un cliente temporalmente, mientras que, el rol es el nivel de acceso antes mencionado. Es necesario aclarar que, aparte de los 3 niveles de acceso existe uno adicional que representa al usuario administrador del middleware dentro del middleware.

El único usuario capaz de gestionar a los clientes del middleware es el administrador del middleware.

3.5.1. Seguridad implementada

El esquema de seguridad usado es *JSON Web Token* (JWT) que permite manejar el acceso de los usuarios y además el tiempo de sesión de cada uno de ellos. Todos los atributos del usuario son almacenados en texto plano en una base de datos MySQL, excepto la contraseña.

Para la contraseña se utiliza el algoritmo BCrypt, algoritmo de hash muy usado en estos casos que encripta el valor de la contraseña. Por otro lado, encima de esta encripatación se encuentra la capa de blockchain que será la que maneje el acceso a la autorización del sistema, esta seguridad se detalla en la tesis titulada: *Propuesta de Identidad Digital para Historial Clínico Unificado utilizando tecnología Blockchain* (Cáceres & Peralta, 2021). La capa actúa como proxy validando la identidad de cada una de las peticiones realizadas al middleware

3.5.2. Manejo de la extracción de datos

Como se mencionó, el middleware extrae y provee datos dependiendo del nivel de acceso que tenga el cliente en cuestión, cabe aclarar que este nivel de acceso coincide con lo especificado en la tesis titulada: *Propuesta de Identidad Digital para*

Historial Clínico Unificado utilizando tecnología Blockchain (Cáceres & Peralta, 2021). La descripción y acceso a la información de cada nivel se detalla a continuación:

- Nivel de acceso 1: contempla el acceso únicamente a los datos demográficos de los pacientes, es decir todos los datos correspondientes al formulario 001 del MSP y listados en la tabla 3.
- Nivel de acceso 2: en este caso el acceso se eleva y además de incluir lo mencionado en el nivel de acceso 1, se puede acceder a todos los antecedentes del paciente, tanto personales, como familiares. Estos datos se encuentran en los formularios 002 y 003 en las secciones de anamnesis – consulta externa y anamnesis, respectivamente.
- Nivel de acceso 3: en este nivel se puede acceder a todos los datos disponibles en el middleware, es decir, además de incluir lo mencionado en los niveles de acceso 1 y 2, se puede acceder a los formularios 002, 003 y 005 por completo.

Como la extracción de los datos se realiza a bases de datos externas, el esquema de seguridad de cada una de ellas es variable, por lo que, es de vital importancia saber el esquema de antemano y que los administradores de cada una de las bases de datos, creen las credenciales necesarias para el acceso del middleware a sus propias bases de datos.

Al momento de la creación de este middleware, se han encontrado únicamente esquemas de seguridad basados en JWT.

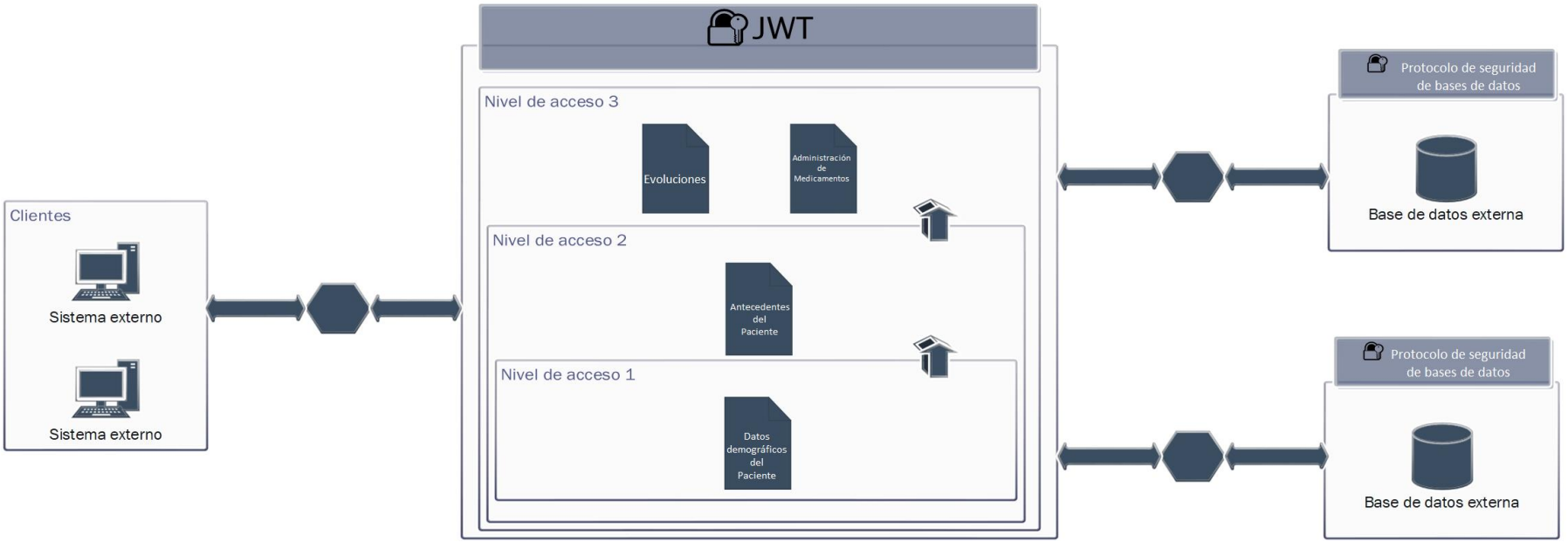


Figura 3.3. Esquema de seguridad. Fuente: Elaboración propia

3.6. Construcción de los servicios para FHIR y HL7

Para la construcción de los servicios relacionados a los estándares FHIR y HL7, actualmente, existen numerosas opciones de implementación, por lo que, en esta sección se presenta una breve comparación con la que se determinó la mejor para este caso.

Esta comparación tuvo en cuenta 5 atributos clave:

- **Modelos de implementación en la nube:** describe la forma en que se almacenan los modelos.
- **Modelos FHIR soportados:** se refiere a todos los modelos FHIR que acepta la implementación.
- **Basado en código abierto:** describe si la implementación se basa en código abierto.
- **Opciones de servidor elástico:** se refiere a si la implementación tiene opciones de servidor elástico.
- **Serverless:** describe si la implementación puede prescindir de un servidor o no.

De todas las opciones disponibles, 5 se ajustan provisionalmente al objetivo de este trabajo. Estas son:

- **HAPI FHIR:** es una implementación completa del estándar HL7 FHIR para la interoperabilidad sanitaria en Java.
- **SMILE CDR:** Smile CDR es un repositorio de datos clínicos completo y especialmente diseñado según el estándar HL7 FHIR que se utiliza para almacenar registros médicos. Basado en HAPI FHIR.
- **FHIR AZURE:** la API de Azure para FHIR permite una conexión rápida a las fuentes de datos actuales, como los sistemas de registros médicos electrónicos o las bases de datos de investigación.
- **API GOOGLE DE CLOUD HEALTHCARE:** La API Cloud Healthcare permite intercambiar datos de una manera simple y estandarizada entre aplicaciones y soluciones de atención médica compiladas en Google Cloud.

- **API de código abierto FHIR AWS:** kit de herramientas de software de código abierto que se puede utilizar para agregar capacidades de una interfaz FHIR al software de salud existente.

La Tabla 9 presenta un resumen de las funcionalidades de las diferentes opciones propuestas. Cabe señalar que SMILE CDR se basa en HAPI FHIR, que es un repositorio completo de datos clínicos y está especialmente diseñado de acuerdo con el estándar HL7 FHIR que se utiliza para almacenar registros médicos.

Además, este junto con HAPI FHIR requieren una fase posterior donde se determina el lugar de hospedaje, lo que conlleva una serie de pasos adicionales. Por otro lado, Azure, Google Cloud y AWS ofrecen estos servicios de forma manejada o *managed*, lo que facilita su desarrollo y mantenimiento.

Tabla 9. Comparativa entre las diferentes opciones de implementación del estándar FHIR. Fuente: Elaboración propia

Atributo \ Plataforma	HAPI FHIR	SMILE CDR	FHIR AZURE	CLOUD HEALTH -CARE API GOOGLE	OPEN SOURCE FHIR APIS AWS
Modelos de implementación en la nube	Necesita nube pública	Necesita nube pública	Necesita nube pública	Necesita nube pública	Necesita nube pública
Modelos FHIR soportados	DSTU1, DSTU2, DSTU2.1, DSTU3, R4, R5	DSTU1, DSTU2, DSTU2.1, DSTU3, R4, R5	R4, STU3	R4, STU3, DSTU2	STU3 y R4.0.1
Basado en código abierto	Si	Si	Tiene una opción de código abierto para implementar en sus servidores.	No	Si
Opciones de servidor elástico	Depende del servidor	Depende del servidor	Si	Si	Si
Serverless	No	No	Si	Si	Si

Con los resultados de la comparación, se decidió usar HAPI FHIR como la mejor implementación para este caso, ya que alberga la mayoría de los modelos FHIR junto con SMILE CDR, pero la diferencia es que HAPI FHIR es de uso gratuito mientras que, SMILE CDR tiene un costo asociado a su uso. Dado que, HAPI FHIR es una API de Java, se decidió implementar el middleware sobre este lenguaje. Para facilitar el desarrollo, y por la naturaleza del middleware y de servicios de este trabajo, se decidió usar junto al framework de Spring.

3.6.1. Controladores REST

Para el manejo de todos los datos extraídos en las tablas desde la 3 hasta la 8, se elaboraron 13 controladores REST que reciben cada una de las solicitudes HTTP. Estos controladores tienen asociados los modelos y lógica asociada para su funcionamiento. Los mismos se describen como sigue:

AllergyIntoleranceRest

Este controlador es el encargado de manejar todos los datos relacionados a las **alergias** del paciente. Se encuentra en el siguiente endpoint:

[Base-address]/fhir/AllergyIntolerance/

La consulta se puede hacer de manera individual, es decir, incluyendo el id de la alergia y el id del paciente al final del endpoint. La petición GET debe hacerse a la siguiente URL:

[Base-address]/fhir/AllergyIntolerance?patient:Patient=$id&identifier=$ident

Donde $id se deber reemplazar por el id del paciente y $ident se debe reemplazar por el identificador de la alergia. Por otro lado, también se puede consultar todas las alergias de un determinado paciente, para esto, la solicitud GET debería hacerse a la URL:

[Base-address]/fhir/AllergyIntolerance?patient:Patient=$id

BackgroundPersonalAffectionRest

La tarea de este controlador es manejar todos los **antecedentes personales** de cada uno de los pacientes. Este controlador se encuentra en el endpoint:

[Base-address]/fhir/Condition/

Para obtener un solo antecedente personal de un paciente es necesario realizar la petición GET a la siguiente URL:

[Base-address]/fhir/AllergyIntolerance?patient:Patient=$id&identifier=$ident

Donde $id se deber reemplazar por el id del paciente y $ident se debe reemplazar por el identificador del antecedente personal. Para consultar todos los antecedentes de un determinado paciente, la petición GET debe hacerse a:

[Base-address]/fhir/Condition?patient:Patient=$id

EvolutionRest

Este controlador es el encargado de dar acceso a los datos referentes a una **evolución** de un paciente, el endpoint de este controlador es:

[Base-address]/fhir/ClinicalImpression/

El único método permitido por este controlador es el de obtener todas las evoluciones referentes a un paciente. Para obtener estos datos es necesario realizar la petición GET a la siguiente URL:

[Base-address]/fhir/Condition?patient:Patient=$id&history=true

FamilyBackgroundRest

Este controlador es el encargado de dar acceso a los datos referentes a los **antecedentes del familiar** de un paciente, el endpoint de este controlador es:

[Base-address]/fhir/FamilyMemberHistory/

Uno de los métodos permitidos por este controlador es el de obtener los antecedentes de un paciente en específico. Para obtener estos datos es necesario realizar la petición GET a la siguiente URL:

[Base-address]/FamilyMemberHistory?patient:Patient=$id&identifier=$ident

Aquí el identificador al final de la URL debe ser reemplazado por el identificador del antecedente familiar en cuestión. Por otro lado, para obtener todos los antecedentes familiares de un paciente se debe realizar una petición GET a:

[Base-address]/fhir/FamilyMemberHistory?patient:Patient=$id

MedicalImpressionRest

Este controlador es el encargado de dar acceso a los datos referentes a los **diagnósticos** de un paciente, el endpoint de este controlador es:

[Base-address]/fhir/ClinicalImpression/

Uno de los métodos permitidos por este controlador es el de obtener un diagnóstico de un paciente en específico. Para obtener estos datos es necesario realizar la petición GET a la siguiente URL:

[Base-address]/ClinicalImpression?patient:Patient=$id&identifier=$ident

Aquí el identificador al final de la URL debe ser reemplazado por el identificador del diagnóstico en cuestión. Por otro lado, para obtener todos los diagnósticos de un paciente se debe realizar una petición GET a:

[Base-address]/fhir/ClinicalImpression?patient:Patient=$id

MedicationPrescriptionRest

Este controlador es el encargado de dar acceso a los datos referentes a las **recetas** realizadas en un determinado PSM, el endpoint de este controlador es:

[Base-address]/fhir/MedicationDispense/

El único método aceptado por este controlador es el de obtener una receta en específico. Para obtener estos datos es necesario realizar la petición GET a la siguiente URL:

[Base-address]/fhir/MedicationDispense?identifier=$ident

MedicationRest

Este controlador es el encargado de dar acceso a los datos referentes a los **medicamentos** referenciados durante el proceso de extracción de datos, el endpoint de este controlador es:

[Base-address]/fhir/Medication/

Es importante recalcar que para este caso se usó la codificación brindada por *Snomed-CT*[7] que es una terminología clínica que permite identificar, entre otras cosas, a los medicamentos.

[7] SNOMED - Home | SNOMED International

Uno de los métodos permitidos por este controlador es el de obtener un medicamento en específico. Para obtener estos datos es necesario realizar la petición GET a la siguiente URL:

[Base-address]/fhir/Medication/$id

Aquí el identificador al final de la URL debe ser reemplazado por el identificador del medicamento en cuestión. Por otro lado, existe un método que permite obtener todos los medicamentos pertenecientes a una prescripción en específico, para esto se debe realizar una petición GET a:

[Base-address]/fhir/Medication?identifier=$ident

PatientRest

Este controlador es el encargado de dar acceso a los **datos demográficos** referentes a los pacientes, el endpoint de este controlador es:

[Base-address]/fhir/Patient/

El único método permitido por este controlador es el de obtener a un paciente en específico. Para obtener estos datos es necesario realizar la petición GET a la siguiente URL:

[Base-address]/fhir/Patient/$id

PersonalBackgroungRest

Este controlador es el encargado de dar acceso a todos los datos referentes a los antecedentes de un paciente esto incluye: **las alergias, vacunas, antecedentes personales y antecedentes familiares**, el endpoint de este controlador es:

[Base-address]/fhir/Condition/

El único método permitido por este controlador es el de obtener los antecedentes de un paciente en específico. Para obtener estos datos es necesario realizar la petición GET a la siguiente URL:

[Base-address]/fhir/Condition?patient:Patient=$id&history=true

PlanRest

Este controlador es el encargado de dar acceso a todos los datos referentes a los **planes de salud** ofertados por los diferentes PSM, el endpoint de este controlador es:

[Base-address]/fhir/CarePlan/

Uno de los métodos permitidos por este controlador es el de obtener un plan de salud según su id. Para obtener estos datos es necesario realizar la petición GET a la siguiente URL:

[Base-address]/fhir/CarePlan?identifier=$ident

El método restante es el de obtener todos los planes de salud aplicados a un paciente, este servicio subyace en la siguiente URL:

[Base-address]/fhir/CarePlan?patient:Patient=$id

PrescriptionsRest

Este controlador es el encargado de dar acceso a todos los datos referentes a las **prescripciones** realizadas a un paciente, el endpoint de este controlador es:

[Base-address]/fhir/MedicationAdministration/

El único método permitido por este controlador es el de obtener una prescripción relacionada a un paciente en específico. Para obtener estos datos es necesario realizar la petición GET a la siguiente URL:

[Base-address]/
MedicationAdministration?patient:Patient=$id&idPrescription=$idpre

Donde $idpre debe ser reemplazado por el identificador de la prescripción en cuestión.

VaccineRest

Este controlador es el encargado de dar acceso a todos los datos referentes a las **vacunas** realizadas a un paciente, el endpoint de este controlador es:

[Base-address]/fhir/Immunization/

Uno de los métodos permitidos por este controlador es el de obtener una vacuna aplicada a un paciente en específico. Para obtener estos datos es necesario realizar la petición GET a la siguiente URL:

[Base-address]/fhi/rImmunization?patient:Patient=$id&identifier=$ident

El método restante es el que permite obtener todas las vacunas aplicadas a un paciente en específico. Para esto se debe hacer la solicitud GET a:

[Base-address]/fhir/Immunization?patient:Patient=$id

VitalSignsRest

Este controlador es el encargado de dar acceso a todos los **signos vitales** obtenidos de determinado paciente, el endpoint de este controlador es:

[Base-address]/fhir/Observation/

El único método permitido por este controlador es el de obtener los signos vitales obtenidos en una revisión de un paciente en específico. Para obtener estos datos es necesario realizar la petición GET a la siguiente URL:

[Base-address]/fhir/Observation?subject:Patient=$id&identifier=$ident

Donde $ident debe ser reemplazado por el identificador de los signos vitales en cuestión.

3.6.2. Módulo CIE 10

Las enfermedades son las mismas independientemente del PSM en cuestión, se generó un módulo que se encarga de obtener las enfermedades siguiendo la clasificación internacional CIE10 mencionada en la sección 2.7.

Para esto, se maneja un archivo JSON que contiene toda la especificación del CIE10. En este archivo, la información se encuentra estructurada de manera que para cada código del CIE10 existe una descripción correspondiente.

Dado que, esta información puede ser de interés para los PSM que se encuentren del middleware se ha decidido desarrollar un endpoint que se encarga de devolver el código y descripción de un elemento del CIE10 según su código. El endpoint está estructurado como sigue:

[Base-address]/CIE10/$id

Donde $id es el código CIE10 a solicitar.

3.7. Implementación del sistema

Dado que, como se mencionó en la sección 3.1, el procedimiento de conversión se aplicará a los datos relacionados con la salud almacenados en Ecuador, es necesario tener una descripción de cada uno de estos recursos.

Esta descripción se puede obtener del documento: Expediente único para la Historia Clínica emitido en 2007 mediante Acuerdo Ministerial NO. 0000116 de Ecuador de donde se obtuvieron los siguientes documentos:

- "Anamnesis"
- “Consulta Externa - Anamnesis y Examen Físico”
- “Consulta Externa - Evolución”
- “Evolución y prescripciones”
- “Examen Físico”

Esta descripción se puede encontrar en la sección 3.4 de este documento.

Otro punto importante de la implementación es la elección del lenguaje con el que el middleware será desarrollado. En este caso, y por lo presentado en la sección 3.6, el lenguaje de desarrollo es JAVA en su versión 16, que es la última versión al momento de elaborar este trabajo de titulación. Como se explica en la sección 3.6, se hizo uso de dos herramientas más para el desarrollo que son: Spring boot y HAPI FHIR.

1. **Se establece el tipo de formato de intercambio.**

En este caso, el formato elegido para el intercambio es XML porque este formato usa lenguaje humano, es legible y comprensible. Otra razón para utilizar este formato es su compatibilidad con JAVA.

2. **Se establecen las terminologías.**

Al igual que, en el paso anterior, en este caso se pueden encontrar varias opciones que son válidas para el desarrollo, sin embargo, teniendo en cuenta que el middleware está enfocado a ser desplegado en América Latina y específicamente en Ecuador, se ha decidido utilizar CIE10 en su versión en español (CIE10) (ver sección 3.6.2).

Por otro lado, para definir los roles de los miembros de la familia, se utilizó el sistema de códigos V3-RoleCode[8], este sistema de códigos es ampliamente usado por FHIR para codificar los roles familiares.

[8] RoleCode - FHIR v4.0.1 (hl7.org)

3. **Establecer los mapeos que se usarán.**

Para este caso, se decidió utilizar el trabajo de LinuxForHealth (2020) como conversor de datos de HL7 V2 a FHIR, ya que este documento se enfoca en la obtención de recursos FHIR.

4. **Establecer el RESTful API sobre HTTP.**

El método de solicitud que se utiliza es GET como se menciona en la sección 3.3.

5. **Recopilar todos los recursos FHIR**

En este paso se llevó a cabo la conversión de cada uno de los atributos descritos en el "Expediente único para la Historia Clínica" y presentado en la sección 3.3. Cada uno de los atributos se alineó con un campo de un recurso FHIR o con un recurso completo, según el caso.

Cabe señalar que, si no se encuentra específicamente la información requerida, FHIR presenta la opción de hacer extensiones a cada uno de sus recursos, por lo que, si este es el caso, se debe optar por esta opción.

En este caso, un ejemplo serían las vacunas de un paciente, estas corresponden al formulario 002 - "Anamnesis - consulta externa", a la subcategoría de "Antecedentes personales", inicialmente iría el nombre de la vacuna y una descripción, sin embargo, los sistemas de información médica agregan el código CIE10, más una identificación.

Por ejemplo, un recurso interesante presente en anamnesis, son las vacunas, en la era actual saber que vacunas tiene un paciente se ha vuelto prioritario. Con la inserción de la información de la aplicación de las vacunas se vuelve evidente la ventaja del uso de FHIR para intercambiar la información de pacientes vacunados a nivel mundial.

En FHIR las vacunas están bajo el recurso de Inmunización por lo que, la alineación sería la siguiente:

Tabla 10. Alineamiento de recursos. Fuente: Elaboración propia

Vacuna (Anamnesis)	Immunization
Id del paciente	patient
Fecha	occurrence

Id	identifier
Nombre	vaccineCode
Descripción	note
Código	vaccineCode

Adicionalmente, el recurso especifica que se debe establecer el estado de la vacuna y la cantidad de dosis ya que esta es parte del historial del paciente, para que la vacuna sea listada debe haberse administrado por completo, por lo que, este campo debe ir siempre en “complete”, y si el número de dosis no existe, se especificará el número 1.

Para el caso en que la alineación no sea posible, como es el caso del apellido materno de un paciente, se deben utilizar extensiones, un ejemplo se presenta en la tabla 11:

Tabla 11. Alineamiento de campos no básicos de los recursos. Fuente: Elaboración propia

Paciente (Registro de Primera admisión)	Patient
Apellido Materno	patient-mothersMaidenName (extensión del paciente)

Si ninguna extensión coincide con el campo, se debe definir una en algún lugar accesible en Internet y se puede usar dentro de FHIR.

En esta fase, también, se generó cada uno de los archivos requeridos por el middleware para incluir un nuevo proveedor de servicios médicos en la conversión. Para ello, se decidió utilizar el enfoque de ingeniería basada en modelos (MDE, por sus siglas en inglés).

Aquí el nuevo proveedor de servicios médicos ingresa la información de la lista de conversión, previamente obtenida, dentro del modelo generado para el recurso.

Esta fase tiene como entrada el documento de la lista de conversión y como salida lo siguiente:

• **Clases de conversión generadas**: las clases necesarias para el funcionamiento del middleware.

• **Extractor de datos generados**: una clase que tiene la especificación de los puntos finales de los que se extraerán los datos.

La figura 3.4 muestra el modelo generado para el caso del paciente; este modelo es similar a una de las clases. Cada organización puede tener una definición de paciente dentro de su sistema. A su vez, estas dos entidades describen cada uno de los atributos necesarios para la conversión.

El proveedor de servicios médicos inserta el nombre de los atributos equivalentes en su base de datos. Este proceso se repite para cada uno de los recursos obtenidos para el middleware.

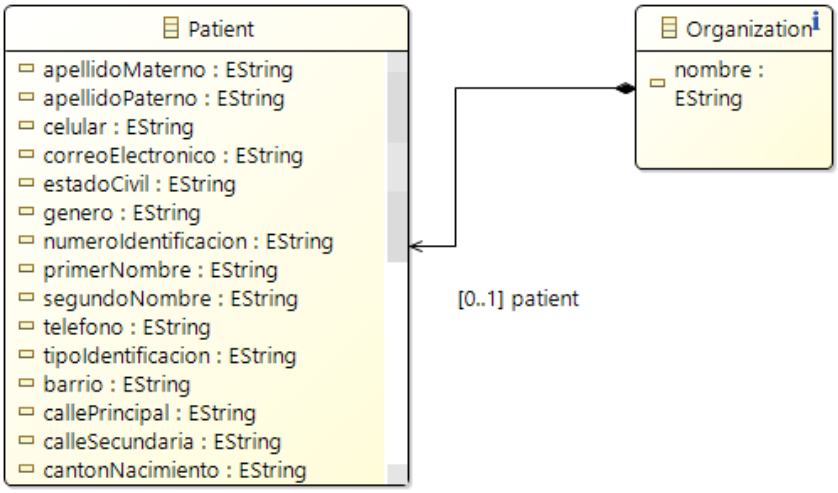


Figura 3.4. MDE del paciente. Fuente: Elaboración propia

6. Establecer las URLs para la búsqueda.

La estructura de las URL para la búsqueda utilizada es la sugerida por FHIR e implementada por HAPI FHIR. Esto se puede resumir en la siguiente cadena:

GET *[base] / [type]? Name = value & ... {& _ format = [mime-type]}}*

Donde [base] es la URL base del servidor, [type] es el nombre del recurso FHIR y los parámetros son una serie de pares de nombre-valor codificados en la URL.

En este caso, el servidor es el que determina el conjunto de recursos que coincide con los criterios especificados y devuelve los resultados en un paquete con todos los recursos.

Siguiendo nuestro ejemplo de la vacuna, la URL sería:

http://localhost:8080/fhir/Immunization?patient:Patient:1

Todas las URL de búsqueda se listan en la sección 3.6.1.

7. Establezca las URLs para las operaciones.

Dado que, el objetivo de este middleware es únicamente la extracción de datos inicialmente no se establece ningún tipo de operación sobre los datos, sin embargo, esto puede cambiar en un futuro.

8. Establezca el uso asincrónico.

Para este caso, se pretende que el uso del Sistema sea síncrono para todos los recursos, dado que, la cantidad de información no es demasiada y no se realizará demasiado procesamiento por parte del servidor, por esto, no se define el uso asíncrono.

9. Diseño de comunicaciones con el resto de sistemas

En este caso, el diseño de la comunicación es el que se presenta en la figura 1 de la sección 1.4. De esta forma se modela y codifica la extracción de datos de las diferentes bases de datos consorciadas.

10. Codificación y pruebas

El marco fue codificado y probado con varios tipos de estructuras de datos de los sistemas de información médica locales en Ecuador. Además, cada recurso se probó con Instance Validator, que es el validador de instancias HAPI FHIR.

11. Establecer el profiling.

En este caso, se establecen todas las restricciones inferidas del Expediente único para la Historia Clínica, es decir, solo se utilizarán los recursos obtenidos a través de este proceso, sin posibilidad de sumar el resto de los recursos especificados por FHIR.

Además, de que el uso de la API estará restringido por los niveles de acceso presentados en la sección 3.5.2. Como se indicó anteriormente, el módulo CIE10 se utilizará en su versión en español.

3.7.1. Modelo lógico obtenido

La arquitectura modular del susbsistema se presenta en la figura 3.5. El enfoque utilizado para el desarrollo del modelo lógico ha sido el modular, con esto se ha subdivido el subsistema en 3 módulos: módulo de estandarización (ES), módulo de integración (IN), módulo de gestión de usuarios (GU).

El módulo de **ES** es el que permite al susbsistema realizar las distintas tareas de conversión de datos, este módulo será transparente para los usuarios. El módulo de **IN** es el encargado de permitir la generación de los archivos necesarios para la inclusión de nuevos proveedores de servicios médicos, este es accedido por el administrador del middleware quién es el encargado de esta tarea.

Finalmente, el módulo de GU permite al administrador del middleware realizar las diferentes tareas de gestión de los usuarios pertenecientes al subsistema, como establecer sus niveles de acceso, editar su información, revocar su acceso, a parte de la creación y eliminación.

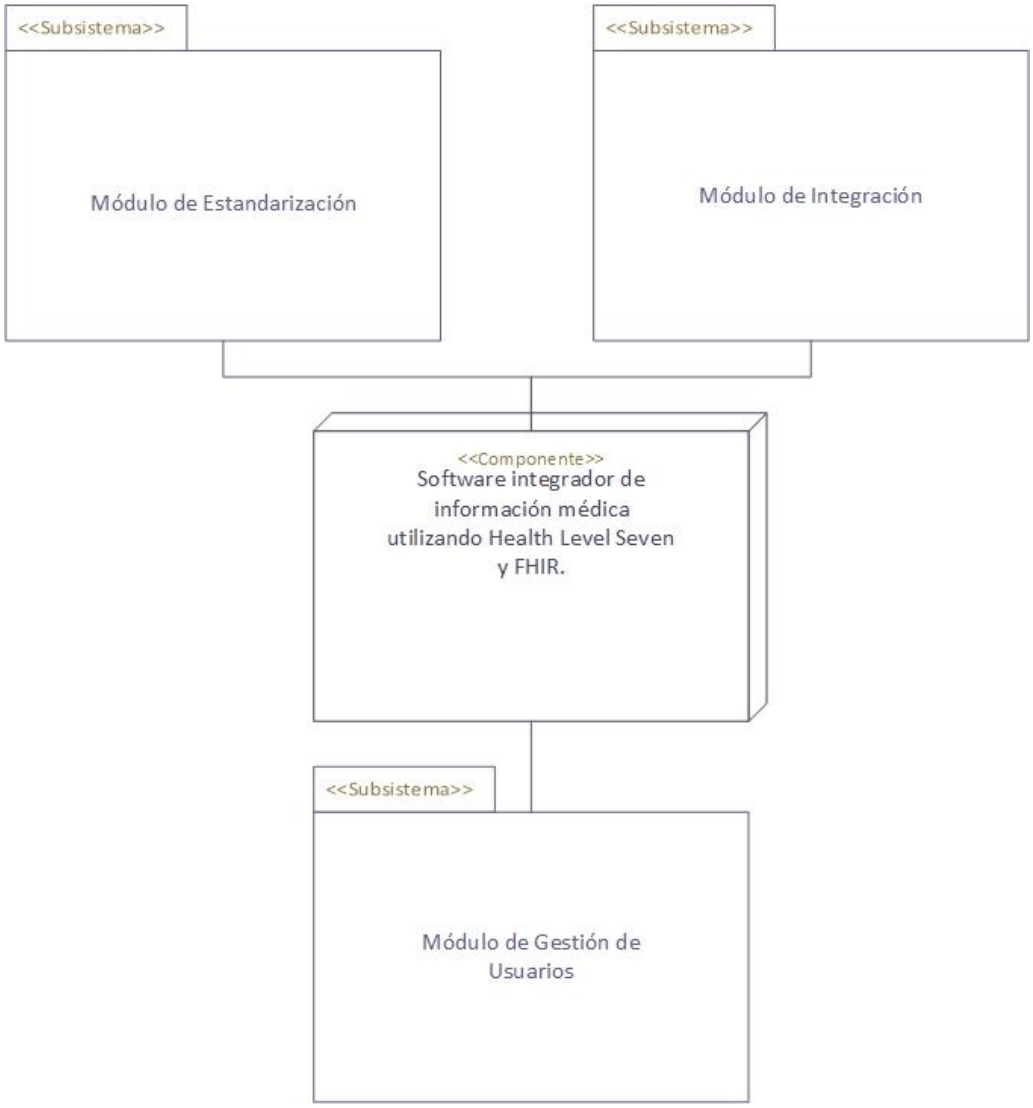


Figura 3.5. Arquitectura modular del subsistema. Fuente: Elaboración propia

La funcionalidad de cada uno de los módulos se presenta a continuación:

Módulo de ES: como se mencionó anteriormente, este módulo es el encargado de recopilar y transformar la información de los distintos proveedores de servicios médicos y convertirla a FHIR.

La conversión inicia con la búsqueda de cualquier información perteneciente a un paciente, es decir, si se busca a un paciente con un numero de cedula especifico, el middleware lo busca en todas las bases de datos consorciadas y devuelve el resultado de la búsqueda en el estándar FHIR.

Este módulo además se encarga de gestionar los diferentes accesos de los usuarios. Para esto se almacena una contraseña y usuario que se requerirá para poder consultar los diferentes *endpoints* expuestos por el middleware.

Módulo de IN: Este módulo es el encargado de generar los archivos necesarios para la inclusión de un nuevo PSM al middleware. La conversión inicia con la solicitud de un PSM, posteriormente, se solicita llenar los documentos de correlación de cada uno de los recursos que el subsistema permite.

En este módulo se encuentra un proceso de *Model Driven Engineering* (MDE) que permite la generación automática de los archivos. Por esto, el administrador debe reemplazar la información obtenida del PSM en el MDE y ejecutar el proceso. Una vez que se generen los archivos los posiciona en los respectivos lugares dentro del middleware.

Módulo de GU: Este módulo es el encargado de realizar todas las operaciones de gestión de usuarios dentro del middleware, dado que, existen varios niveles de acceso, la necesidad de este módulo es evidente. El módulo tiene una interfaz web a la cual solo puede acceder el administrador del middleware.

Una vez autenticado, el módulo permite: la creación, eliminación, edición y lectura de los diferentes usuarios en el middleware, de una manera estructurada. Este módulo se comunica directamente con el subsistema.

3.7.2. Herramienta de gestión de usuarios

Dado que, el manejo del acceso a los datos depende del nivel de acceso de cada usuario se decidió elaborar una interfaz gráfica que facilite al administrador del middleware gestionar cada uno de los detalles de los usuarios existentes.

Los servicios de autenticación presentados en la sección 3.5, serán consumidos por una aplicación web.

Esta aplicación fue construida con el *framework React*, ya que se es una tendencia al momento de redactar este documento de titulación y esto facilita el hallazgo de documentación oportuna

La aplicación se basa en tres pantallas principales:

1. **Pantalla de inicio de sesión:** en esta pantalla se implementaron los servicios de inicio de sesión para los administradores del middleware, cabe

recalcar que el único tipo de usuario capaz de acceder a esta aplicación es el usuario administrador del middleware. La pantalla se puede observar en la figura 3.6.

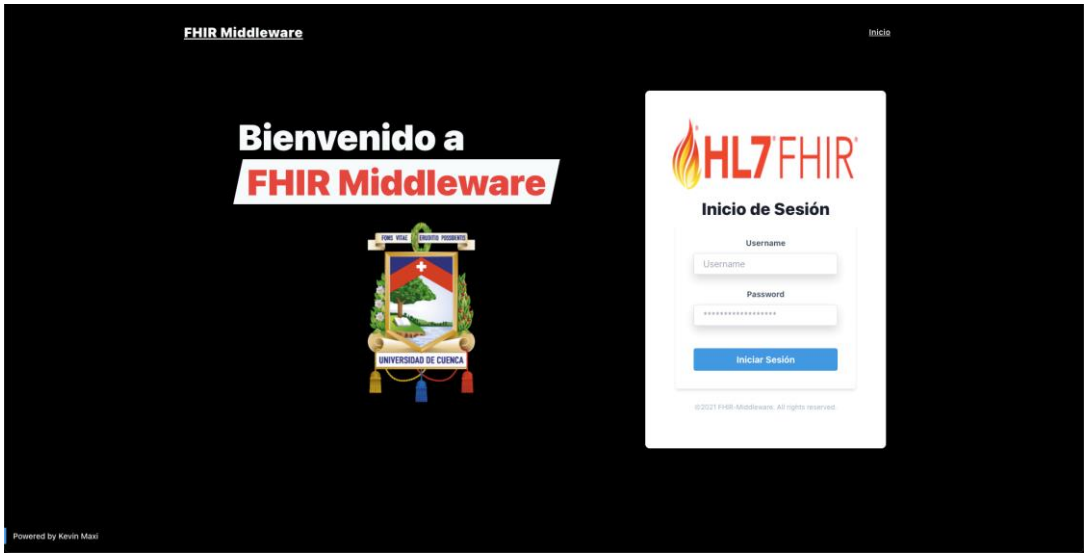


Figura 3.6 Pantalla de inicio de sesión. Fuente: Elaboración propia

2. Pantalla de recuento de usuarios: esta pantalla muestra a todos los usuarios registrados en el middleware incluidos a los administradores. Desde aquí se puede acceder a añadir nuevos usuarios, editar usuarios, y borrar usuarios. También es posible cerrar sesión. Esta pantalla se muestra en la figura 3.7

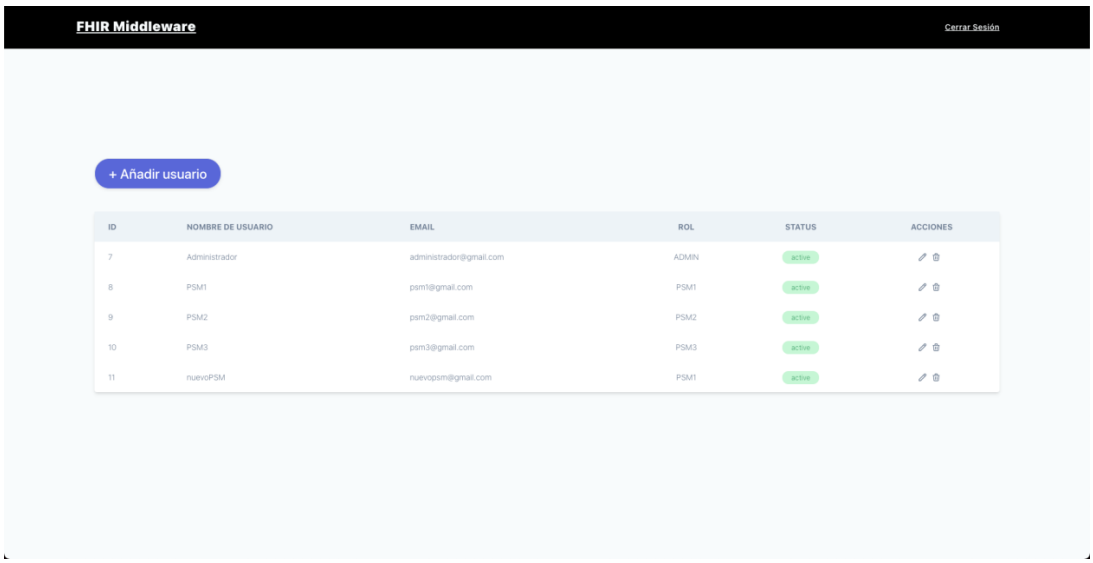


Figura 3.7 Pantalla de recuento de Usuarios. Fuente: Elaboración propia

3. Pantalla de creación/edición de usuarios: esta pantalla permite la edición o creación de usuarios. Cuando se elige la opción de edición los campos de email y contraseña no son editables, dado que, el primero es la clave primaria y el segundo es un campo de seguridad que no puede ser editado por los administradores. Esta pantalla se presenta en la figura 3.8

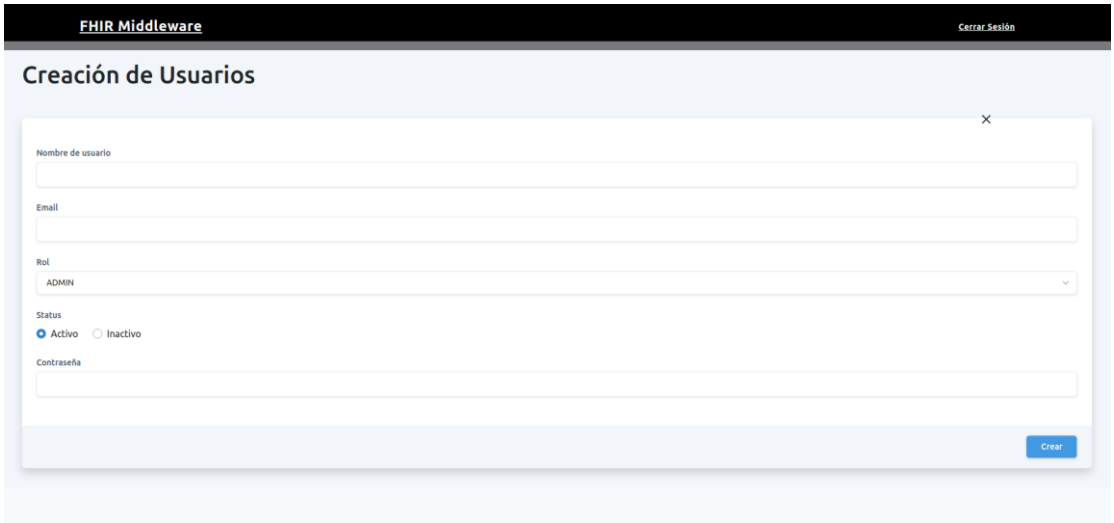


Figura 3.8 Pantalla de creación/edición usuarios. Fuente: Elaboración propia

CAPÍTULO 4: ANÁLISIS Y PRESENTACIÓN DE RESULTADOS

Luego de la construcción del middleware, un paso importante es validar su funcionamiento de acuerdo con lo planteado en esta tesis. Para ello, se realiza el diseño e implementación de un cuasi-experimento.

Un cuasi-experimento según C. Wholin (2012) es una prueba en donde las actividades emergen de las características de los sujetos que participan en la investigación, en este tipo de pruebas las actividades de los participantes no se pueden elegir de forma dinámica.

El objetivo principal del cuasi-experimento, siguiendo la plantilla de Meta / Pregunta / Métrica (C. Wholin et al., 2012) es:

- Analizar los recursos de prueba generados con el middleware desarrollado, con el propósito de realizar una evaluación con respecto a su efectividad al realizar la conversión de datos médicos en cualquier tipo de formato, a recursos FHIR correctamente elaborados.
  Además de determinar su flexibilidad con datos faltantes, todo esto, desde el punto de vista de ingenieros de software y personal médico calificado, dentro del contexto de las diferentes bases de datos de información médica del Ecuador.

El interés de este cuasi-experimento recae en la posibilidad de determinar si el middleware es capaz de convertir cualquier estructura de base de datos a un recurso FHIR de manera que se pueda compartir con cualquier sistema de información médica sin importar la estructura interna de los datos (heterogeneidad).

Para esto, se plantean y estudian las siguientes preguntas de investigación experimental:

- PIE1: ¿El middleware desarrollado es capaz de transformar los datos al estándar FHIR de modo que, se habilite la interoperabilidad?
- PIE2: ¿El middleware desarrollado es capaz de realizar las transformaciones de datos sin importar la estructura interna de cada base de datos, es decir su heterogeneidad?

- PIE3: ¿El middleware desarrollado cumple con los requisitos planteados en esta tesis?

Por consiguiente, PIE1 es evaluada mediante las hipótesis:

- $H1_0$: Se percibe que los resultados obtenidos por el middleware no son suficientes para alcanzar el objetivo de interoperabilidad entre sistemas de información médica, $H1_0=\neg H1_1$.

Mientras que, para PIE2:

- $H2_0$: El middleware no es flexible a la falta de datos o estructura interna de las bases de datos para la conversión, $H2_0=\neg H2_1$.

Por último, PIE3:

- $H3_0$: El middleware desarrollado difiere de los requisitos planteados en uno o más de los especificados, $H3_0=\neg H3_1$.

4.1. Variables

En cuanto a las variables que se tendrán en cuenta para el cuasi-experimento, se tiene:

4.1.1. Variables independientes

Se considera que la única variable independiente en este caso es:

- **Cantidad de atributos por clase**: la cantidad de atributos en una clase que representa un objeto que interactúa en el flujo de los datos de la industria médica, es un punto importante, dado que, la transformación depende directamente de la cantidad y calidad de cada uno de estos datos.

4.1.2. Variables dependientes

La variable dependiente que se espera sea influenciada en cierta medida por la variable dependiente es:

- **Efectividad para la transformación de datos**: para las PIE1 y PIE2 es necesario medir la efectividad que se tiene al usar la herramienta para transformar los datos a recursos FHIR, por lo que, es necesario representar esto

en un resultado tangible que permita responder de manera adecuada a la pregunta planteada y sin subjetividades.

4.1.3. Grupo de control:

Para el grupo de control se establecerá la conversión manual de cada uno de los recursos que se plantearan a continuación. De esta manera, se observa con claridad la diferencia entre usar el middleware e intentar realizar la conversión manual. El grupo de control constará de 10 usuarios de tipo desarrollador de los sistemas del PSM y 8 de tipo administrador del middleware.

4.2. Métricas

4.2.1. Tasa de transformación de datos:

La métrica de tasa de transformación de datos es una ponderación de la cantidad de datos usados y transformados a recursos FHIR de una entidad específica, contra la cantidad de datos provistos por la estructura interna de una base de datos en específico. En la ecuación (1) se presenta la fórmula para realizar el cálculo de esta métrica.

$$TTD = \frac{\text{número de datos incluidos en el recurso FHIR}}{\text{número de datos disponibles en la base de datos}}$$

4.3. Material Experimental

El material experimental que se usa es una breve descripción de los recursos FHIR donde se detalla la información del recurso a transformar, así como, cada uno de sus atributos. Este documento depende del recurso a utilizar.

En este caso, un ejemplo sería la descripción del recurso *Patient*, que define a un paciente. En la siguiente *URL* se encuentra la información directamente del sitio web:

Patient - FHIR v4.0.1 (hl7.org).

Otro recurso experimental, es la estructura de un paciente dentro de una base de datos específica, que, por temas de confidencialidad, es entregada al momento de ejecutar el cuasi-experimento posterior a un acuerdo de confidencialidad.

4.3.1. Casos de prueba

En este apartado se presentan los casos de prueba que se realizaron:

CASO DE PRUEBA 01: Generación automática de archivos para la inclusión de la entidad paciente de un nuevo PSM
Casos de uso que intervienen: Requisitos funcionales que intervienen: RQF18 Requisitos no funcionales que se comprueban: RQFN03, RQFN05, RQFN07, RQFN08, RQFN09, RQFN10
El usuario administrador del middleware, a través del metamodelo provisto, ingresará los atributos de la base de datos de la entidad que represente a un paciente, ejecutará el *MDE* y obtendrá los archivos necesarios para incluir al nuevo PSM en el middleware, estos archivos deberán ser movidos a la carpeta raíz del proyecto. Todo esto con la ayuda del manual.

CASO DE PRUEBA 02: Control de usuarios
Casos de uso que intervienen: Requisitos funcionales que intervienen: RQF11, RQF13, RQF16, RQF17, RQF19 Requisitos no funcionales que se comprueban: RQFN01, RQFN02, RQFN05, RQFN06, RQFN10, RQFN11
El usuario administrador del middleware, ingresará a la plataforma con sus credenciales y generará dos nuevos usuarios con credenciales de simulación para los sistemas de los PSM, en este caso, se dirigirá a la URL que contiene esta interfaz con sus credenciales, posterior creará un usuario con una el nombre y contraseña, y se asignará el nivel de acceso básico. Finalmente repetirá el mismo proceso para el segundo usuario, pero esta vez con nivel de acceso avanzado.

CASO DE PRUEBA 03: Inicio de sesión usuarios externos
Casos de uso que intervienen: Requisitos funcionales que intervienen: RQF11, RQF13, RQF16, RQF17, RQF19 Requisitos no funcionales que se comprueban: RQFN01, RQFN05, RQFN06, RQFN11
El usuario: Sistemas de los proveedores de servicios médicos, simulará el funcionamiento del middleware haciendo solicitudes a través de un cliente http, en este caso, se dirigirá a la URL de inicio de sesión con sus credenciales, y en el body se le devolverá el token necesario para autenticarse, este token debe ser enviado siempre en los headers para poder acceder a los servicios del middleware.

CASO DE PRUEBA 04: Extracción de datos correspondientes a un paciente
Casos de uso que intervienen: Requisitos funcionales que intervienen: RQF01, RQF02, RQF03, RQF09, RQF10, RQF14, RQF16, RQF19 Requisitos no funcionales que se comprueban: RQFN01, RQFN05, RQFN06, RQFN11
El usuario Sistemas de los proveedores de servicios médicos, simulará el funcionamiento del middleware haciendo solicitudes a través de un cliente http, en este caso, hará una solicitud a la URL, *dominio/fhir/Patient/$id*, donde $id debe ser reemplazado por el id del paciente del cuál se quiere obtener la información, la respuesta debería ser el recurso FHIR de tipo Patient con los datos provistos por la base de datos del ejemplo

CASO DE PRUEBA 05: Extracción de datos correspondientes al historial médico de un paciente

Casos de uso que intervienen: Requisitos funcionales que intervienen: RQF01, RQF02, RQF04, RQF05, RQF06, RQF14, RQF16, RQF19 Requisitos no funcionales que se comprueban: RQFN01, RQFN05, RQFN06, RQFN11
El usuario Sistemas de los proveedores de servicios médicos, simulará el funcionamiento del middleware haciendo solicitudes a través de un cliente http, en este caso, hará una solicitud a la URL, *dominio/fhir/Condition?patient:Patient=$id&history=true*, donde *$id* debe ser reemplazado por el id del paciente del cuál se quiere obtener la información, la respuesta debería ser el recurso FHIR de tipo Bundle con los datos provistos por la base de datos del ejemplo.

CASO DE PRUEBA 06: Extracción de datos correspondientes al historial médico de un paciente con errores en el funcionamiento
Casos de uso que intervienen: Requisitos funcionales que intervienen: RQF01, RQF02, RQF04, RQF05, RQF06, RQF14, RQF16, RQF19 Requisitos no funcionales que se comprueban: RQFN01, RQFN04, RQFN05, RQFN06, RQFN11
El usuario Sistemas de los proveedores de servicios médicos, simulará el funcionamiento del middleware haciendo solicitudes a través de un cliente http, en este caso, hará una solicitud a la URL, dominio/fhir/Patient/$id, donde $id debe ser reemplazado por el id del paciente del cuál se quiere obtener la información, en este caso el acceso a la base de datos será removido, para probar la resiliencia al error, la respuesta debería ser el error de que la no se encuentra el objeto en la base de datos consultada.

CASO DE PRUEBA 07: Extracción de datos correspondientes a los signos vitales de un paciente

Casos de uso que intervienen: Requisitos funcionales que intervienen: RQF01, RQF02, RQF08, RQF09, RQF10, RQF14, RQF16, RQF19 Requisitos no funcionales que se comprueban: RQFN01, RQFN05, RQFN06, RQFN11
El usuario Sistemas de los proveedores de servicios médicos, simulará el funcionamiento del middleware haciendo solicitudes a través de un cliente http, en este caso, hará una solicitud a la URL, *dominio/fhir/Observation?subject:Patient=$id&identifier=$ident*, donde *$id* debe ser reemplazado por el id del paciente del cuál se quiere obtener la información, mientras que, *$ident* debe ser reemplazado por el identificador de los signos vitales en cuestión, la respuesta debería ser el recurso FHIR de tipo Bundle con los datos provistos por la base de datos del ejemplo.

CASO DE PRUEBA 08: Extracción de datos correspondientes a las prescripciones de los pacientes
Casos de uso que intervienen:

Requisitos funcionales que intervienen: RQF01, RQF02, RQF07, RQF09, RQF10, RQF14, RQF16, RQF19 Requisitos no funcionales que se comprueban: RQFN01, RQFN05, RQFN06, RQFN11
El usuario Sistemas de los proveedores de servicios médicos, simulará el funcionamiento del middleware haciendo solicitudes a través de un cliente http, en este caso, hará una solicitud a la URL, *dominio/fhir/MedicationAdministration?patient:Patient=$id&idPrescription=$ident*, donde *$id* debe ser reemplazado por el id del paciente del cuál se quiere obtener la información, mientras que, *$ident* debe ser reemplazado por el identificador de la prescripción en cuestión, la respuesta debería ser el recurso FHIR de tipo Patient con los datos provistos por la base de datos del ejemplo.

4.3.2. Documentos para la capacitación

El documento de capacitación para los usuarios desarrolladores de los sistemas de los proveedores de servicios médicos se encuentra en el anexo 3.

El documento de capacitación para los usuarios Administradores del middleware se encuentra en el anexo 4.

4.3.3. Documento para la evaluación

El documento en el anexo 5 se usará para evaluar y verificar la aceptación del middleware con los usuarios desarrolladores de los sistemas de los proveedores de servicios médicos.

Por otro lado, tenemos la encuesta dirigida a los usuarios administradores del middleware en el anexo 6.

4.3.4. Encuestas

En esta sección se presentan todas las encuestas generadas para la recopilación de datos con el objetivo de verificar la aceptación y adaptación del middleware.

- **Encuesta #1:** Esta encuesta es para todos los usuarios de tipo Administrador del middleware.

Encuesta para verificar la aceptación y adaptación del middleware - Administrador	
Nombre:	
Cargo:	
Para las siguientes preguntas asigne un valor de 1 al 5 según corresponda, donde 1 significa totalmente en desacuerdo y 5 totalmente de acuerdo	
Generación automática de archivos	
1. Al momento de generar los archivos, estos cumplen satisfactoriamente con la estructura objetivo	
2. Incluir los archivos generados, es suficiente para habilitar la extracción de datos de la organización médica en cuestión	
3. Generar los archivos, me ha parecido un proceso fácil e intuitivo dados mis conocimientos en desarrollo	
Control y gestión de usuarios	
4. El acceso a la herramienta, es simple e intuitivo	
5. La interfaz provista para el control y gestión de los usuarios es intuitiva y fácil de usar.	
6. La creación de usuarios se puede realizar de manera satisfactoria	
7. La eliminación de usuarios se puede realizar de manera satisfactoria	
8. La lectura de usuarios se puede realizar de manera satisfactoria	
9. La asignación de roles de usuarios se puede realizar de manera satisfactoria	
10. La edición de roles de usuarios se puede realizar de manera satisfactoria	

- **Encuesta #2:** Esta encuesta está dirigida para todos los usuarios de tipo Desarrolladores de los PSM

Encuesta para verificar la aceptación y adaptación del middleware - Desarrollador	
Nombre:	
Cargo:	

Para las siguientes preguntas asigne un valor de 1 al 5 según corresponda, donde 1 significa totalmente en desacuerdo y 5 totalmente de acuerdo	
Acceso al middleware	
1. Al iniciar sesión, la contraseña y usuario son suficientes para autenticarme en el middleware y realizar las consultas	
2. El token obtenido en la respuesta me permite autenticarme en el middleware	
Funcionalidades de conversión y extracción	
3. Los datos obtenidos al realizar la petición son consistentes con los almacenados en la base de datos del PSM	
4. La respuesta de la conversión cumple satisfactoriamente con los lineamientos de XML o JSON	
5. La respuesta de la conversión sigue satisfactoriamente el estándar FHIR	
6. La URL consultada sigue las convenciones sugeridas por FHIR	
7. El resultado obtenido para cada una de las consultas es consistente con lo solicitado a través de la solicitud GET	
8. El error mostrado al momento de fallo es legible y fácil de entender	
9. Cuando un error se suscita el sistema no detiene su funcionamiento y sigue en espera de más solicitudes	
10. La extracción de datos correspondientes a un paciente es correcta y mantiene todos los datos del PSM	
11. La extracción de datos correspondientes al historial médico de un paciente es correcta y mantiene todos los datos del PSM	
12. La extracción de datos correspondientes a los signos vitales de un paciente es correcta y mantiene todos los datos del PSM	
13. La extracción de datos correspondientes a las prescripciones de los pacientes es correcta y mantiene todos los datos del PSM	
14. Ingrese el TTD calculado para la pregunta 1	
15. Ingrese el TTD calculado para la pregunta 2	
16. Ingrese el TTD calculado para la pregunta 3	
17. Ingrese el TTD calculado para la pregunta 4	

4.4. Diseño experimental

Para la parte del diseño experimental se han seleccionado, el diseño interindividual y el diseño intraindividual.

Para el caso de la intraindividual, dado que, lo que se requiere es obtener las respuestas a cada PIE planteadas, de manera que, se pueda generalizar la conclusión obtenida de este cuasi-experimento.

Por otro lado, el middleware creado está enfocado a ser utilizado por varias personas una vez que entre en producción, por lo que, es de mayor importancia determinar los resultados en diferentes individuos.

Mientras que, para el caso de la interindividual, se requiere obtener las respuestas de los sujetos con respecto a su ocupación, es decir, por un lado, los desarrolladores de software y por otro lado el personal médico capacitado.

Cabe recalcar que, el recurso seleccionado será tomado de manera aleatoria para evitar cualquier sesgo a la hora de obtener los resultados, dado que, la cantidad de recursos convertida por el middleware es muy amplia, se considera inviable realizar el cuasi-experimento con todas las entidades obtenidas.

4.5. Procedimiento experimental

Para el procedimiento experimental se decidió hacer uso del modelo de evaluación de Metamodelo de ingeniería de procesos de software (SPEM, por sus siglas en inglés), dada su relación y amplio uso dentro de la rama de desarrollo de software y su evaluación. El resumen del modelo se puede observar en la figura 4.1

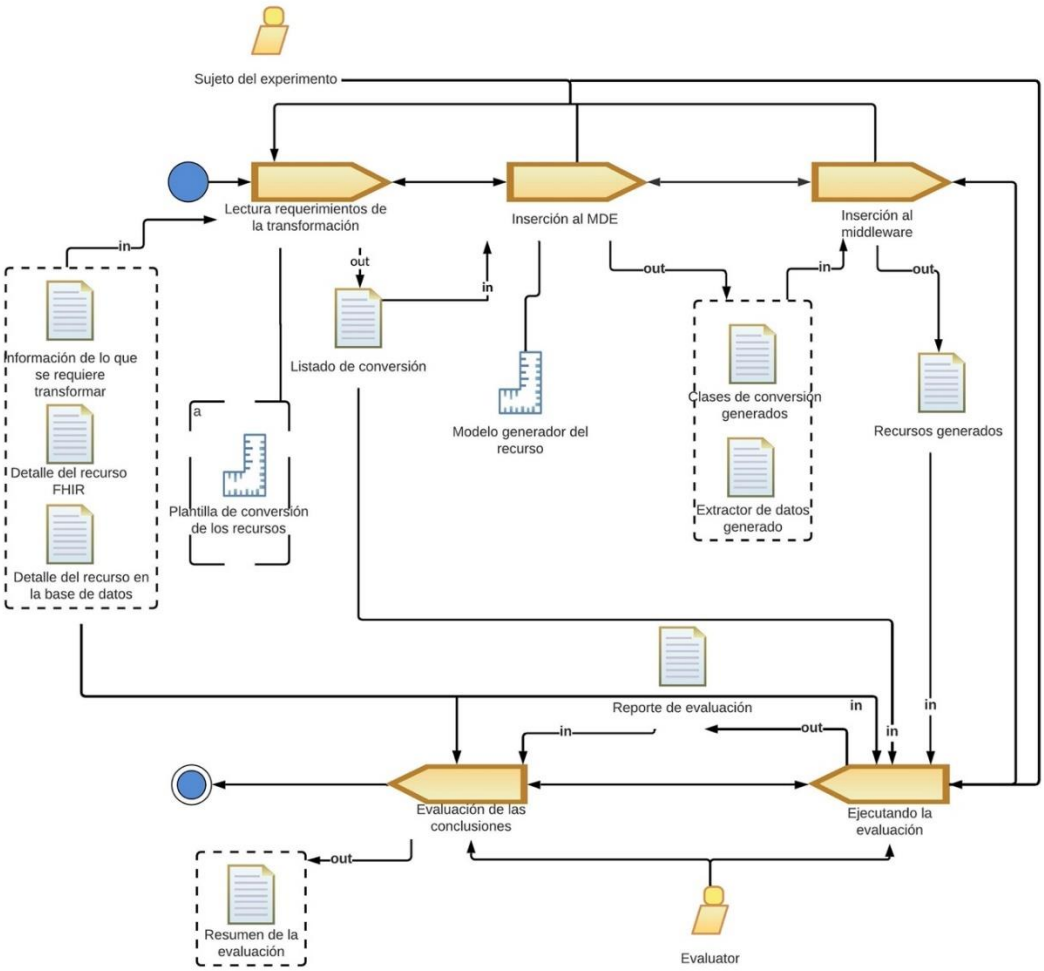


Figura 4.1: Modelo SPEM para la evaluación. Fuente: Elaboración propia.

4.5.1. Lectura de requerimientos de la transformación

En esta fase se realiza la lectura de cada uno de los documentos de entrada, que son:

- **Información de lo que se requiere transformar:** en este documento se encuentra una breve descripción de lo que es FHIR y su uso, además se encuentra la descripción del ejercicio a ejecutarse.
- **Detalle del recurso FHIR:** en este documento se muestra la tabla con la descripción del recurso FHIR en cuestión, como se indicó previamente en el ejemplo

- **Detalle del recurso en la base de datos:** este documento contiene la descripción del recurso en cuestión, pero con respecto a la base de datos del proveedor de servicios médicos en específico.

Además, se provee de una plantilla de conversión donde se irá alineando cada uno de los atributos de las clases de las bases de datos con un listado legible para el ser humano.

Esta fase tiene como salida el siguiente documento:

- **Listado de conversión:** este documento contiene un listado de la traslación de los atributos de las clases a un lenguaje natural.

4.5.2. Inserción al MDE

En esta fase se genera cada uno de los archivos requeridos por el middleware para incluir a un nuevo proveedor de servicios médicos a la conversión.

Para ello, se decidió usar el enfoque de ingeniería basada en modelos (MDE, por sus siglas en inglés). Aquí se deberá ingresar la información del listado de conversión dentro del modelo generado para el recurso.

Esta fase tiene como entrada el documento de listado de conversión y como salida a:

- **Clases de conversión generadas:** Las clases necesarias para el funcionamiento del middleware
- **Extractor de datos generado:** Una clase que tiene la especificación de los *endpoints* de donde serán extraídos los datos

4.5.3. Inserción al middleware

Esta fase tiene como entrada a los documentos salida de la fase de inserción de MDE, los cuales serán ubicados dentro del middleware para su prueba.

Una vez realizado esto se procederá a acceder a la *URL* que obtenga el recurso en mención, recordando lo mencionado en la especificación de FHIR. Este paso nos dará el archivo con el recurso generado a través del middleware, que es el único documento de salida.

4.5.4. Ejecutando la evaluación

En esta fase se ejecuta la comparación y se aplican las métricas descritas con anterioridad para obtener el resultado de la eficiencia de la herramienta.

Dado que, la métrica es una ponderación se ha considerado que lo obtenido debe ser interpretado como sigue:

- Menor o igual a 1: la herramienta ha convertido satisfactoriamente todos los atributos de la clase en mención a atributos del recurso FHIR.
- Menor o igual a 0.8: la herramienta ha convertido de manera aceptable todos los atributos de la clase en mención a atributos del recurso FHIR.
- Menor o igual a 0.6: la herramienta ha convertido pobremente todos los atributos de la clase en mención a atributos del recurso FHIR
- Menor o igual a 0.4: la herramienta tiene deficiencias para convertir todos los atributos de la clase en mención a atributos del recurso FHIR

Los documentos de entrada para esta fase son:

- Información de lo que se requiere transformar
- Detalle del recurso FHIR
- Detalle del recurso en la base de datos
- Listado de conversión
- Recursos generados

Mientras que, la salida es el documento:

- **Reporte de evaluación**: documento que contiene las respuestas del sujeto del cuasi-experimento para cada una de las PIE.

4.5.5. Evaluación de las conclusiones

Esta fase tiene como entrada los documentos:

- Información de lo que se requiere transformar
- Detalle del recurso FHIR
- Detalle del recurso en la base de datos
- Reporte de evaluación

Además, en esta fase se realiza la evaluación de cada uno de los reportes de evaluación con el fin de obtener una conclusión general y responder cada una de las PIE con sus respectivas hipótesis. Además, como salida se obtiene el resumen de la evaluación.

4.6. Definición y selección de la muestra

La muestra serán dos grupos de usuarios: desarrolladores de software y administradores del middleware.

Dado que, lo que se requiere medir se puede incluir dentro del término de usabilidad se puede realizar con al menos 5 personas por grupo según Jakob Nielsen (1994), sin embargo, este cuasi-experimento busca una muestra de al menos 10 participantes para el grupo de los desarrolladores y 5 para el grupo de los administradores.

Este tipo de muestra es denominado por conveniencia dado que, no existen criterios que deban tenerse en cuenta para que los participantes sean o no considerados dentro de la muestra, además decrementará los tiempos y costos que se generan al momento de crear estas muestras.

4.7. Pruebas de funcionalidad

Para la obtención de estos resultados se realizaron los experimentos con dos grupos de sujetos.

Figura 4.2. Primer grupo de sujetos, médicos del Laboratorio de telemedicina e ingenieros de la Universidad de Cuenca

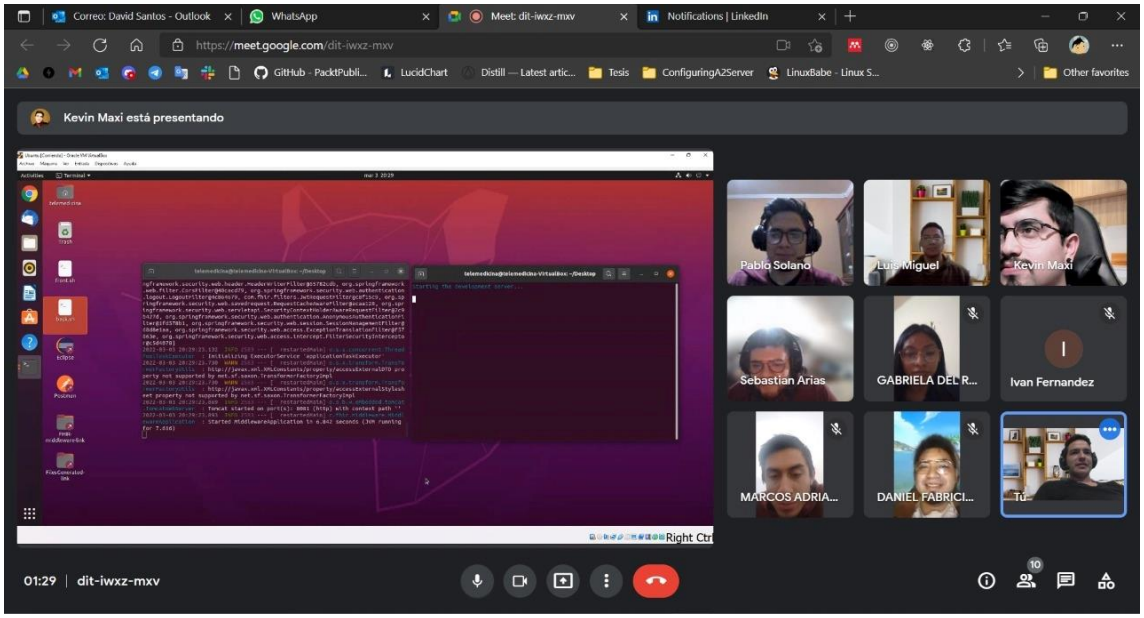


Figura 4.3 Segundo grupo de sujetos estudiantes de ingeniería, ingenieros y técnicos en informática. (I)

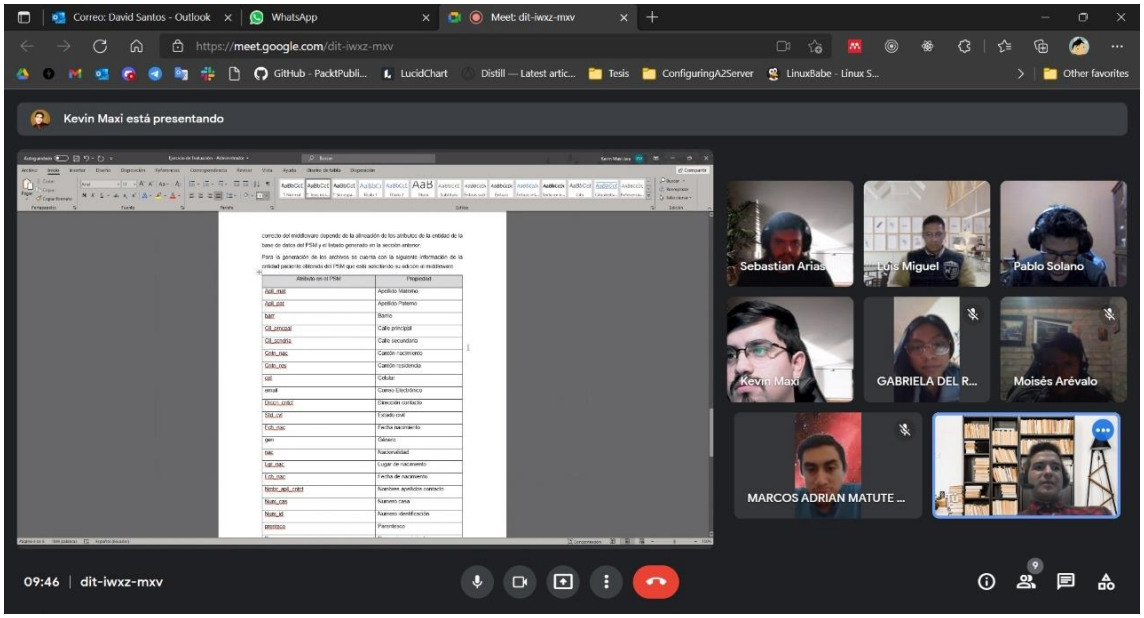


Figura 4.4 Segundo grupo de sujetos estudiantes de ingeniería, ingenieros y técnicos en informática. (II)

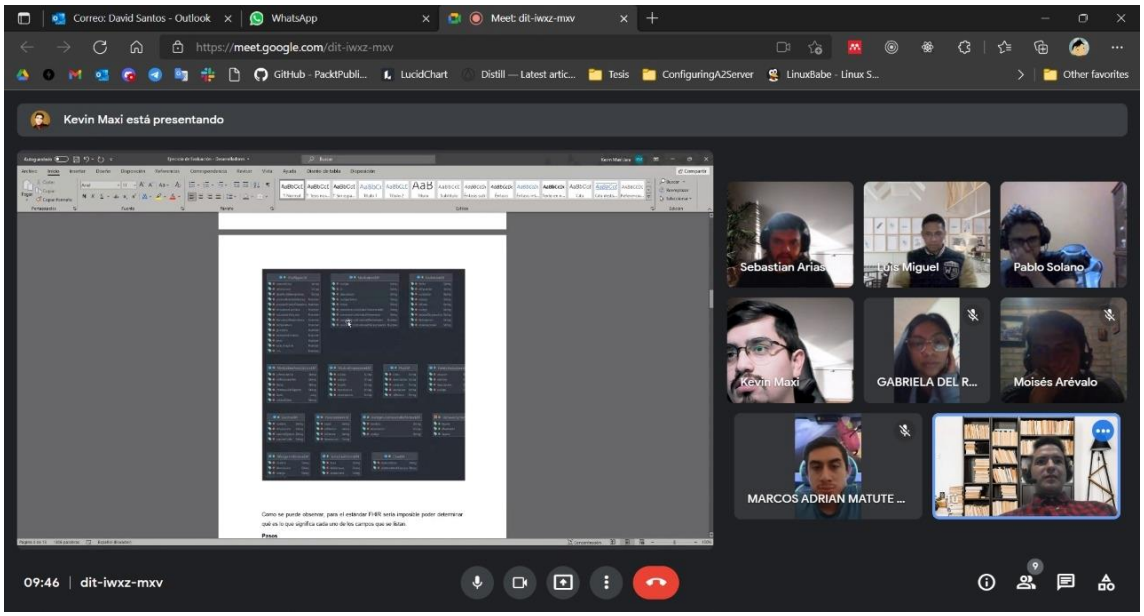


Figura 4.5 Segundo grupo de sujetos estudiantes de ingeniería, ingenieros y técnicos en informática. (III)

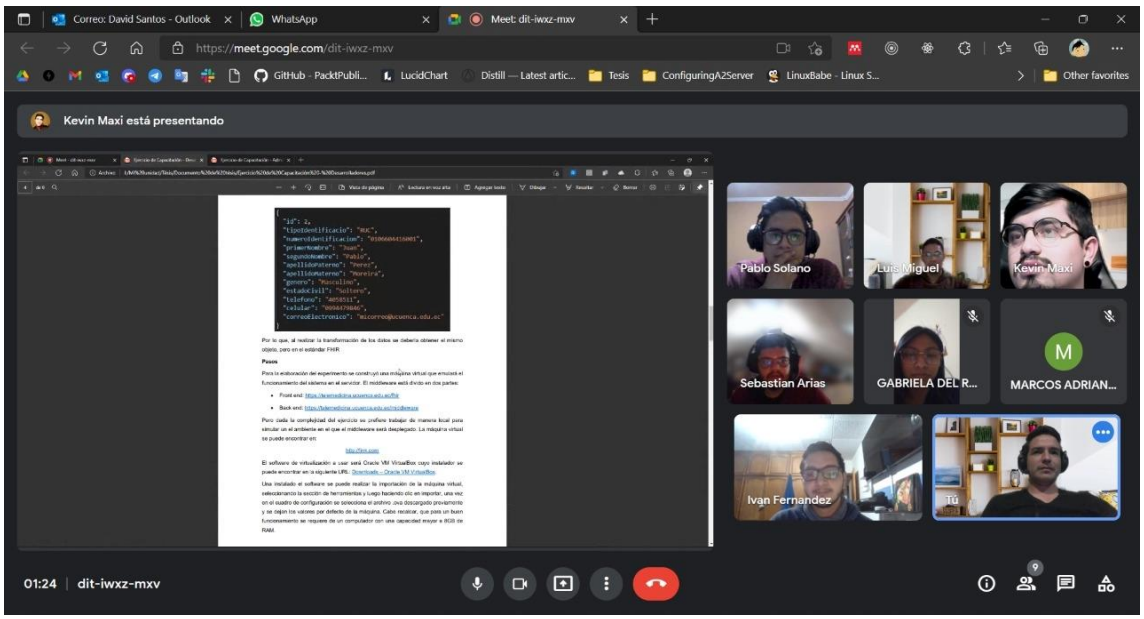


Figura 4.6 Segundo grupo de sujetos estudiantes de ingeniería, ingenieros y técnicos en informática. (IV)

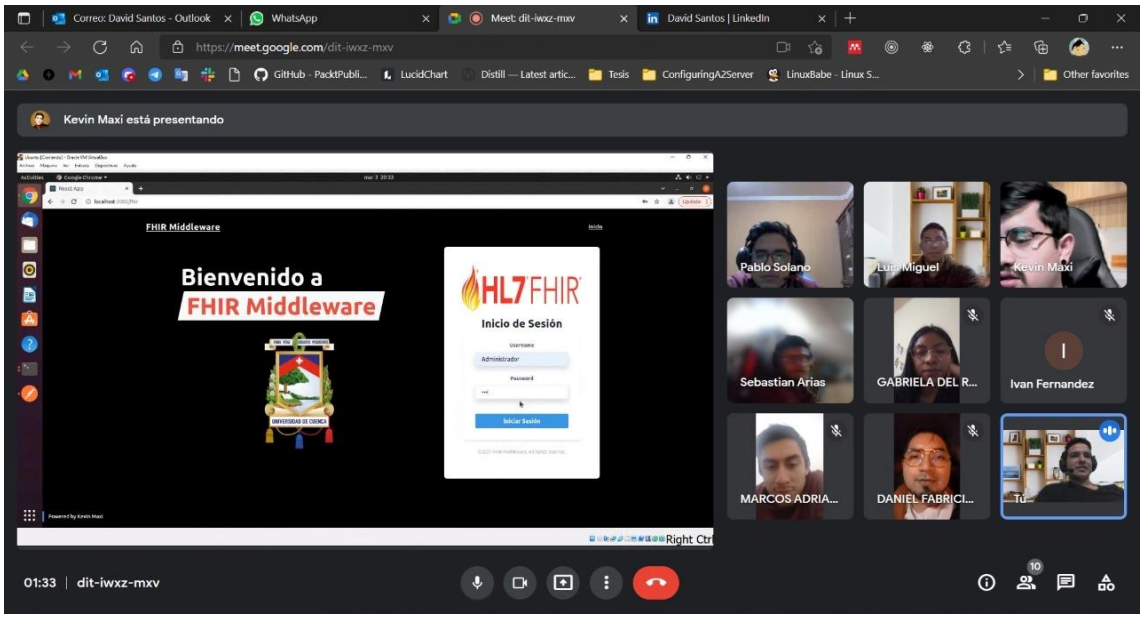


Figura 4.7 Segundo grupo de sujetos estudiantes de ingeniería, ingenieros y técnicos en informática. (V)

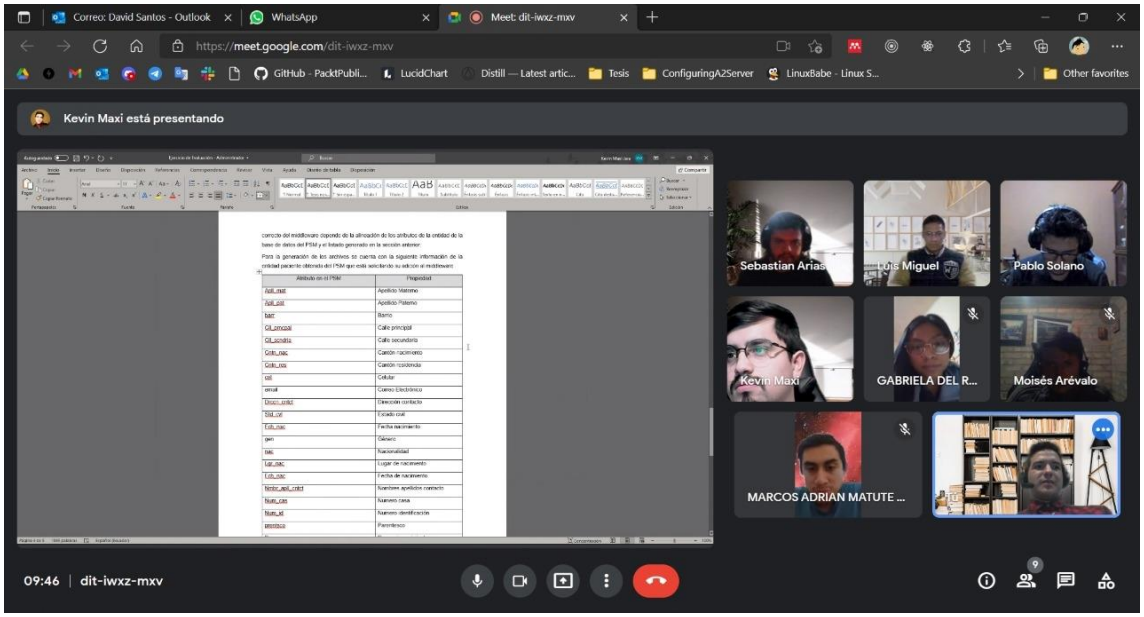


Figura 4.8 Segundo grupo de sujetos estudiantes de ingeniería, ingenieros y técnicos en informática. (VI)

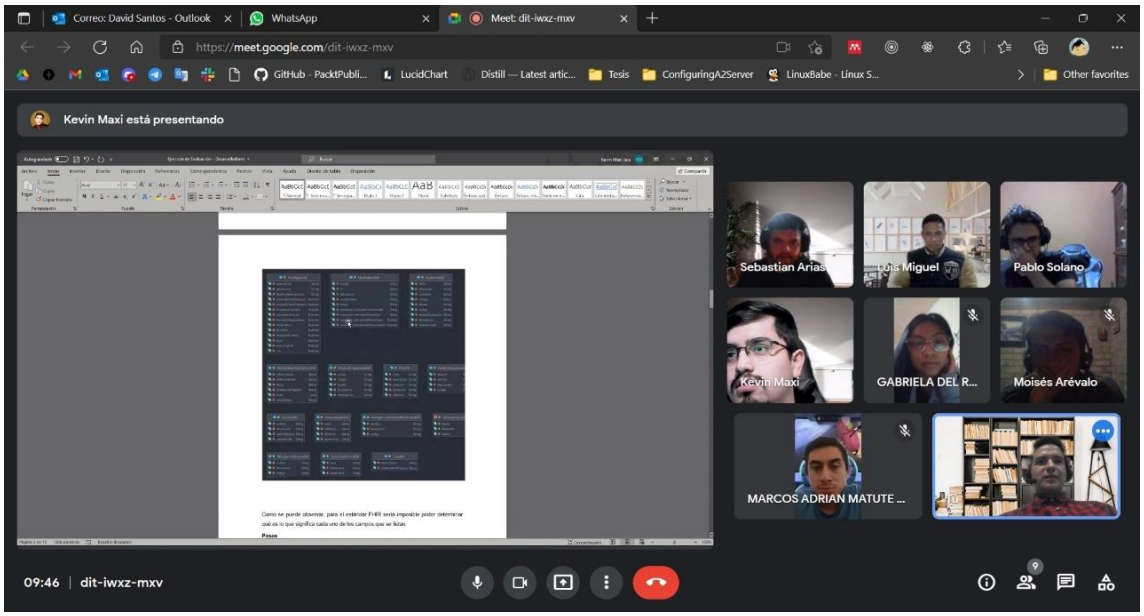


Figura 4.9 Segundo grupo de sujetos estudiantes de ingeniería, ingenieros y técnicos en informática. (VII)

4.8. Resultados de las pruebas de funcionalidad

En las siguientes dos tablas se presentan los resultados obtenidos al realizar las encuestas a cada uno de los participantes.

Dentro de estas tablas se muestran: los tiempos que tomo responder cada pregunta, así como, el promedio de todos los participantes para cada una de ellas. Finalmente, en la columna de total se presenta el tiempo total que le tomo realizar la encuesta a cada uno de los participantes

Tabla 11. Resultados de tiempos en encuesta desarrolladores. Fuente: Elaboración propia

Encuesta para los usuarios desarrolladores de los sistemas de los PSM						
Cantidad de personas encuestadas					10	
	Ejercicio 1	Ejercicio 2	Ejercicio 3	Ejercicio 4	Ejercicio 5	Total
Desarrollador 1	1m02s	0m51s	1m39s	1m28s	9m41s	14m41s
Desarrollador 2	0m54s	0m59s	1m24s	1m52s	9m30s	14m39s
Desarrollador 3	1m08s	1m10s	1m02s	2m04s	9m53s	15m17s
Desarrollador 4	1m16s	1m09s	1m00s	1m11s	9m05s	13m41s
Desarrollador 5	1m12s	1m22s	1m36s	2m01s	10m54s	17m05s
Desarrollador 6	1m12s	1m17s	2m10s	1m01s	9m40s	15m20s
Desarrollador 7	1m11s	1m08s	2m03s	1m17s	10m22s	16m01s
Desarrollador 8	1m24s	1m12s	1m31s	2m07s	10m50s	17m04s
Desarrollador 9	1m36s	1m05s	2m06s	2m09s	9m37s	16m33s
Desarrollador 10	1m30s	1m02s	2m50s	1m14s	10m51s	17m27s
Promedio	1m15s	1m08s	1m44s	1m36s	10m02s	15m45s

Tabla 12. Resultados de tiempos en encuesta administradores. Fuente: Elaboración propia

Encuesta para los usuarios administradores del middleware					
Cantidad de personas encuestadas				5	
	Ejercicio 1	Ejercicio 2	Ejercicio 3	Ejercicio 4	Total
Administrador 1	0m35s	0m20s	0m17s	12m13s	13m25s
Administrador 2	0m37s	0m26s	0m35s	14m18s	15m56s
Administrador 3	0m38s	0m55s	0m35s	16m21s	18m29s
Administrador 4	0m46s	1m15s	0m58s	16m47s	19m46s
Administrador 5	0m50s	0m55s	0m44s	16m25s	18m54s
Administrador 6	1m17s	0m56s	0m18s	15m53s	17m24s
Administrador 7	1m17s	0m42s	0m13s	18m04s	20m14s
Administrador 8	1m18s	0m32s	0m16s	16m34s	18m40s
Promedio	0m55s	0m45s	0m30s	15m48s	17m58s

Determinados los resultados de los tiempos de las encuestas, en las tablas 13 y 14 se presentan los resultados obtenidos para cada una de las encuestas propuestas en este documento.

Las preguntas corresponden a cada fila mientras las columnas corresponden a cada uno de los participantes, en la columna denominada promedio se presenta el resultado promedio obtenido por pregunta.

Dado que, cada una de las preguntas están en sentido positivo, el objetivo es obtener el mayor puntaje posible para cada una de las preguntas, mientras que, lo ideal

para las preguntas de la Encuesta para los usuarios desarrolladores de los sistemas de los PSM desde la 14 hasta la 17 es obtener el puntaje más cercano a 1.

Tabla 13. Resultados encuesta desarrolladores. Fuente: Elaboración propia

Encuesta para los usuarios desarrolladores de los sistemas de los PSM											
Cantidad de personas encuestadas											10
	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	Promedio
Pregunta 1	5	4	5	5	5	5	5	5	5	3	4,7
Pregunta 2	5	4	5	5	5	5	5	5	5	5	4,9
Pregunta 3	5	5	5	5	5	5	5	5	4	5	4,9
Pregunta 4	5	4	5	5	4	5	5	5	5	5	4,8
Pregunta 5	4	4	5	5	5	5	5	5	3	5	4,6
Pregunta 6	3	3	5	5	5	5	5	5	5	5	4,6
Pregunta 7	5	5	5	5	5	5	5	5	5	5	5
Pregunta 8	5	5	5	5	5	4	5	5	5	3	4,7
Pregunta 9	5	5	5	5	4	4	5	5	5	5	4,8
Pregunta 10	4	4	5	5	5	4	5	5	4	5	4,6
Pregunta 11	5	4	5	5	5	5	5	5	4	5	4,8
Pregunta 12	4	4	5	5	5	5	5	5	5	5	4,8
Pregunta 13	4	5	5	5	5	4	5	5	4	4	4,6
Pregunta 14	1	1	0,9	1	1	1	1	1	1	1	0,99
Pregunta 15	1	1	0,9	1	1	1	1	1	1	1	0,99
Pregunta 16	1	1	0,9	1	1	1	1	1	1	1	0,99
Pregunta 17	1	1	0,9	1	1	1	1	1	1	1	0,99

Tabla 14. Resultados encuesta administradores. Fuente: Elaboración propia

Encuesta para los usuarios administradores del middleware									
Cantidad de personas encuestadas									8
	A1	A2	A3	A4	A5	A6	A7	A8	Promedio
Pregunta 1	5	5	5	5	5	5	5	5	5
Pregunta 2	5	5	5	5	5	5	4	5	4,9
Pregunta 3	5	4	4	5	4	5	5	5	4,6
Pregunta 4	5	5	4	5	4	5	4	4	4,5
Pregunta 5	5	4	5	5	3	5	5	5	4,6
Pregunta 6	5	4	4	5	5	5	5	5	4,8
Pregunta 7	5	5	4	5	5	5	5	5	4,9
Pregunta 8	5	5	5	5	5	5	4	4	4,8
Pregunta 9	5	5	4	5	5	5	5	5	4,9
Pregunta 10	5	5	5	5	5	5	5	5	5

Los resultados de los tiempos en la tabla de resultados de desarrolladores nos muestran un tiempo promedio de 17 minutos con 58 segundos, lo que nos indica que para el usuario desarrollador le toma alrededor de este tiempo entender toda la estructura de cómo se maneja la conversión a FHIR.

Además, en este tiempo logra obtener todas las respuestas que el middleware provee por lo que, a partir de ese tiempo se puede decir que este usuario ha experimentado con todas las funcionalidades disponibles para él.

En cuanto a los resultados de tiempo obtenidos para el tipo de usuario administrador del middleware, el tiempo promedio nos muestra que le tomaría alrededor de 15 minutos con 45 segundos añadir por completo un nuevo PSM al middleware, que es un tiempo relativamente corto en comparación con el tiempo que un proceso manual de creación de clases le tomaría

4.9. Análisis de los resultados

4.9.1. Prueba de normalidad de Shapiro-Wilk

Para determinar si la distribución de nuestras variables tiene una distribución normal, es necesario aplicar una prueba.

Dada que, la cantidad de muestra en este caso es menor a 50 la prueba a aplicar es la prueba de normalidad de Shapiro-Wilk.

Esta prueba determina si la hipótesis nula proviene de una distribución normal, de Poisson o exponencial, pero para este caso contrastaremos solamente con la normal. Al insertar los datos en el software SPSS® v20. Y al aplicar la prueba se obtienen los siguientes resultados donde PA son las siglas de pregunta de administrador. Además, se establece un valor de p = 0,05

Tabla 15. Prueba de Shapiro-Wilk para la encuesta de administrador. Fuente: Elaboración Propia

		PA1	PA2	PA3	PA4	PA5	PA6	PA7	PA8	PA9	PA10
N		8	8	8	8	8	8	8	8	8	8
Parámetros normales[a,b]	Media	5,000	4,875	4,625	4,500	4,625	4,750	4,875	4,750	4,875	5,000
	Desviación típica	,0000[c]	,3536	,5175	,5345	,7440	,4629	,3536	,4629	,3536	,0000[c]
Diferencias más extremas	Absoluta		,513	,391	,325	,443	,455	,513	,455	,513	
	Positiva		,362	,261	,325	,307	,295	,362	,295	,362	
	Negativa		-,513	-,391	-,325	-,443	-,455	-,513	-,455	-,513	
Z de Shapiro-Wilk			1,451	1,105	,920	1,253	1,288	1,451	1,288	1,451	
Sig. asintót. (bilateral)			,030	,174	,366	,087	,072	,030	,072	,030	

a. La distribución de contraste es la Normal.

b. Se han calculado a partir de los datos.

c. La distribución no tiene varianza para las variables P1 y P10. No es posible realizar la prueba de Shapiro-Wilk para una muestra.

Tabla 16. Prueba de Shapiro-Wilk para la encuesta de desarrollador. Fuente: Elaboración propia

		PD1	PD2	PD3	PD4	PD5	PD6	PD7	PD8	PD9	PD10	PD11	PD12	PD13	PD14	PD15	PD16	PD17
N		10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
Parámetros normales[a,b]	Media	4,700	4,900	4,900	4,800	4,600	4,600	5,000	4,700	4,800	4,600	4,800	4,800	4,600	,990	,990	,990	,990
	Desviación típica	,6749	,3162	,3162	,4216	,6992	,8433	,0000[c]	,6749	,4216	,5164	,4216	,4216	,5164	,0316	,0316	,0316	,0316
Diferencias más extremas	Absoluta	,472	,524	,524	,482	,416	,482		,472	,482	,381	,482	,482	,381	,524	,524	,524	,524
	Positiva	,328	,376	,376	,318	,284	,318		,328	,318	,277	,318	,318	,277	,376	,376	,376	,376
	Negativa	-,472	-,524	-,524	-,482	-,416	-,482		-,472	-,482	-,381	-,482	-,482	-,381	-,524	-,524	-,524	-,524
Z de Shapiro-Wilk		1,491	1,657	1,657	1,525	1,317	1,525		1,491	1,525	1,204	1,525	1,525	1,204	1,657	1,657	1,657	1,657
Sig. asintót. (bilateral)		,023	,008	,008	,019	,062	,019		,023	,019	,110	,019	,019	,110	,008	,008	,008	,008

a. La distribución de contraste es la Normal.

b. Se han calculado a partir de los datos.

c. La distribución no tiene varianza para esta variable. No es posible realizar la prueba de Shapiro-Wilk para una muestra.

Como se puede ver en la tabla 15, las preguntas de administrador desde la 3 a la 8 tienen una distribución normal, es decir superan el valor de p propuesto mientras que, las preguntas 2 y 9 no superan el valor, por lo que, se aplicó la prueba de Wilcoxon con un valor de prueba de 3, para mantener el valor neutro de la escala de Likert aplicada.

Por otro lado, en la tabla 16, se presentan los resultados obtenidos para las preguntas de desarrollador, donde las preguntas: 5, 10, y 13 tienen una distribución normal para las preguntas restantes, al igual que con la encuesta de administrador, se aplicó la prueba de Wilcoxon con el valor de prueba de 3.

Es por esto que, con los resultados obtenidos y al analizarlos se puede concluir que:

La hipótesis *H0: Un middleware informático basado en tecnologías web y en estándares internacionales HL7 no permite la interoperabilidad correcta de diferentes sistemas de información médica.* Se rechaza dadas las siguientes respuestas a las preguntas de investigación obtenidas:

- **PIE1: ¿El middleware desarrollado es capaz de transformar los datos al estándar FHIR de modo que, se habilite la interoperabilidad?**

Los resultados obtenidos por las preguntas 3, 4, 6,7, 10, 11, 12, 13, 14, 15, 16 y 17 de la encuesta dirigida a los desarrolladores de sistemas del PSM, permiten rechazar la hipótesis $H1_0$ y asegurar que el middleware desarrollado es capaz de transformar los datos al estándar FHIR, habilitando, de este modo, la interoperabilidad

- **PIE2: ¿El middleware desarrollado es capaz de realizar las transformaciones de datos sin importar la estructura interna de cada base de datos?**

Los resultados obtenidos en las preguntas 14, 15, 16 y 17 de la encuesta dirigida a los desarrolladores de sistemas del PSM permiten rechazar $H2_0$ y asegurar que el middleware desarrollado es capaz de realizar las transformaciones de datos sin importar la estructura interna de cada base de datos. Esto además se corrobora con la prueba de Shapiro-Wilk lo que indica que el middleware usado si influye al momento de convertir los datos al estándar FHIR.

- **PIE3: ¿El middleware desarrollado cumple con los requisitos planteados en esta tesis?**

Los promedios de cada pregunta de las encuestas superan los 4 puntos de satisfacción y los resultados obtenidos en las tablas 15 y 16 permiten rechazar la hipótesis $H3_0$ lo que indica

un alto grado de aceptación por parte de los usuarios encuestados, además nos indica que el mismo cumple con los requisitos planteados y desarrollados para la construcción del middleware.

Por otro lado, el middleware es capaz de adaptarse a cualquier tipo de PSM, que es uno de los objetivos principales del middleware, ya que es un punto central para la interoperabilidad

Por todo esto se puede afirmar que el middleware cumple con los requisitos planteados para esta tesis.

4.10. Análisis de rendimiento de usuario

Por otro lado, también se decidió medir el tiempo que cada uno de estos usuarios utiliza para cumplir las tareas planteadas.

Las preguntas planteadas conjuntamente con el tiempo medido, permiten calcular el rendimiento del usuario en términos de eficiencia y efectividad.

Para el caso de los desarrolladores las preguntas 14, 15, 16, y 17 permiten medir la relación que existe entre la conversión de datos existentes con lo extraído por el middleware.

Hay que recordar que un valor igual a 1 en esta métrica nos indica que todos los datos existentes han sido convertidos al estándar FHIR.

Al tomar el valor promedio obtenido en estas 4 preguntas del 0.99 y el tiempo promedio para completar el ejercicio de evaluación de 15 minutos con 45 segundos, se puede afirmar que la efectividad del usuario es del 99% mientras que, la eficiencia varía entre 13 minutos 41 segundos y 17 minutos 37 segundos.

Finalmente, hay que recalcar que los tiempos fueron solicitados al terminar de leer la indicación del ejercicio en cuestión. Por lo que, estos tiempos solo incluyen la ejecución de la tarea sumado el tiempo de espera del sistema.

Tabla 16. Resumen efectividad y eficiencia. Fuente: Elaboración propia

Variable	**Min**	**Max**	**Media**	**Desv. tip.**
Efectividad	0,9	1	0,99	0,0316
Eficiencia	5	5	5	0,0316

4.11. Amenazas a la validez

Dada la naturaleza del cuasi-experimento planteado es necesario presentar las amenazas que podrían afectar la validez de los resultados antes presentados.

Para representar estas amenazas se ha decidido dividirlas en tres: (i) *Validez interna,* que, es la que está relacionada con el middleware en sí, es decir, con las interfaces construidas, bases de datos de los PSM y con los estándares usados; (ii) *Validez externa,* la cual permite analizar la generalización de hallado y obtenido al ejecutar las evaluaciones; y (iii) *Validez de la conclusión,* que finalmente, validará la posibilidad de obtener una conclusión acertada con los resultados obtenidos.

Para la validez interna, hay que tener en cuenta la naturaleza de los participantes del cuasi-experimento y de la muestra. Los participantes son todos relacionadas al área de la informática e ingeniería, por lo que, no es muy común encontrar experiencia con datos médicos entre ellos.

Al iniciar los experimentos se pudo evidenciar la falta de conocimiento en esta área. Esto es algo que se podía imaginar, por lo que, en la fase de capacitación se añadió una introducción a la problemática y estándares relacionados.

Por otro lado, por la naturaleza del área, la variedad de dispositivos tecnológicos, puede resultar en afectaciones para los resultados o incluso llegar a imposibilitarlos.

Por esto, es que se decidió llevar a cabo los experimentos en vivo a través de una conferencia web, con conexiones a internet y con equipos propios y separados.

Para la validez externa, la principal amenaza percibida es la posible falta de entendimiento del dominio del sistema, por esto se planteó ejercicios simples y naturales en el área antes mencionada, además los ejercicios, contaron todo el tiempo con un contraste de resultado exitoso para evitar cualquier falso positivo.

Además, los ejercicios planteados nacieron desde los requerimientos del sistema, es decir, se diseñaron los ejercicios necesarios como para cubrir al menos una vez cada funcionalidad del sistema.

Por último, para la validez de la conclusión, la amenaza de la naturaleza de la muestra es de las principales. Empero, la muestra de sujetos fue seleccionada específicamente del área

necesaria para hacer uso del sistema, esto nos permite validar el sistema con los usuarios potenciales y prácticamente únicos del sistema.

El cumplimiento de estas características permite validar la usabilidad y el cumplimiento de los requisitos planteados de una manera técnica.

Finalmente, este trabajo de titulación dio como resultado un artículo científico que fue parte del *International Conference on Smart Technologies, Systems and Applications* SmartTech-IC 2021 y que fue publicado como parte del libro: *Smart Technologies, Systems and Applications* en *Springer Link (Maxi & Morocho, 2022)* y que se puede encontrar en la siguiente URL.

Integrating Medical Information Software Using Health Level Seven and FHIR: A Case Study | SpringerLink

CAPÍTULO 5: CONCLUSIONES Y TRABAJO FUTURO

En este capítulo se presentan las conclusiones obtenidas al desarrollar el presente trabajo de titulación, finalmente, se plantean los posibles trabajos futuros.

5.1. Conclusiones

La globalización y el constante uso de tecnología en estos días, ha llevado a la industria de la medicina a una constante necesidad de mejora del manejo de la información recolectada por todo este campo. En especial en este caso donde los datos son críticos.

En Ecuador aún se puede visibilizar la falta de interoperabilidad de sistemas de información médica, sin embargo, varios esfuerzos ya se han realizado para intentar alcanzar tan ansiado objetivo. Aunque, se usan ciertos estándares para codificar ciertos ámbitos del área de la salud, se puede evidenciar aún la falta de un estándar. Por esto, el objetivo de este trabajo de titulación ha presentado una solución viable para todos los sistemas de información médica actuales, con el cual sea posible estandarizar los datos médicos encontrados en los formularios 001, 002, 003 y 005.

Este middleware permite reducir los esfuerzos de los desarrolladores de sistemas de información médica al presentar un proceso limpio y relativamente corto para interpretar datos de diferentes instituciones médicas, sin importar el desconocimiento de estructura interna, todo esto a través de las listas de conversión obtenidas por los expertos de cada uno de los sistemas de información médica. Asimismo, con la pandemia se han evidenciado otros beneficios de tener sistemas interoperables, como es el caso de las vacunas, donde con este tipo de sistemas se podría acceder al registro de vacunación de los pacientes, así como a posibles efectos secundarios, información que podría ser relevante a nivel internacional, sobre todo, si se trata de pacientes que cruzan límites internacionales.

Es así, que se cumple con el objetivo general planteado por este trabajo de titulación: *Diseñar e implementar un middleware de información médica usando estándares HL7 y FHIR, que permita la interoperabilidad de sistemas de información médica.* Además, que este middleware abre un camino muy importante para la interoperabilidad de información médica ecuatoriana.

Al desarrollar el middleware se pudo evidenciar la falta de información con respecto al tema. El área es relativamente nueva, con respecto a otros, por lo que, la información encontrada

es muy reducida y prácticamente toda proviene del sitio web de FHIR, sin embargo, su uso es cada vez más extendido, por lo que, es importante estar al tanto de estos estándares.

Se diseñó y desarrolló un *framework* que permite la interoperabilidad en diferentes sistemas de información médica.

Para lograrlo, se hizo uso de los estándares internacionales HL7 y FHIR, como también ciertas lecciones aprendidas de trabajos anteriores relacionados con el área. Todo esto cumple con toda la metodología propuesta, para que los sistemas que forman parte del *framework* puedan intercambiar información entre ellos de manera segura y confiable.

Asimismo, se cumplieron todos los objetivos específicos a través de la metodología presentada en el capítulo 2 de este trabajo de titulación.

Para *OE1: Analizar los requerimientos del software para la correcta construcción del software.* Se cumple en el capítulo 4 específicamente en la sección 3.2, donde se definen todos los requerimientos y diagramas del software. Este objetivo es crucial porque su cumplimiento es la base de todo el middleware.

Para *OE2: Definir la arquitectura FHIR, HL7 y diagramar el software propuesto.* Este objetivo se cumple en el capítulo 4 a partir de la sección 3.3, donde se detalla todo el proceso de definición, diagramación y construcción de la arquitectura y componentes del middleware.

Para *OE3: Implementar el software usando una metodología ágil.* Del mismo, como en los objetivos antes planteados este cumple en el capítulo 4 a partir de la sección 3.6 donde se detalla el proceso al aplicar la metodología propuesta y donde al final del capítulo se obtiene el software completo.

Finalmente, cabe recalcar que, no se ha podido llegar a realizar la integración con *BlockChain* dadas las limitaciones temporales del presente trabajo de titulación y los contratiempos presentados en el trascurso del desarrollo conjunto de ambos proyectos.

5.2. Líneas de trabajo futuro

Sin embargo, al realizar este middleware, al igual que, en otros documentos, se ha encontrado la dificultad existente en la correlación de los datos.

Asimismo, la expresividad de los atributos de los esquemas de base de datos es muy compleja, por lo que, es importante el uso de este middleware, ya que, eliminaría este

inconveniente a través de un mapeo mucho más rígido que permite a los proveedores de servicios médicos trabajar directamente con este marco.

Cabe señalar que, este es el paso previo para un alineamiento de ontologías real, lo que permitiría integrar una mayor expresividad semántica en la interoperabilidad y que sirve como una propuesta de trabajo futuro.

REFERENCIAS

Bender, D., & Sartipi, K. (2013). HL7 FHIR: An agile and RESTful approach to healthcare information exchange. *Proceedings of CBMS 2013 - 26th IEEE International Symposium on Computer-Based Medical Systems*, 326–331. https://doi.org/10.1109/CBMS.2013.6627810

Benson, T., & Grieve, G. (2021). *Principles of Health Interoperability*. Springer International Publishing. https://doi.org/10.1007/978-3-030-56883-2

Benson, T., & Grieve, G. (2021). The benefits of one standard (figure). En *Principles of Health Interoperability*. Springer International Publishing. https://doi.org/10.1007/978-3-030-56883-2

Boussadi, A., & Zapletal, E. (2017). A Fast Healthcare Interoperability Resources (FHIR) layer implemented over i2b2. *BMC Medical Informatics and Decision Making*, *17*(1), 1–12. https://doi.org/10.1186/s12911-017-0513-6

Bresnick, J. (2016). *4 Basics to Know about the Role of FHIR in Interoperability*. Health IT Analytics. https://healthitanalytics.com/news/4-basics-to-know-about-the-role-of-fhir-in-interoperability

Peralta-Velecela, D., Cáceres-Salamea, M. C., Morocho, V. (2021). *nDigital identity Proposal for Unified Medical Record using Blockchain technology*. IEEE Fifth Ecuador Technical Chapters Meeting (ETCM), 2021, 1-6. https://doi.org/10.1109/ETCM53643.2021.9590679.

de Sanidad, M., & Bienestar Social, C. (2020). *CIE • 10 • ES Clasificación Internacional de Enfermedades-10.ª Revisión Sistema de Clasificación de Procedimientos MINISTERIO DE SANIDAD, CONSUMO Y BIENESTAR SOCIAL MINISTERIO DE LA PRESIDENCIA, RELACIONES CON LAS CORTES E IGUALDAD BOLETÍN OFICIAL DEL ESTADO*. https://doi.org/10.ª

Dolin, R. H., Rogers B., & Jaffe, C., (2015). Health Level Seven Interoperability Strategy: Big Data, Incrementally Structured. *Methods of information in medicine*, *54*(01), 75-82. https://doi.org/10.3414/ME14-01-0030

Häyrinen, K., Saranto, K., & Nykänen, P. (2008). Definition, structure, content, use and impacts of electronic health records: A review of the research literature. In *International Journal of Medical Informatics* (Vol. 77, Issue 5, pp. 291–304). https://doi.org/10.1016/j.ijmedinf.2007.09.001

Kruk, M., Gage, A. Arsenault, C., Jordan, K., Leslie, H., Roder-Dewan, S. … Pate, M., (2018). *High-quality health systems in the Sustainable Development Goals era: time for a revolution*. https://doi.org/10.1016/S2214-109X(18)30386-3

HL7.org. (2021). *Introducing HL7 FHIR*. https://www.hl7.org/fhir/summary.html#status

HL7. (2021a). *Asynchronous Request Pattern*.

HL7. (2021b). *Extended Operations on the RESTful API*. https://www.hl7.org/fhir/operations.html

HL7. (2021c). *Profiling FHIR*. https://www.hl7.org/fhir/profiling.html

HL7. (2021d). *RESTful API*. https://www.hl7.org/fhir/http.html

HL7. (2021e). *Search*. https://www.hl7.org/fhir/search.html

ISO. (2010). ISO/TR 20514:2005(en), Health informatics — Electronic health record — Definition, scope and context. *Iso*.

Kamel, P. I., & Nagy, P. G. (2018). Patient-Centered Radiology with FHIR: an Introduction to the Use of FHIR to Offer Radiology a Clinically Integrated Platform. *Journal of Digital Imaging*, *31*(3), 327–333. https://doi.org/10.1007/s10278-018-0087-6

Khvastova, M., Witt, M., Essenwanger, A., Sass, J., Thun, S., & Krefting, D. (2020). Towards Interoperability in Clinical Research - Enabling FHIR on the Open-Source Research Platform XNAT. *Journal of Medical Systems*, *44*(8). https://doi.org/10.1007/s10916-020-01600-y

Kilintzis, V., Chouvarda, I., Beredimas, N., Natsiavas, P., & Maglaveras, N. (2019). Supporting integrated care with a flexible data management framework built upon Linked Data, HL7 FHIR and ontologies. *Journal of Biomedical Informatics*, *94*(October 2018), 103179. https://doi.org/10.1016/j.jbi.2019.103179

Kiourtis, A., Mavrogiorgou, A., Menychtas, A., Maglogiannis, I., & Kyriazis, D. (2019). Structurally Mapping Healthcare Data to HL7 FHIR through Ontology Alignment. *Journal of Medical Systems*,

43(3). https://doi.org/10.1007/s10916-019-1183-y
Leroux, H., Metke-Jimenez, A., & Lawley, M. J. (2017). Towards achieving semantic interoperability of clinical study data with FHIR. *Journal of Biomedical Semantics*, *8*(1), 1–14. https://doi.org/10.1186/s13326-017-0148-7
LinuxForHealth hl7v2-fhir-converter [Source code]. https://github.com/LinuxForHealth/hl7v2-fhir-converter (2020).
Lopez, E., Berlin, M., Stein, R., Cozzi, E., Bermudez, A., Mandirola Brieux, H., Díaz Maffini, M., Moldes, R., Bousquet, T., Mac Culloch, P., Rizzato Lede, D. A., Speranza, C. D., & López Osornio, A. (2020). Results of the use of the teleconsultation platform after 2 months of implementation. *Studies in Health Technology and Informatics*, *270*, 1377–1378. https://doi.org/10.3233/SHTI200450
Maxi, K., Morocho, V. (2022). Integrating Medical Information Software Using Health Level Seven and FHIR: A Case Study. In: Narváez, F.R., Proaño, J., Morillo, P., Vallejo, D., González Montoya, D., Díaz, G.M. (eds) Smart Technologies, Systems and Applications. SmartTech-IC 2021. Communications in Computer and Information Science, vol 1532. Springer, Cham. https://doi.org/10.1007/978-3-030-99170-8_7
Ministerio de Salud Pública. (2011). *Acuerdo Ministerial NO. 00001190*. https://aplicaciones.msp.gob.ec/salud/archivosdigitales/documentosDirecciones/dnn/archivos/ACUERDO+0002+TOMO 04+2011+20191028+0173.pdf
Ministerio de Salud Pública del Ecuador. (2006). Expediente único para la Historia Clínica. *Expediente Para Historia Clinica*, *1*(6), 70. http://www.conasa.gob.ec/biblioteca/Otros/bt15 - Expediente único para la Historia Clínica.pdf
Murray, C. J. L., & Frenk, J. (2000). Theme Papers A framework for assessing the performance of health systems. In *World Health*.
Organización Panamericana de la Salud. (2017). *Agenda de Desarrollo Sostenible para las Américas 2018-2030, Objetivo 6: Fortalecer los sistemas de información para la salud*. https://www.paho.org/hq/index.php?option=com_content&view=article&id=14601:sustainable-health-agenda-for-the-americas-2018-2030-goal-6-information-systems-for-health&Itemid=42350&lang=es
Ryan, A. (2006). Towards Semantic Interoperability in Healthcare: Ontology Mapping from SNOMED-CT to HL7 version 3. In *Proceedings of the second Australasian workshop on Advances in ontologies*. http://dl.acm.org/citation.cfm?id=1273668
Saripalle, R., Runyan, C., & Russell, M. (2019). Using HL7 FHIR to achieve interoperability in patient health record. *Journal of Biomedical Informatics*, *94*. https://doi.org/10.1016/j.jbi.2019.103188
Sharma, M., & Aggarwal, H. (2019). HL-7 Based Middleware Standard for Healthcare Information System: FHIR. In *Lecture Notes in Networks and Systems* (Vol. 46, pp. 889–899). Springer. https://doi.org/10.1007/978-981-13-1217-5_87
Thakkar, M., & Davis, D. C. (2006). Risks, barriers, and benefits of EHR systems: a comparative study based on size of hospital. In *Perspectives in health information management* (Vol. 3). http://www.ncbi.nlm.nih.gov/pubmed/18066363%0Ahttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC2047303

ANEXOS

Anexo 1 – Especificación de Requerimientos:

Revision History

Name	Date	Reason For Changes	Version

1. Introducción

1.1 Propósito

El propósito de esta sección de ERS (Especificación de Requerimientos de Software) es presentar una descripción detallada del proyecto de software: Software integrador de información médica utilizando Health Level Seven y FHIR.
La interoperabilidad entre los sistemas de los diferentes sistemas médicos es un punto crucial para la calidad de la atención médica. Pacientes restringidos a un solo proveedor de servicios médicos, una pobre eficiencia de los recursos y la incapacidad de llegar a una universalidad, son algunos de los diferentes problemas que se dan por una pobre interoperabilidad.
Esta especificación describe únicamente una parte de un sistema completo. La parte descrita aquí hace referencia únicamente al software integrador de información médica utilizando Health Level Seven (HL7) y Fast Health Interoperability Resources (FHIR).

1.2 Convenciones del Documento

Este documento fue creado en base de la plantilla para Especificación de Requerimientos de Software del IEEE estándar 830.

1.3 Audiencia destino y sugerencias de lectura

Este documento está destinado a desarrolladores, usuarios y testers.

1.4 Alcance del producto

El principal objetivo de este software es proporcionar un middleware que permita obtener y servir los diferentes datos (sin importar su forma interna) de los pacientes, de los diferentes proveedores de servicios médicos, utilizando el estándar FHIR y HL7 habilitando así la interoperabilidad entre estos sistemas.

El principal beneficio obtenido es la interoperabilidad entre los diferentes sistemas de los proveedores de servicios médicos. Además de mejorar la seguridad pública permitiendo que la información precisa de los pacientes puedan ser accesibles desde cualquier lugar.

Las meta a alcanzar es:

- Proveer de un software que integre información médica de diferentes fuentes, sin importar la estructura interna, utilizando Health Level Seven y Fast Health Interoperability Resources.

1.5 Definiciones, acrónimos y abreviaturas

En la siguiente tabla se muestra información sobre algunos términos y acrónimos que se utilizaran en el este documento.

Nombre	Descripción
SIIMH	Software integrador de información médica utilizando Health Level Seven y FHIR.
UF	Usuario final, persona o institución que usará el sistema.
Stakeholder	Personas u organizaciones que están interesadas en la realización de un sistema o proyecto.
ERS	Especificación de Requisitos Software
RQ	Requerimiento Funcional
PSM	Proveedor de Servicios Médicos

2. Descripción general

2.1 Perspectiva del producto

El software aquí especificado es un producto nuevo, y es parte de un software más amplio. Este software hará función de extractor y proveedor de información médica de los pacientes almacenados en las diferentes bases de datos, tanto externas como consorciadas. Para la extracción se hará uso de una capa de seguridad que será la encargada de validar la autenticación y autorización de los pacientes haciendo uso de la tecnología blockchain, parte que no será detallada ni elaborada en este documento. La forma en la que se manejará la comunicación se puede apreciar en la figura 1 presentada en la introducción de este documento de titulación, donde principalmente se manejará una interfaz RESTful con el metaformato XML para enviar datos.

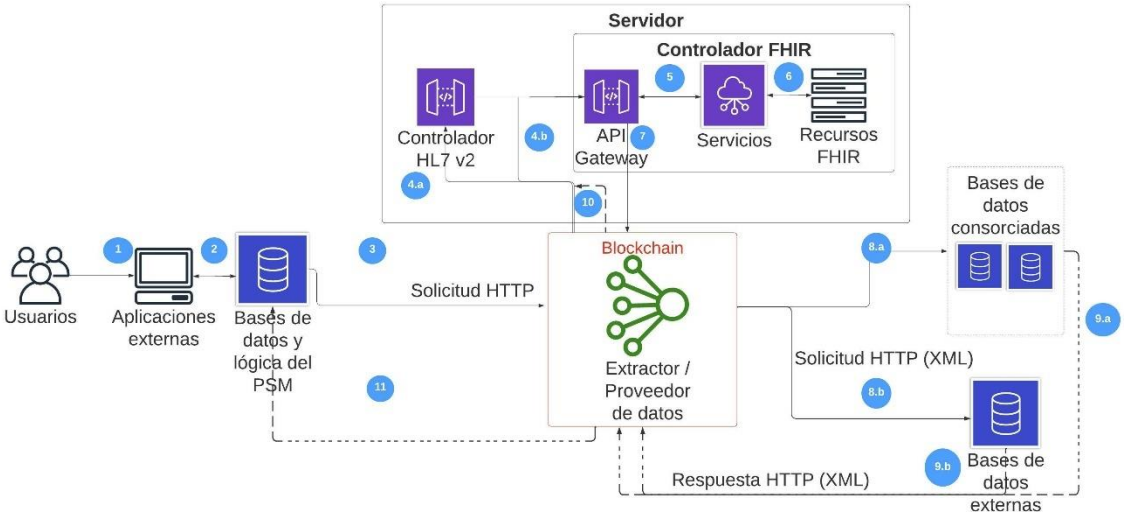


Figure 1 Arquitectura del Sistema

2.2 Funciones del producto

Las principales funcionalidades que el software proveerá, se describen de manera general. Además, se describen las actividades que realizan los usuarios en el sistema.

- Los usuarios poseerán credenciales que les permitan hacer uso del software.
- Los usuarios podrán solicitar datos de pacientes que se encuentren dentro de las bases de datos consorciadas y de otros clientes.
- El software permitirá enviar los datos de los clientes automáticamente después de una solicitud.
- El software recibirá la aprobación de los pacientes a través de la capa de BlockChain para acceder a sus datos.
- El software transformará los datos provenientes y los comunicará usando el estándar FHIR.
- El software recibirá los datos de los diferentes sistemas a través de la definición de un XML.
- El software permitirá recibir datos con el estándar HL7 para su posterior transformación a FHIR.
- El software ofrecerá el servicio REST a través del cual se harán todas las comunicaciones.

2.3 Características y clases de usuarios

Los usuarios que usarán el software se describen a continuación:

Tipo de usuario	Sistemas de los proveedores de servicios médicos
Formación	N/A
Actividades	Realizará las peticiones de los pacientes a través de la capa de BlockChain

Tipo de usuario	Desarrolladores de los proveedores de servicios médicos
Formación	Ingeniero de software o relacionado
Actividades	Describirá la relación entre los atributos de las bases de datos y el formulario obtenido de la conversión FHIR

Tipo de usuario	Administrador del middleware
Formación	Ingeniero de software o relacionado
Actividades	Se encargará de realizar la gestión de los usuarios que acceden al middleware y de generar los archivos necesarios para añadir un nuevo PSM.

Tipo de usuario	Cliente BlockChain
Formación	N/A
Actividades	Cliente que permitirá obtener los permisos necesarios para la lectura de los datos

2.4 Ambiente Operativo

El entorno operativo del software se lista a continuación:

- Docker: versión 20.10.7
- Procesador de 64-bit con Second Level Address Translation (SLAT).
- 4GB de RAM.
- Soporte de virtualización de hardware debe estar habilitada.
- Versión 3.10 o mayor del kernel de Linux.
- Versión de iptables 1.4 o mayor.
- Version git 1.7 o mayor.
- Un ejecutable ps, generalmente proporcionado por procps o un paquete similar.
- XZ Utils 4.9 o mayor

2.5 Restricciones de Diseño e Implementación

A continuación, se listan las restricciones de diseño e implementación del middleware a desarrollar:

- El uso de las APIs de los sistemas externos deberá seguir todos los estándares dispuestos en sus respectivas documentaciones.
- El uso de los datos médicos y su respectiva comunicación se regirán a las leyes ecuatorianas actuales.
- Los sistemas externos que quieran acceder al software deberán enviar sus peticiones en XML o JSON a través del protocolo HTTP.
- Los sistemas externos deberán proveer una API por la cual recibirán los datos en XML.
- El middleware es actualmente solo de lectura, es decir, solo se permiten peticiones del tipo GET.

2.6 Suposiciones y dependencias

El funcionamiento y disponibilidad del software recaera en el acuerdo SLA de los servidores usados y en donde el software se encuentre alojado.
Además se asume la completa disponibilidad del módulo de BlockChain para el desarrollo correcto y a tiempo de este software.

3. Requerimientos de interfaz externos

3.1 Interfaces de software

En la figura 2 se presenta el diagrama de casos de uso para la interacción con BlockChain.

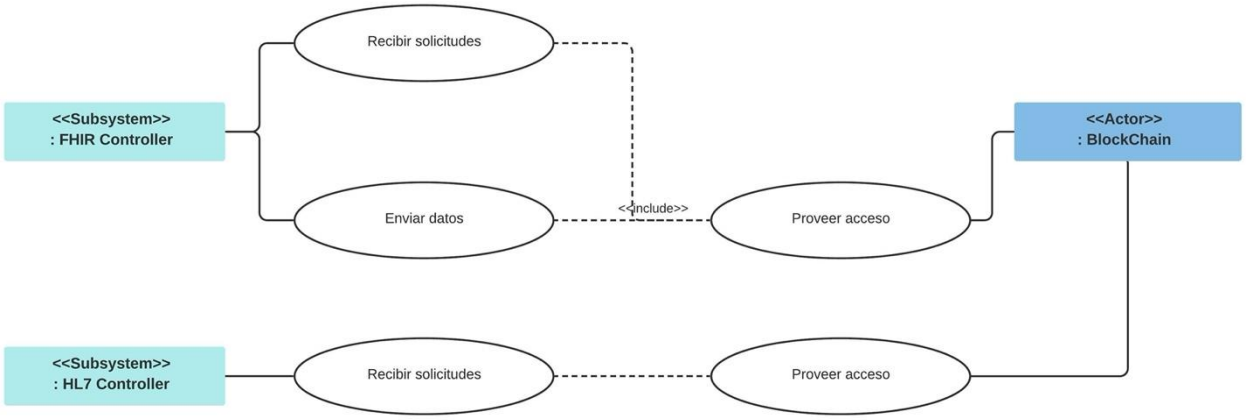


Figure 2 Diagrama de Casos de Uso para interacción del sistema externo BlockChain

La información que intercambian estas dos partes del sistema es:

- Solicitudes de información médica a través de XML.
- Información médica provista por el extractor de datos.
- Autorización a los datos.

3.2 Interfaces de Comunicación

Los estándares de comunicación a usar son HTTP, el acceso de a los datos se manejará a través de la parte del sistema correspondiente a BlockChain, que no está detallada dentro de este documento.

4. Requisitos específicos

4.1 Requisitos funcionales

A continuación, se lista cada uno de los requerimientos específicos funcionales obtenidos para el middleware.

Identificación del requisito:	**RQF01**
Nombre del Requisito:	Extracción de datos

Características:	El middleware deberá extraer los datos de los diferentes sistemas de información médica.
Descripción del requisito:	El middleware recibirá una consulta GET donde se especifique el recurso a obtener, el middleware extraerá los datos necesarios para construir ese recurso recorriendo cada una de los sistemas de información médica hasta hallar, los datos correspondientes.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF02**
Nombre del Requisito:	Transformación de datos
Características:	El middleware deberá transformar los datos obtenidos del sistema de información médica en recursos FHIR.
Descripción del requisito:	El middleware luego de hallar y extraer los datos del recurso en específico debe realizar la transformación a un recurso FHIR para servirlo a través de la respuesta HTTP.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF03**
Nombre del Requisito:	Transformación del formulario: Admisión.

Características:	El middleware deberá ser capaz de construir el formulario médico de: Admisión.
Descripción del requisito:	El middleware debe ser capaz de reconstruir a través de los datos extraídos y los recursos FHIR el formulario médico de: Admisión.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF04**
Nombre del Requisito:	Transformación del formulario Anamnesis.
Características:	El middleware deberá ser capaz de construir el formulario médico de: Anamnesis.
Descripción del requisito:	El middleware debe ser capaz de reconstruir a través de los datos extraídos y los recursos FHIR el formulario médico de: Anamnesis.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF05**
Nombre del Requisito:	Transformación del formulario: Consulta Externa – Anamnesis y EF.

Características:	El middleware deberá ser capaz de construir el formulario médico de: Consulta Externa – Anamnesis y EF.
Descripción del requisito:	El middleware debe ser capaz de reconstruir a través de los datos extraídos y los recursos FHIR el formulario médico de: Consulta Externa – Anamnesis y EF.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF06**
Nombre del Requisito:	Transformación del formulario: Consulta Externa – Evolución.
Características:	El middleware deberá ser capaz de construir el formulario médico de: Consulta Externa – Evolución.
Descripción del requisito:	El middleware debe ser capaz de reconstruir a través de los datos extraídos y los recursos FHIR el formulario médico de: Consulta Externa – Evolución.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF07**
Nombre del Requisito:	Transformación del formulario: Evolución y prescripciones.
Características:	El middleware deberá ser capaz de construir el formulario médico de: Evolución y prescripciones.
Descripción del requisito:	El middleware debe ser capaz de reconstruir a través de los datos extraídos y los recursos FHIR el formulario médico de: Evolución y prescripciones.

Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF08**
Nombre del Requisito:	Transformación del formulario: Examen Físico.
Características:	El middleware deberá ser capaz de construir el formulario médico de: Examen Físico.
Descripción del requisito:	El middleware debe ser capaz de reconstruir a través de los datos extraídos y los recursos FHIR el formulario médico de: Examen Físico.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF09**
Nombre del Requisito:	Acceso a las bases de datos
Características:	El middleware deberá contar con el acceso requerido a los datos de los diferentes proveedores de servicios médicos.
Descripción del requisito:	El middleware requerirá para su funcionamiento adecuado, los respectivos permisos para acceder a los datos de los diferentes proveedores de servicios médicos.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF10**
Nombre del Requisito:	Interfaz RESTful
Características:	El middleware deberá comunicarse con los sistemas externos a través de una interfaz RESTful.
Descripción del requisito:	El middleware deberá ser construido alrededor de una arquitectura REST, de esta forma deberá recibir los datos a través de solicitudes GET y devolverá los datos a través del metaformato XML.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF11**
Nombre del Requisito:	Gestión de usuarios
Características:	El middleware deberá permitir la gestión de los usuarios para su acceso al middleware.
Descripción del requisito:	El middleware permitirá la creación, lectura, eliminación y actualización de múltiples usuarios.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF12**
Nombre del Requisito:	Gestión de acceso a los datos

Características:	El middleware deberá permitir asignar el nivel de acceso a los datos
Descripción del requisito:	El middleware permitirá dar acceso a cada uno de los usuarios creados, todo esto a través de una interfaz web simple, esta interfaz estará disponible únicamente para el administrador del middleware.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF13**
Nombre del Requisito:	Manejo de tiempo de sesión
Características:	El middleware deberá manejar el tiempo de sesión de los usuarios.
Descripción del requisito:	El middleware tendrá un tiempo de sesión de 12 horas para cada usuario, por lo que transcurrido este tiempo el usuario deberá nuevamente acceder al sistema.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF14**
Nombre del Requisito:	Formato FHIR

Características:	El middleware deberá responder siempre con recursos FHIR.
Descripción del requisito:	El middleware responderá a cada solicitud con recursos FHIR
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF15**
Nombre del Requisito:	Aceptación de HL7 versión 2
Características:	El middleware será capaz de recibir mensajes HL7 y convertirlos a un formato FHIR.
Descripción del requisito:	El middleware tendrá la capacidad de recibir datos de sistemas que manejen el estándar HL7 v2.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF16**
Nombre del Requisito:	Acceso a nivel básico

Características:	El middleware responderá con la información dependiendo del nivel de acceso que tenga el usuario.
Descripción del requisito:	El middleware diferenciará los dos niveles de acceso que se proponen, en este caso el básico. Se podrá acceder únicamente a los datos que se incluyen en el formulario 001 del MSP
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF17**
Nombre del Requisito:	Acceso a nivel avanzado
Características:	El middleware responderá con la información dependiendo del nivel de acceso que tenga el usuario.
Descripción del requisito:	El middleware diferenciará los dos niveles de acceso que se proponen, en este caso el avanzado. Se podrá acceder a todos los formularios propuestos
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF18**
Nombre del Requisito:	Generación automática de archivos

Características:	Se deberá realizar una propuesta para la generación automática de los archivos necesarios de cada PSM.
Descripción del requisito:	El middleware incluirá un metamodelo extra que a través de MDE (Model Driven Engineering) genere los archivos y clases necesarias para incluir un nuevo proveedor de servicios médicos al middleware.
Requisito No Funcional:	
Prioridad del requisito:	Alta

Identificación del requisito:	**RQF19**
Nombre del Requisito:	Acceso de los proveedores de servicios médicos
Características:	El middleware estará disponible para su acceso a través de internet, a través de una URL.
Descripción del requisito:	El acceso al middleware será provisto a través de internet, este acceso estará siempre sujeto a los permisos obtenidos por BlockChain.
Requisito No Funcional:	
Prioridad del requisito:	Alta

4.2 Requisistos no funcionales

Por otro lado, en esta sección se listan todos los requerimientos específicos no funcionales para el middleware:

Identificación del requisito:	**RQFN01**
Nombre del Requisito:	Eficiencia
Características:	Los datos transformados deberán estar actualizados.

Descripción del requisito:	Cuando se realice una solicitud el middleware deberá obtener los datos en tiempo real, es decir la última versión de los mismos.
Prioridad del requisito:	Alta

Identificación del requisito:	**RQFN02**
Nombre del Requisito:	Seguridad lógica y de datos
Características	Los permisos de acceso al sistema serán ocultados del público en general.
Descripción del requisito:	El administrador es la única persona con la posibilidad de proveer o retirar los permisos de acceso a la aplicación.
Prioridad del requisito	Alta

Identificación del requisito:	**RQFN03**
Nombre del Requisito:	Manuales
Características	Manual de usuario Manual de uso del sistema
Descripción del requisito:	El middleware deberá ser entregado conjuntamente con los manuales para el correcto uso, así como toda la información necesaria para interpretar el funcionamiento del mismo
Prioridad del requisito	Alta

Identificación del requisito:	**RQFN04**
Nombre del Requisito:	Errores
Características	El middleware es resiliente a errores

Descripción del requisito:	El middleware devolverá el error en caso de existir a través de la respuesta HTTP y no detendrá su funcionamiento.
Prioridad del requisito	Alta

Identificación del requisito:	**RQFN05**
Nombre del Requisito:	Disponibilidad
Características	El middleware deberá estar disponible las 24 horas del día
Descripción del requisito:	El middleware proveerá de sus servicios, sin importar la hora en la que se intente acceder al mismo.
Prioridad del requisito	Alta

Identificación del requisito:	**RQFN06**
Nombre del Requisito:	Compatibilidad
Características	El middleware deberá ser compatible con cualquier plataforma que soporte Docker
Descripción del requisito:	El middleware podrá ser alojado en cualquier servidor que soporte la tecnología Docker
Prioridad del requisito	Alta

Identificación del requisito:	**RQFN07**
Nombre del Requisito:	Licenciamiento
Características	El middleware se acogerá a los dispuesto por la empresa INNTRATEC S.A.S
Descripción del requisito:	El middleware manejará su licenciamiento según lo disponga INNTRATEC S.A.S
Prioridad del requisito	Alta

Identificación del requisito:	**RQFN08**
Nombre del Requisito:	Extensibilidad
Características	El middleware debe ser escalable
Descripción del requisito:	El middleware deberá contar con la capacidad de crecer con facilidad, para esto debe implementarse sobre una tecnología que pueda hacerlo. Además, es importante la documentación para la posterior inclusión de nuevo personal al desarrollo. Por otro lado, el middleware debe ser capaz de incluir a nuevos proveedores de servicios médicos con la cantidad mínima de esfuerzo requerida.
Prioridad del requisito	Alta

Identificación del requisito:	**RQFN09**
Nombre del Requisito:	Mantenimiento
Características	El middleware debe contar con un manual para añadir nuevos proveedores de servicios médicos
Descripción del requisito:	El middleware contará con un manual que indique el procedimiento necesario para añadir a nuevos proveedores de servicio médicos al mismo.
Prioridad del requisito	Alta

Identificación del requisito:	**RQFN10**
Nombre del Requisito:	Sistemas de datos médicos
Características	El middleware deberá regirse a las leyes ecuatorianas actuales de protección de datos médicos.
Descripción del requisito:	El middleware se apegará estrictamente a las políticas y leyes alrededor de la protección de datos de la Republica del Ecuador.
Prioridad del requisito	Alta

Identificación del requisito:	**RQFN11**
Nombre del Requisito:	Confidencialidad
Características	El middleware presentará datos personales de los pacientes solamente a sus operadores y con consentimiento previo del paciente en cuestión.
Descripción del requisito:	El middleware mantendrá una estricta seguridad alrededor de los datos y estos serán mostrados únicamente después de la autorización obtenida por BlockChain y a un usuario debidamente autenticado
Prioridad del requisito	Alta

4.3 Requisitos de interfaces

A continuación, se presentan los requisitos de las interfaces para el middleware

- **Interfaces de usuario:** En este caso las interfaces de usuario se dividirán en 3:
    - **Usuario - Sistemas de los proveedores de servicios médicos:** En este caso la comunicación será a través de HTTPS todo el tiempo y a través de las interfaces RESTful
    - **Usuario - Administrador del middleware:** En este caso este tendrá acceso a una interfaz gráfica que le permitirá gestionar a los usuarios y sus niveles de acceso. Además, este tendrá acceso al metamodelo para poder generar los archivos necesarios para la inclusión de nuevos PSM.
    - **Usuario – Cliente BlockChain:** La interfaz con este tipo de usuario también será HTTPS.

4.4 Requisitos funcionales descripción

En esta sección se describe cada uno de los requisitos funcionales mostrados anteriormente, esto con el fin de facilitar el diseño del sistema para el personal involucrado.

- **RQF01:** Extracción de datos

  El middleware deberá contar con una clase que contenga todos los atributos con los nombres específicos de cada proveedor de servicios médicos, además deberá contar con los endpoints de donde extraer los mismos. Con esta información dentro del middleware, este será capaz de extraer la información al momento de una llamada. La extracción se realizará de manera iterativa sobre cada proveedor de servicios médicos y el proceso se detendrá el momento que se encuentre el primer resultado, también el middleware permitirá recibir como un parámetro el PSM específico para su consulta.

- **RQF02:** Transformación de datos

  El middleware tomará cada uno de los atributos extraídos de la fuente de datos y construirá un recurso FHIR de manera que este sea legible para cualquier sistema de información médica. Una vez transformados los datos, el recurso está listo para ser devuelto como respuesta HTTP en formato XML.

- **RQF03:** Transformación del formulario de Admisión

  El middleware representará varios recursos FHIR y entidades, desde los cuáles se podrá reconstruir el formulario de Admisión, los mismos se podrán consultar de acuerdo a la especificación de las URLs de FHIR

- **RQF04:** Transformación del formulario de Anamnesis

El middleware representará varios recursos FHIR y entidades, desde los cuáles se podrá reconstruir el formulario de Anamnesis, los mismos se podrán consultar de acuerdo a la especificación de las URLs de FHIR, cabe recalcar que no se puede consultar específicamente por el formulario en sí, dado que, esto rompería con el estándar FHIR, esto se cumple para los requerimientos RQF04, RQF05, RQF06, RQF07 y RQF08.

- **RQF05:** Transformación del formulario de Consulta Externa – Anamnesis y EF

  El middleware representará varios recursos FHIR y entidades, desde los cuáles se podrá reconstruir el formulario de Consulta Externa – Anamnesis y EF, los mismos se podrán consultar de acuerdo a la especificación de las URLs de FHIR.

- **RQF06:** Transformación del formulario de Consulta Externa – Evolución

  El middleware representará varios recursos FHIR y entidades, desde los cuáles se podrá reconstruir el formulario de Consulta Externa – Evolución, los mismos se podrán consultar de acuerdo a la especificación de las URLs de FHIR.

- **RQF07:** Transformación del formulario de Evolución y prescripciones

  El middleware representará varios recursos FHIR y entidades, desde los cuáles se podrá reconstruir el formulario de Evolución y prescripciones, los mismos se podrán consultar de acuerdo a la especificación de las URLs de FHIR.

- **RQF08:** Transformación del formulario de Examen Físico

  El middleware representará varios recursos FHIR y entidades, desde los cuáles se podrá reconstruir el formulario de Examen Físico, los mismos se podrán consultar de acuerdo a la especificación de las URLs de FHIR.

- **RQF09:** Acceso a las bases de datos

  El middleware accederá al contenido de las bases de datos a través de endpoints REST, por lo que es importante contar con la respectiva autorización. Una vez autenticado, procederá a hacer las llamadas para acceder a los datos definidos en sus clases.

- **RQF10:** Interfaz RESTful

  El middleware será accedido principalmente a través de una interfaz RESTful a la cual se harán las respectivas solicitudes. El formato en el que responderá el middleware dependerá del content-type que se establezca, entre las opciones está JSON y XML, sin embargo, para el intercambio de información entre proveedores de servicios médicos se recomienda XML dado que es más legible.

- **RQF11:** Gestión de usuarios

  El middleware presentará una interfaz de usuario a la cual podrá acceder únicamente el administrador del sistema, con sus respectivas credenciales,

una vez autenticado, el mismo podrá gestionar a los usuarios. La interfaz le permitirá crear, eliminar, leer y actualizar usuarios.

- **RQF12:** Gestión de acceso a los datos

  El middleware presentará una interfaz de usuario a la cual podrá acceder únicamente el administrador del sistema, con sus respectivas credenciales, una vez autenticado, se podrá editar los accesos al middleware de cada usuario.

- **RQF13:** Manejo de tiempo de sesión

  Para aumentar la seguridad y restringir el acceso a los datos médicos de los pacientes el middleware manejará tiempos de sesión para cada uno de los usuarios conectados al mismo, este tiempo será de 12 horas, por lo que al transcurrir 12 horas de la autenticación de cualquier usuario será necesario volver a ingresar las credenciales para seguir accediendo a cualquier funcionalidad del middleware.

- **RQF14:** Formato FHIR

  El estándar principal a usar en el middleware es el estándar FHIR, por lo que cada consulta de los datos del paciente será resuelta a través de una respuesta con un recurso FHIR, todas las consultas de datos de pacientes y de transformación estarán bajo la dirección: dominio/fhir/. Donde dominio, representa la URL del dominio en donde se encuentre el middleware.

- **RQF15:** Aceptación de HL7 versión 2

  El middleware será capaz de recibir mensajes con el estándar HL7 v2, sin embargo, la respuesta será únicamente en FHIR, la transformación de los mensajes de HL7 a FHIR se hará a través de una la herramienta hl7v2-fhir-converter (LinuxForHealth, 2020).

- **RQF16:** Acceso a nivel básico

  El middleware diferenciará los dos niveles de acceso que se proponen, en este caso el básico. Se podrá acceder únicamente a los datos que se incluyen en el formulario 001 del MSP

- **RQF17:** Acceso a nivel avanzado

  El middleware diferenciará los dos niveles de acceso que se proponen, en este caso el avanzado. Se podrá acceder a todos los formularios propuestos

- **RQF18:** Generación automática de archivos

  El middleware contará con una herramienta externa para la generación automática de los archivos necesarios para la adición de nuevos proveedores de servicios médicos. Esta herramienta es un MDE generado en el IDE Eclipse Modelling Tools.

- **RQF19:** Acceso de los proveedores de servicios médicos

El middleware será montado a un servidor con conexión a internet con un dominio específico, al cual se podrán hacer las respectivas consultas de cada uno de los

recursos y en el cuál estarán cada una de las interfaces detalladas en este documento. Este acceso estará siempre sujeto a los permisos obtenidos por BlockChain.

Anexo 2 – Determinación de recursos FHIR:

Determinación de recursos FHIR

A continuación, se listan los recursos FHIR requeridos para representar la información requerida en cada uno de los formularios del Ministerio de Salud Pública del Ecuador y los atributos si es que es el caso.

Formulario 001

Información requerida	Recurso FHIR	Atributo
ENCABEZADO		
Institución del sistema	Organization	identifier
Unidad operativa	Organization	--
COD. UO	Extension	
Parroquia	Extension	
Cantón	Extension	
Provincia	Extension	
Número de historia clínica	Extension	
REGISTRO DE PRIMERA ADMISIÓN		
Apellido paterno	Patient	Name
Apellido materno	Patient	Name
Primer nombre	Patient	Name
Segundo nombre	Patient	Name
Número de cédula	Patient	Identifier
Dirección de residencia habitual	Patient	Address
Barrio	Extension	
Parroquia	Extension	
Cantón	Extension	
Provincia	Extension	
Zona	Extension	
Número teléfono	Contact	Telecom
Fecha nacimiento	Patient	BirthDate
Lugar de nacimiento	Patient extensions	Patient-birthPlace
Nacionalidad	Patient extensions	Patient-nationality
Grupo cultural	Extension	
Edad Años cumplidos	Se puede calcular con	
Sexo	Patient	Gender
Estado civil	Patient	MaritalStatus
Instrucción último año aprobado	Extension	
Fecha de admisión	Extension	
Ocupación	Extension	
Empresa donde trabaja	Extension	

Tipo de seguro de salud	Extension	
Referido de	Extension	
En caso necesario llamar a	Contact	all
Parentesco - afinidad	Contact	relationship
Dirección	Contact	Address
Número de teléfono	Contact	Telecom
Admisionista		
REGISTRO DE NUEVAS ADMISIONES PARA ATENCIONES DE PRIMERA VEZ Y SUBSECUENTES		
REGISTRO DE CAMBIOS		
Fecha	Extension	
Estado civil	Extension	
Instrucción	Extension	
Ocupación	Extension	
Empresa	Extension	
Tipo de seguro de salud	Extension	
Dirección de residencia habitual	Extension	
Barrio	Extension	
Zona	Extension	
Parroquia	Extension	
Cantón	Extension	
Provincia	Extension	
Número de teléfono	Extension	
INFORMACIÓN ADICIONAL		

Formulario 002

5.1. Anamnesis – Consulta externa

Información requerida	**Recurso FHIR**	**Atributo**
ENCABEZADO		
Establecimiento	Organization	all
Nombre	Patient	Name
Apellido	Patient	Name
Sexo	Patient	Gender
Número de hoja	Extension	
Historia clínica	Extension	
MOTIVO DE CONSULTA		
Motivo de consulta	ClinicalImpression	
ANTECEDENTES PERSONALES		
Vacunas	Inmunization	all
Enfermedad perinatal	Condition	all
Enfermedad infancia	Condition	all

Enfermedad adolescencia	Condition	all
Enfermedad alérgica	AllergyIntolerance	all
Enfermedad cardiaca	Condition	all
Enfermedad respiratoria	Condition	all
Enfermedad digestiva	Condition	all
Enfermedad neurológica	Condition	all
Enfermedad metabólica	Condition	all
Enfermedad hemo linfática	Condition	all
Enfermedad urinaria	Condition	all
Enfermedad traumática	Condition	all
Enfermedad quirúrgica	Condition	all
Enfermedad mental	Condition	all
Enfermedad de transmisión sexual	Condition	all
Tendencia sexual	Extension	
Riesgo social	Extension	
Riesgo laboral	Extension	
Riesgo familiar	Extension	
Actividad física	Extension	
Dieta y hábitos	Extension	
Religión y cultura	Extension	
Otro	Extension	
Menarquia edad	Condition	
Menopausia edad	Condition	
Ciclos	Extension	
Vida sexual activa	Extension	
Gesta	Condition	
Partos	Condition	
Abortos	Condition	
Cesáreas	Condition	
Hijos vivos	Extension	
FUM (fecha última menstruación)	Extension	
FUC (fecha última citología)	Extension	
FUP (fecha ultimo parto)	Extension	
Biopsia	Extension	
Método de P. familiar	Extension	
Terapia hormonal	Extension	
Colposcopía	Extension	
Mamografía	Extension	
ANTECEDENTES FAMILIARES		
Cardiopatía	FamilyMemberHistory	Condition
Diabetes	FamilyMemberHistory	Condition
Enfermedad cerebro vascular	FamilyMemberHistory	Condition
Hipertensión	FamilyMemberHistory	Condition
Cáncer	FamilyMemberHistory	Condition
Tuberculosis	FamilyMemberHistory	Condition
Enfermedad mental	FamilyMemberHistory	Condition
Enfermedad infecciosa	FamilyMemberHistory	Condition

Malformación	FamilyMemberHistory	Condition
Otro	FamilyMemberHistory	Condition
ENFERMEDAD O PROBLEMA ACTUAL		
Cronología	Workflow	----
Localización	Condition	
Características	Condition	
Intensidad	Condition	
Causa aparente	Condition	
Factores que agravan o mejoran	Condition	
Síntomas asociados	Condition	
Evolución	Extension	
Medicamentos que recibe	MedicationStatement	
Resultados de exámenes anteriores	Condition	
Condición actual	Condition	
REVISIÓN ACTUAL DE ÓRGANOS Y SISTEMAS		
Órganos de los sentidos	Observation	all
Respiratorio	Observation	all
Cardio vascular	Observation	all
Digestivo	Observation	all
Genital	Observation	all
Urinario	Observation	all
Musculo esquelético	Observation	all
Endocrino	Observation	all
Hemo Linfático	Observation	all
Nervioso	Observation	all
SIGNOS VITALES		
Fecha	Extension	
Presion arterial	Observation	
Pulso por minuto	Observation	Vitalsigns-Heart rate
Temperatura (grados centígrados)	Observation	Vitalsigns-Body temperatura
EXAMEN FÍSICO		
DIAGNOSTICOS		
Descripción	DiagnosticReport	All
CIE	Extension	
Presuntivo o definitivo	Extension	
PLANES		
PIE DE PAGINA		
Fecha de control	Extension	
Hora fin	Extension	
Médico	Practitioner	Identifier
Código	Extension	
Firma	Extension	

5.2. Consulta externa - Evolución

Información requerida	Recurso FHIR	Atributo
Fecha	Extension	
Hora	Extension	
Evolución	ClinicalImpression	all
Prescripciones	MedicationRequest	All
Medicamentos	Medication	Identifier or code

Formulario 003

5.3. Anamnesis

Información requerida	Recurso FHIR	Atributo
ENCABEZADO		
Establecimiento	Organization	all
Nombre	Patient	Name
Apellido	Patient	Name
Sexo	Patient	Gender
Número de hoja	Extension	
Historia clínica	Extension	
MOTIVO DE CONSULTA		
Motivo de consulta	AdverseEvent (actual)	
ANTECEDENTES PERSONALES		
Vacunas	Inmunization	all
Enfermedad perinatal	Condition	all
Enfermedad infancia	Condition	all
Enfermedad adolescencia	Condition	all
Enfermedad alérgica	AllergyIntolerance	all
Enfermedad cardiaca	Condition	all
Enfermedad respiratoria	Condition	all
Enfermedad digestiva	Condition	all
Enfermedad neurológica	Condition	all
Enfermedad metabólica	Condition	all
Enfermedad hemo linfática	Condition	all
Enfermedad urinaria	Condition	all
Enfermedad traumática	Condition	all
Enfermedad quirúrgica	Condition	all
Enfermedad mental	Condition	all
Enfermedad de transmisión sexual	Condition	all
Tendencia sexual	Extension	
Riesgo social	Extension	
Riesgo laboral	Extension	
Riesgo familiar	Extension	
Actividad física	Extension	

Dieta y hábitos	Extension	
Religión y cultura	Extension	
Otro	Extension	
Menarquia edad	Condition	
Menopausia edad	Condition	
Ciclos	Extension	
Vida sexual activa	Extension	
Gesta	Condition	
Partos	Condition	
Abortos	Condition	
Cesáreas	Condition	
Hijos vivos		
FUM (fecha última menstruación)	Extension	
FUC (fecha última citología)	Extension	
FUP (fecha ultimo parto)	Extension	
Biopsia	Extension	
Método de P. familiar	Extension	
Terapia hormonal	Extension	
Colposcopía	Extension	
Mamografía	Extension	
ANTECEDENTES FAMILIARES		
Cardiopatía	FamilyMemberHistory	Condition
Diabetes	FamilyMemberHistory	Condition
Enfermedad cerebro vascular	FamilyMemberHistory	Condition
Hipertensión	FamilyMemberHistory	Condition
Cáncer	FamilyMemberHistory	Condition
Tuberculosis	FamilyMemberHistory	Condition
Enfermedad mental	FamilyMemberHistory	Condition
Enfermedad infecciosa	FamilyMemberHistory	Condition
Malformación	FamilyMemberHistory	Condition
Otro	FamilyMemberHistory	Condition
ENFERMEDAD O PROBLEMA ACTUAL		
Cronología	Workflow	----
Localización	Condition	
Características	Condition	
Intensidad	Condition	
Causa aparente	Condition	
Factores que agravan o mejoran	Condition	
Síntomas asociados	Condition	
Evolución	Extension	
Medicamentos que recibe	MedicationStatement	
Resultados de exámenes anteriores	Condition	
Condición actual	Condition	
REVISIÓN ACTUAL DE ÓRGANOS Y SISTEMAS		
Órganos de los sentidos	Observation	all
Respiratorio	Observation	all

Cardio vascular	Observation	all
Digestivo	Observation	all
Genital	Observation	all
Urinario	Observation	all
Musculo esquelético	Observation	all
Endocrino	Observation	all
Hemo Linfático	Observation	all
Nervioso	Observation	all

5.4. Examen Físico

Información requerida	**Recurso FHIR**	**Atributo**
SIGNOS VITALES Y MEDICIONES		
Índice de masa corporal	Observation	Vitalsigns-mass index
Presión arterial	Observation	
Frecuencia cardiaca por min	Observation	Vitalsigns-Heart rate
Frecuencia respiratoria por min	Observation	Vitalsigns-Respiratory rate
Temperatura Bucal (grados centígrados)	Observation	Vitalsigns-Body temperature
Temperatura axilar (grados centígrados)	Observation	Vitalsigns-Body temperature
Peso (KG)	Observation	Vitalsigns-Vital signs panel
Talla (m)	Observation	Vitalsigns-Vital signs panel
Perímetro cefálico (cm)	Observation	Vitalsigns-Vital signs panel
EXAMEN FÍSICO (CP/SP)		
Piel y faneras	Observation	
Cabeza	Observation	
Ojos	Observation	
Oidos	Observation	
Nariz	Observation	
Boca	Observation	
Oro Faringe	Observation	
Cuello	Observation	
Axilas - Mamas	Observation	
Torax	Observation	
Abdomen	Observation	
Columna vertebral	Observation	
Ingle - Periné	Observation	
Miembros superiores	Observation	
Miembros inferiores	Observation	
Órganos de los sentidos	Observation	

Respiratorio	Observation	
Cardio Vascular	Observation	
Digestivo	Observation	
Genital	Observation	
Urinario	Observation	
Musculo esquelético	Observation	
Endocrino	Observation	
Hemo linfático	Observation	
Neurológico	Observation	
DIAGNOSTICOS		
Descripción	Extension	
CIE	Extension	
Presuntivo o definitivo	Extension	
PLANES		
PIE DE PAGINA		
Fecha de control	Extension	
Hora fin	Extension	
Médico	**Practitioner**	Identifier
Código		
Firma		

Formulario 005

5.5. Evolución y prescripciones

Información requerida	Recurso FHIR	Atributo
ENCABEZADO		
Establecimiento	Organization	all
Nombre	Patient	Name
Apellido	Patient	Name
Sexo	Patient	Gender
Número de hoja	Extension	
Historia clínica	Extension	
EVOLUCIÓN		
Fecha	Extension	
Hora	Extension	
Evolución	ClinicalImpression	all
Prescripciones	MedicationRequest	All
Medicamentos	Medication	Identifier or code

Recursos adicionales necesarios para el sistema

Recurso FHIR	Atributo

Patient	active
Medication	all
Practitioner	all

Anexo 3 – Ejercicio de capacitación para el middleware (Desarrolladores):

Ejercicio de capacitación para el middleware

¿Qué es FHIR?

El recurso de interoperabilidad Fast Healthcare es un estándar de datos desarrollado y alimentado por HL7 International. FHIR se creó teniendo en cuenta la complejidad de los datos de atención médica y adopta un enfoque moderno basado en Internet para conectar diferentes elementos discretos (Bresnick, 2016).

“La filosofía detrás de FHIR es construir un conjunto base de recursos que, ya sea por sí mismos o combinados, satisfagan la mayoría de los casos de uso comunes. Los recursos de FHIR tienen como objetivo definir el contenido y la estructura de la información para el conjunto de información central que comparten la mayoría de las implementaciones (Bresnick, 2016).

Los elementos de datos, o "recursos", tienen cada uno una etiqueta que actúa como un identificador único, al igual que la URL de una página web. (Bresnick, 2016)

Cuando pida algo en Amazon, por ejemplo, mire la línea de su navegador, explicó Micky Tripathi, director ejecutivo de Massachusetts eHealth Collaborative y presidente del grupo de trabajo de interoperabilidad eHI (Bresnick, 2016).

Si ha iniciado sesión y hace clic en algo, lo que verá es una URL que dice "https" y luego esta enorme serie de sinsentidos. Ese es un sistema de recuperación de consultas que se genera en su navegador y se envía a Amazon, y luego Amazon devuelve inmediatamente los resultados de forma segura (Bresnick, 2016).

En Internet, los usuarios de todo el mundo pueden acceder a la misma URL y completar las mismas tareas utilizando cualquier navegador estándar que se

ejecute en cualquier dispositivo habilitado para la web, ya sea un teléfono inteligente, computadora de escritorio o tableta con Windows, Apple, Android o Linux (Bresnick, 2016).

FHIR espera hacer lo mismo: permitir a los desarrolladores crear aplicaciones de "navegador" estandarizadas que permitan el acceso a los datos sin importar qué "sistema operativo" EHR respalde la infraestructura del usuario (Bresnick, 2016).

Un recurso FHIR puede ser un paquete individual de información que incluye metadatos, texto o elementos de datos particulares, pero también puede agruparse en colecciones que crean documentos clínicos, similar al C-CDA, pero mucho más flexible (Bresnick, 2016).

Los recursos de FHIR se pueden utilizar para crear documentos que representen una composición: un conjunto de información coherente que es una declaración de información de atención médica, en particular que incluye observaciones y servicios clínicos. Un documento es un conjunto inmutable de recursos con una presentación fija que es autorizada y / o certificada por humanos, organizaciones y dispositivos (Bresnick, 2016).

Al crear una URL accesible y estándar para estos paquetes de información en lugar de simplemente pasar documentos individuales de un sistema a otro, varias aplicaciones diferentes pueden apuntar a la misma versión de los mismos datos cada vez (Bresnick, 2016).

La razón del porqué este tipo de estándares son construidos e implementados en cientos de sistemas de información médica es habilitar la interoperabilidad entre estos.

Por ejemplo, un recurso de FHIR que representa a un paciente se vería tal cual se muestra en la siguiente URL: https://www.hl7.org/fhir/patient-example.xml.html

La lista completa de recursos FHIR se puede encontrar en la siguiente URL: Resourcelist - FHIR v4.0.1 (hl7.org)

Middleware FHIR

Para habilitar la interoperabilidad de los sistemas de los proveedores de servicios médicos en la república del Ecuador se desarrolló un middleware, todo esto mediante el uso de las normas internacionales HL7 y FHIR, y teniendo en cuenta ciertas lecciones aprendidas de trabajos previos relacionados con el área.

El funcionamiento del middleware es a través de consultas HTTP principalmente GET donde se solicita la información en específica relacionada a los datos médicos de un paciente en específico. Estas URL siguen los lineamientos sugeridos por FHIR para la construcción de las URLs y el formato en la que las consultas son llevadas a cabo. Por ejemplo, si se requiere obtener la información concerniente a las vacunas del paciente con id 1 la URL, debería ser la siguiente:

http://localhost:8081/middleware/fhir/Immunization?patient:Patient=1

El middleware siempre responde en formato XML o JSON dependiendo del encabezado que se le envié, cabe aclarar que también requiere de un token obtenido con la autenticación del respectivo usuario.

El funcionamiento principal del middleware radica en la extracción de datos de diferentes bases de datos previamente definidas dentro del middleware, para convertirlas al estándar FHIR y servirlas a los distintos sistemas de información médica que las soliciten. De esta forma y con esta información los proveedores de servicios médicos (PSM) podrían transformarlas a una estructura que su sistema de información médica sea capaz de manejar.

Objetivos

El objetivo de este documento de capacitación es realizar un ejercicio donde se muestre el funcionamiento correcto del middleware FHIR para el caso en el que se requiere obtener el recurso del paciente.

Descripción del caso

Existe un PSM llamada *EM* que se encuentra actualmente dentro del middleware y tiene una estructura interna del paciente propia, es decir, no tiene una relación directa con el estándar FHIR ni con ningún otro sistema de información médica. Su entidad paciente se muestra en la siguiente imagen:

PersonEM	
tipoIdentificacion	String
numeroIdentificacion	String
primerNombre	String
segundoNombre	String
apellidoPaterno	String
apellidoMaterno	String
genero	String
estadoCivil	String
telefono	String
celular	String
correoElectronico	String

Como se puede observar, para el estándar FHIR sería imposible poder determinar qué es lo que significa cada uno de los campos que se listan. Además, dentro de la base de datos se encuentra el siguiente paciente ficticio:

```
{
  "id": 2,
  "tipoIdentificacio": "RUC",
  "numeroIdentificacion": "0106604416001",
  "primerNombre": "Juan",
  "segundoNombre": "Pablo",
  "apellidoPaterno": "Perez",
  "apellidoMaterno": "Moreira",
  "genero": "Masculino",
  "estadoCivil": "Soltero",
  "telefono": "4058511",
  "celular": "0994479846",
  "correoElectronico": "micorreo@ucuenca.edu.ec"
}
```

Por lo que, al realizar la transformación de los datos se debería obtener el mismo objeto, pero en el estándar FHIR

Pasos

Para la elaboración del experimento se construyó una máquina virtual que emulará el funcionamiento del sistema en el servidor. El middleware está divido en dos partes:

- Front end: https://telemedicina.ucuenca.edu.ec/fhir
- Back end: https://telemedicina.ucuenca.edu.ec/middleware

Pero dada la complejidad del ejercicio se prefiere trabajar de manera local para simular un el ambiente en el que el middleware será desplegado. La máquina virtual se puede encontrar en:

http://link.com

El software de virtualización a usar será Oracle VM VirtualBox cuyo instalador se puede encontrar en la siguiente URL: Downloads – Oracle VM VirtualBox.

Una instalado el software se puede realizar la importación de la máquina virtual, seleccionando la sección de herramientas y luego haciendo clic en importar, una vez en el cuadro de configuración se selecciona el archivo .ova descargado previamente y se dejan los valores por defecto de la máquina. Cabe recalcar, que para un buen funcionamiento se requiere de un computador con una capacidad mayor a 8GB de RAM.

Una vez dentro de la máquina virtual, para este caso abriremos la terminal de Ubuntu y ejecutaremos los siguientes comandos:

cd Desktop

sh front.sh

En una pestaña nueva y sin cerrar la anterior ejecutamos los siguientes comandos:

sh back.sh

*Importante: el directorio debe ser siempre Desktop

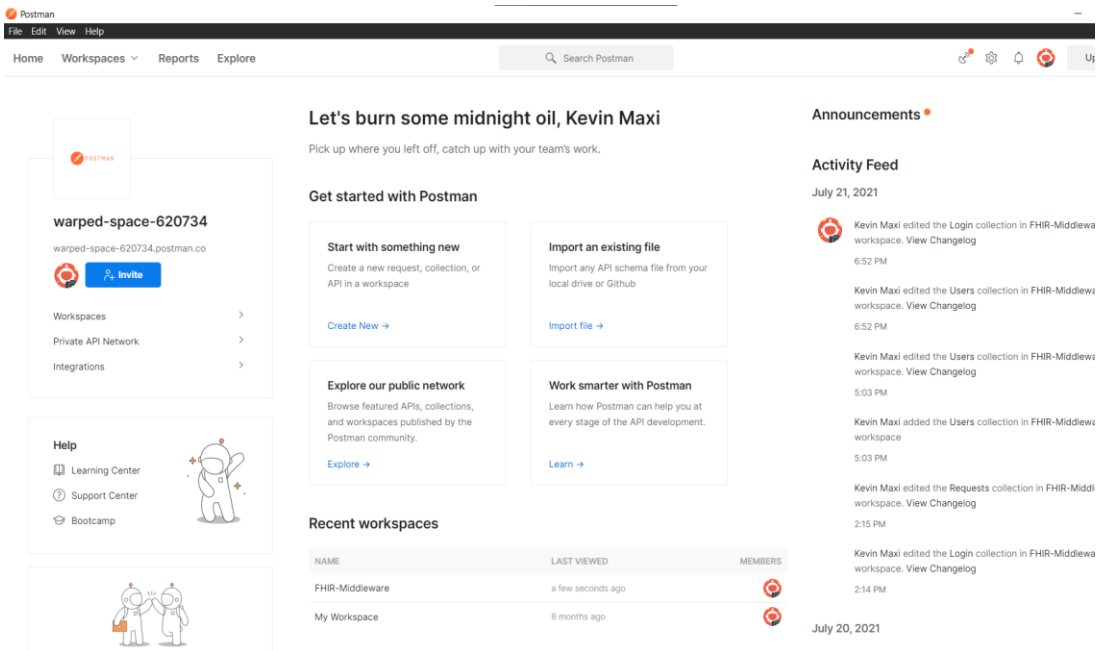
Una vez realizada esta tarea es importante mantener abierta la ventana o ventanas del terminal para el resto del experimento.

El middleware cuenta con una capa de seguridad que maneja la autenticación de los usuarios previo a ingresar al sistema, dada la sensibilidad de los datos que maneja, además cuenta con un tiempo de sesión de 12 horas. Para este ejemplo se puede acceder con las siguientes credenciales:

- Usuario: PSM3

- Contraseña: psm1234

Dado que no se cuenta con el acceso a un software médico simularemos el funcionamiento a través de un cliente HTTP, que en este ejemplo será Postman®



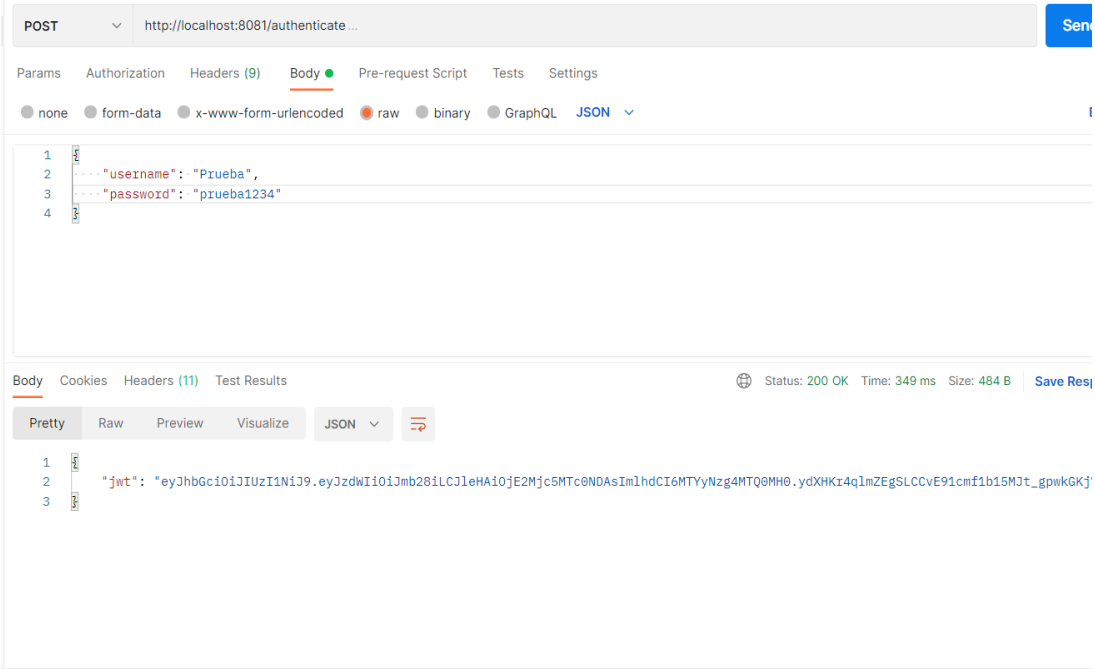
Para autenticarnos haremos una solicitud POST a la URL:

http://localhost:8081/middleware/authenticate

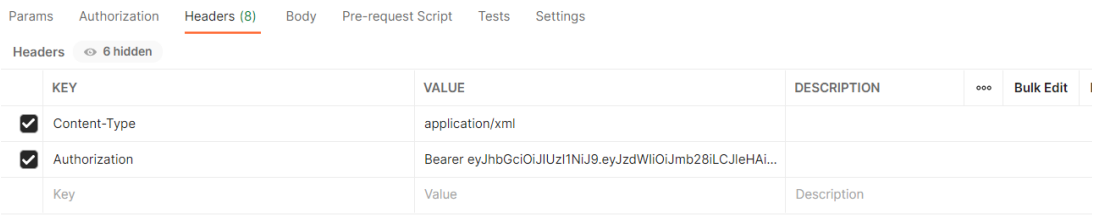
En él la sección de body enviaremos las credenciales de prueba:

```
{
    "username": "PSM3",
    "password": "psm1234"
}
```

Al ejecutar el POST el middleware nos devolverá nuestro token, en el body, en campo de jwt. El ejemplo en postman se puede observar en la siguiente imagen.



Una vez obtenido el token disponemos de 12 horas para realizar las transacciones con el mismo. Para los siguientes pasos es importante incluir el token obtenido en header como se muestra en la siguiente imagen.

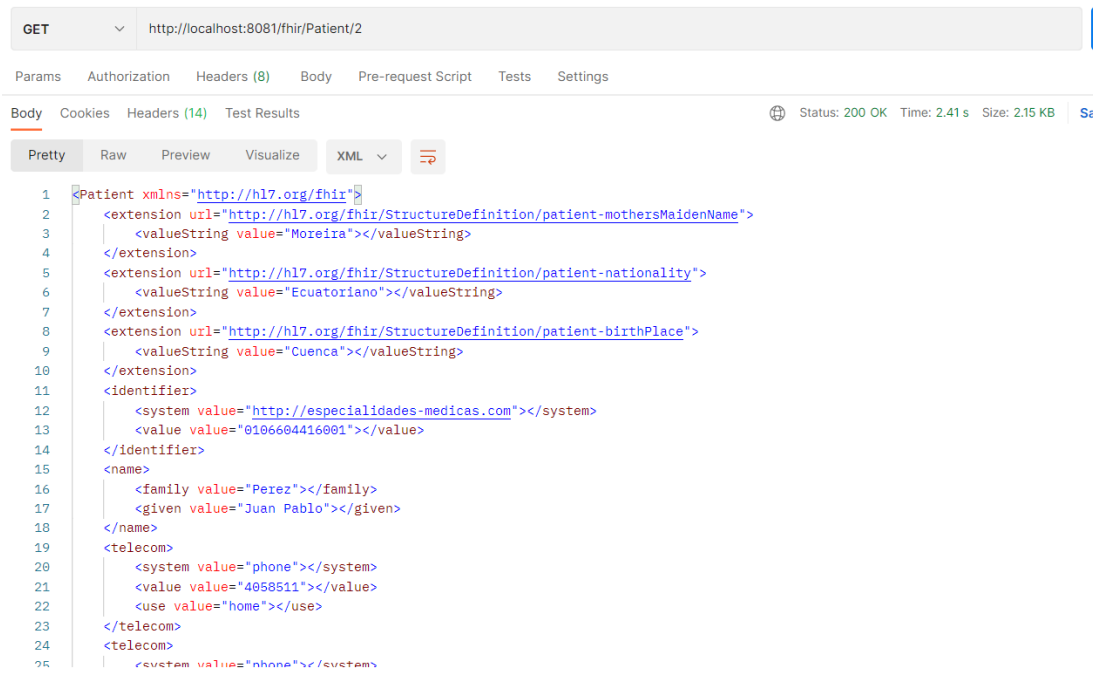


Es importante especificar el tipo de contenido que en este caso será XML.

Una vez configurados los Header, procedemos a ejecutar la operación de consulta de los datos del paciente con id 2, para esto vamos a hacer una solicitud GET a la url:

http://localhost:8081/middleware/fhir/Patient/2 `

Cabe recalcar que esta URL sigue lo definido por FHIR para la construcción de las mismas, al ejecutar la operación el middleware nos responde con el XML del paciente en el estándar FHIR



El resultado obtenido debería ser idéntico a lo descrito a continuación:

```
<Patient xmlns="http://hl7.org/fhir">
    <extension url="http://hl7.org/fhir/StructureDefinition/patient-
mothersMaidenName">
        <valueString value="Moreira"></valueString>
    </extension>
    <extension url="http://hl7.org/fhir/StructureDefinition/patient-
nationality">
        <valueString value="Ecuatoriano"></valueString>
    </extension>
    <extension url="http://hl7.org/fhir/StructureDefinition/patient-
birthPlace">
        <valueString value="Cuenca"></valueString>
    </extension>
    <identifier>
        <system value="http://especialidades-medicas.com"></system>
        <value value="0106604416001"></value>
    </identifier>
    <name>
        <family value="Perez"></family>
        <given value="Juan Pablo"></given>
    </name>
    <telecom>
        <system value="phone"></system>
        <value value="4058511"></value>
        <use value="home"></use>
    </telecom>
    <telecom>
        <system value="phone"></system>
        <value value="0994479846"></value>
        <use value="mobile"></use>
```

```
    </telecom>
    <telecom>
        <system value="email"></system>
        <value value="micorreo@ucuenca.edu.ec"></value>
        <use value="home"></use>
    </telecom>
    <gender value="male"></gender>
    <address>
        <extension url="http://hl7.org/fhir/StructureDefinition/iso21090-
ADXP-houseNumber">
            <valueString value="S/N"></valueString>
        </extension>
        <line value="Mi calle principal"></line>
        <line value="Sin secundaria"></line>
        <state value="Azuay"></state>
    </address>
    <maritalStatus>
        <coding>
            <system value="http://terminology.hl7.org/CodeSystem/v3-
MaritalStatus"></system>
            <code value="U"></code>
            <display value="unmarried"></display>
        </coding>
    </maritalStatus>
    <contact>
        <name>
            <family value="Jara Nancy Lorena"></family>
        </name>
        <telecom>
            <system value="phone"></system>
            <value value="0994479846"></value>
            <use value="home"></use>
        </telecom>
        <address>
            <line value="Calle del Escudo Nacional"></line>
        </address>
    </contact>
</Patient>
```


Conclusiones

Como se puede observar el proceso es relativamente simple, teniendo en cuenta también que la información del paciente, no contiene demasiada complejidad ni incluye datos médicos.

Anexo 4 – Ejercicio de capacitación para el middleware (Administradores)

Ejercicio de capacitación para el middleware

¿Qué es FHIR?

El recurso de interoperabilidad Fast Healthcare es un estándar de datos desarrollado y alimentado por HL7 International. FHIR se creó teniendo en cuenta la complejidad de los datos de atención médica y adopta un enfoque moderno basado en Internet para conectar diferentes elementos discretos (Bresnick, 2016).

“La filosofía detrás de FHIR es construir un conjunto base de recursos que, ya sea por sí mismos o combinados, satisfagan la mayoría de los casos de uso comunes. Los recursos de FHIR tienen como objetivo definir el contenido y la estructura de la información para el conjunto de información central que comparten la mayoría de las implementaciones (Bresnick, 2016).

Los elementos de datos, o "recursos", tienen cada uno una etiqueta que actúa como un identificador único, al igual que la URL de una página web. (Bresnick, 2016)

Cuando pida algo en Amazon, por ejemplo, mire la línea de su navegador, explicó Micky Tripathi, director ejecutivo de Massachusetts eHealth Collaborative y presidente del grupo de trabajo de interoperabilidad eHI (Bresnick, 2016).

Si ha iniciado sesión y hace clic en algo, lo que verá es una URL que dice "https" y luego esta enorme serie de sinsentidos. Ese es un sistema de recuperación de consultas que se genera en su navegador y se envía a Amazon, y luego Amazon devuelve inmediatamente los resultados de forma segura (Bresnick, 2016).

En Internet, los usuarios de todo el mundo pueden acceder a la misma URL y completar las mismas tareas utilizando cualquier navegador estándar que se ejecute en cualquier dispositivo habilitado para la web, ya sea un teléfono inteligente, computadora de escritorio o tableta con Windows, Apple, Android o Linux (Bresnick, 2016).

FHIR espera hacer lo mismo: permitir a los desarrolladores crear aplicaciones de "navegador" estandarizadas que permitan el acceso a los datos sin importar qué "sistema operativo" EHR respalde la infraestructura del usuario (Bresnick, 2016).

La clave de esto es el recurso. Un recurso FHIR puede ser un paquete individual de información que incluye metadatos, texto o elementos de datos particulares, pero también puede agruparse en colecciones que crean documentos clínicos, similar al C-CDA, pero mucho más flexible (Bresnick, 2016).

Los recursos de FHIR se pueden utilizar para crear documentos que representen una composición: un conjunto de información coherente que es una declaración de información de atención médica, en particular que incluye observaciones y servicios clínicos. Un documento es un conjunto inmutable de recursos con una presentación fija que es autorizada y / o certificada por humanos, organizaciones y dispositivos (Bresnick, 2016).

Al crear una URL accesible y estándar para estos paquetes de información en lugar de simplemente pasar documentos individuales de un sistema a otro, varias aplicaciones diferentes pueden apuntar a la misma versión de los mismos datos cada vez (Bresnick, 2016).

La razón del porqué este tipo de estándares son construidos e implementados en cientos de sistemas de información médica es habilitar la interoperabilidad entre estos.

Por ejemplo, un recurso de FHIR que representa a un paciente se vería tal cual se muestra en la siguiente URL: https://www.hl7.org/fhir/patient-example.xml.html

La lista completa de recursos FHIR se puede encontrar en la siguiente URL: Resourcelist - FHIR v4.0.1 (hl7.org)

Middleware FHIR

Para habilitar la interoperabilidad de los sistemas de los proveedores de servicios médicos en la república del Ecuador se desarrolló un middleware, todo esto mediante el uso de las normas internacionales HL7 y FHIR, y teniendo en cuenta ciertas lecciones aprendidas de trabajos previos relacionados con el área.

Para poder acceder a todas las funcionalidades del middleware es necesario contar con un usuario y sus credenciales previamente creadas, por lo que una de las principales tareas del usuario Administrador del middleware es la gestión de los usuarios y los niveles de acceso. Otra tarea principal del administrador del

middleware es la inclusión de nuevos PSM dentro del middleware, por lo que el mismo será el encargado de recibir la información correspondiente al alineamiento entre los atributos de la base de datos en específico y el modelo generado para la transformación. Con esta información realizará la generación de los archivos necesarios a través del metamodelo construido para luego incluirlos dentro del middleware.

Objetivos

El objetivo de este documento de capacitación es realizar un ejercicio donde se muestre el funcionamiento correcto del middleware FHIR para el caso en el que se requiere realizar la gestión de un usuario nuevo y la generación de los archivos necesarios para incluir a un PSM dentro del middleware.

Descripción del caso

1. **Gestión de usuarios**

Existe un nuevo usuario con nivel de acceso avanzado que requiere ser añadido al middleware para que pueda realizar las respectivas transacciones sus credenciales serían las siguientes:

- Usuario: nuevoPSM
- Email: nuevopsm@gmail.com
- Rol: PSM1
- Status: Activo
- Contraseña: PSMseguro

Por otro lado, se requiere actualizar el nombre del usuario PSM1 a Hospital Nacional y cambiar su nivel de acceso a PSM3.

Finalmente se desea eliminar al usuario PSM2 de la lista de usuarios.

2. **Generación de archivos**

Para la transformación de los datos se estableció una clase general con ciertos atributos necesarios para la transformación a FHIR, por lo que, el funcionamiento correcto del middleware depende de la alineación de los atributos de la entidad de la base de datos del PSM y el listado generado en la sección anterior.

Para la generación de los archivos se cuenta con la siguiente información de la entidad paciente obtenida del PSM que está solicitando su adición al middleware:

Atributo en el PSM	Propiedad
Apll_mat	Apellido Materno
Apll_pat	Apellido Paterno
barr	Barrio
Cll_prncpal	Calle principal
Cll_scndria	Calle secundaria
Cntn_nac	Cantón nacimiento
Cntn_res	Cantón residencia
cel	Celular
email	Correo Electrónico
Drccn_cntct	Dirección contacto
Std_cvl	Estado civil
Fch_nac	Fecha nacimiento
gen	Género
nac	Nacionalidad
Lgr_nac	Lugar de nacimiento
Fch_nac	Fecha de nacimiento
Nmbr_apll_cntct	Nombres apellidos contacto
Num_cas	Numero casa
Num_id	Numero identificación
prentsco	Parentesco
Prrq_nac	Parroquia nacimiento
Prrq_resid	Parroquia residencia
Prim_nmbr	Primer nombre
Prov_nac	Provincia nacimiento
Prov_res	Provincia residencia
Ref_res	Referencia residencia

Seg_nmbr	Segundo nombre
tel	Teléfono
Tel_cont	Teléfono contacto
Tp_id	Tipo identificación
https://medical-data-server.herokuapp.com/patientsHospitalPrueba	Url Todos
https://medical-data-server.herokuapp.com/patientsHospitalPrueba	Url Uno

Nombre de la Organización: HospitalPrueba

Es importante que los atributos no contengan espacios intermedios y tampoco guiones.

Pasos

Para la elaboración del experimento se construyó una máquina virtual que emulará el funcionamiento del sistema en el servidor. El middleware está divido en dos partes:

- Front end: https://telemedicina.ucuenca.edu.ec/fhir
- Back end: https://telemedicina.ucuenca.edu.ec/middleware

Pero dada la complejidad del ejercicio se prefiere trabajar de manera local para simular un el ambiente en el que el middleware será desplegado. La máquina virtual se puede encontrar en:

http://link.com

El software de virtualización a usar será Oracle VM VirtualBox cuyo instalador se puede encontrar en la siguiente URL: Downloads – Oracle VM VirtualBox.

Una instalado el software se puede realizar la importación de la máquina virtual, seleccionando la sección de herramientas y luego haciendo clic en importar, una vez en el cuadro de configuración se selecciona el archivo .ova descargado previamente y se dejan los valores por defecto de la máquina. Cabe recalcar, que para un buen funcionamiento se requiere de un computador con una capacidad mayor a 8GB de RAM.

Una vez dentro de la máquina virtual, para este caso abriremos la terminal de Ubuntu y ejecutaremos los siguientes comandos:

cd Desktop

sh front.sh

En una pestaña nueva y sin cerrar la anterior ejecutamos los siguientes comandos:

sh back.sh

*Importante: el directorio debe ser siempre Desktop

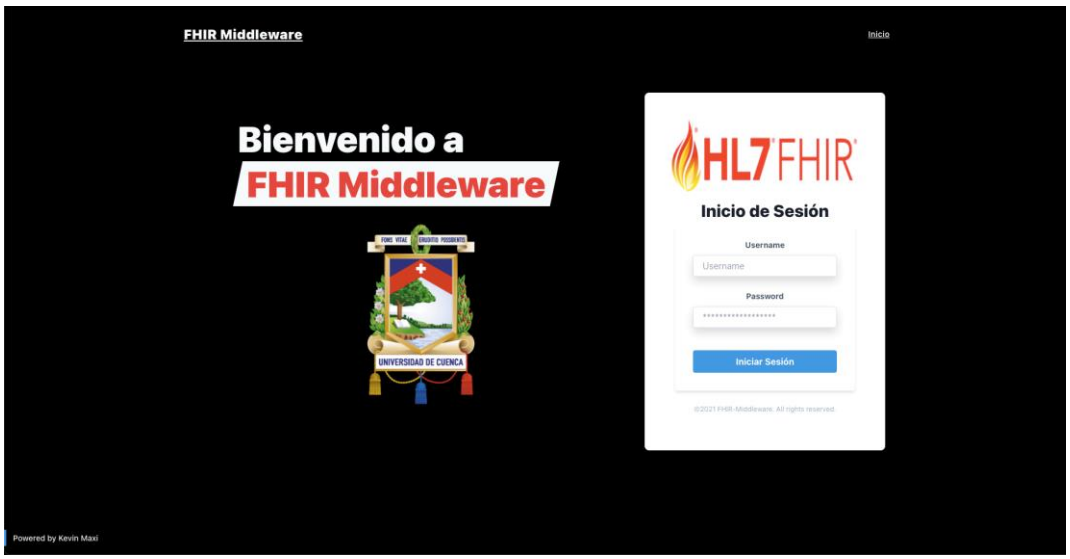
Una vez realizada esta tarea es importante mantener abierta la ventana o ventanas del terminal para el resto del experimento.

El middleware cuenta con una capa de seguridad que maneja la autenticación de los usuarios previo a ingresar al sistema, dada la sensibilidad de los datos que maneja, además cuenta con un tiempo de sesión de 12 horas. Para este ejemplo se puede acceder con las siguientes credenciales:

- Usuario: Administrador
- Contraseña: administrador12

Para esto se puede dirigir a la url:

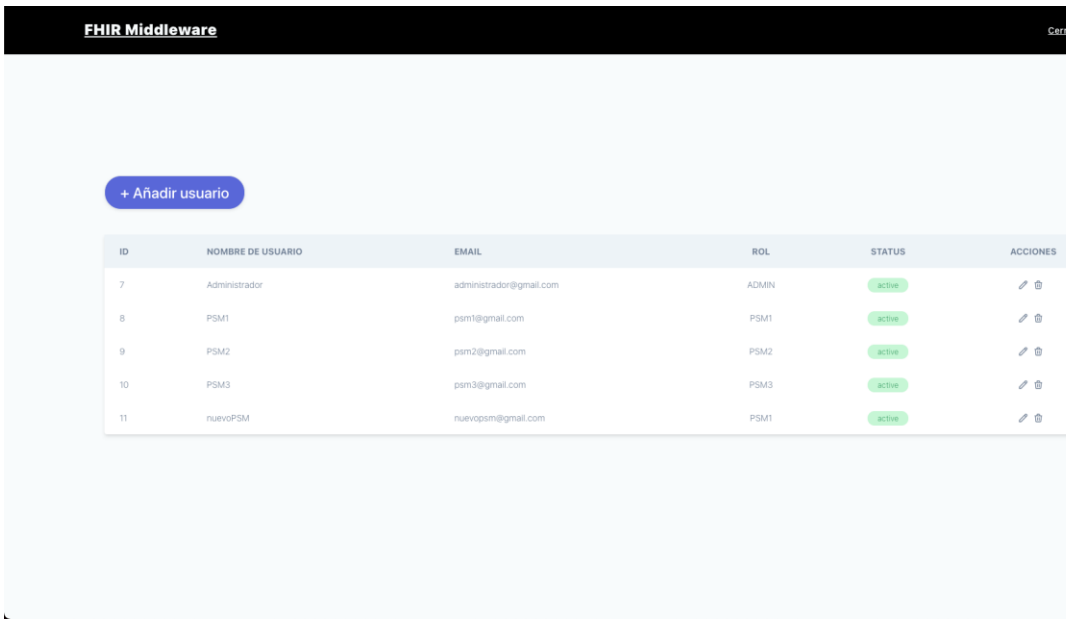
http://localhost:3000/



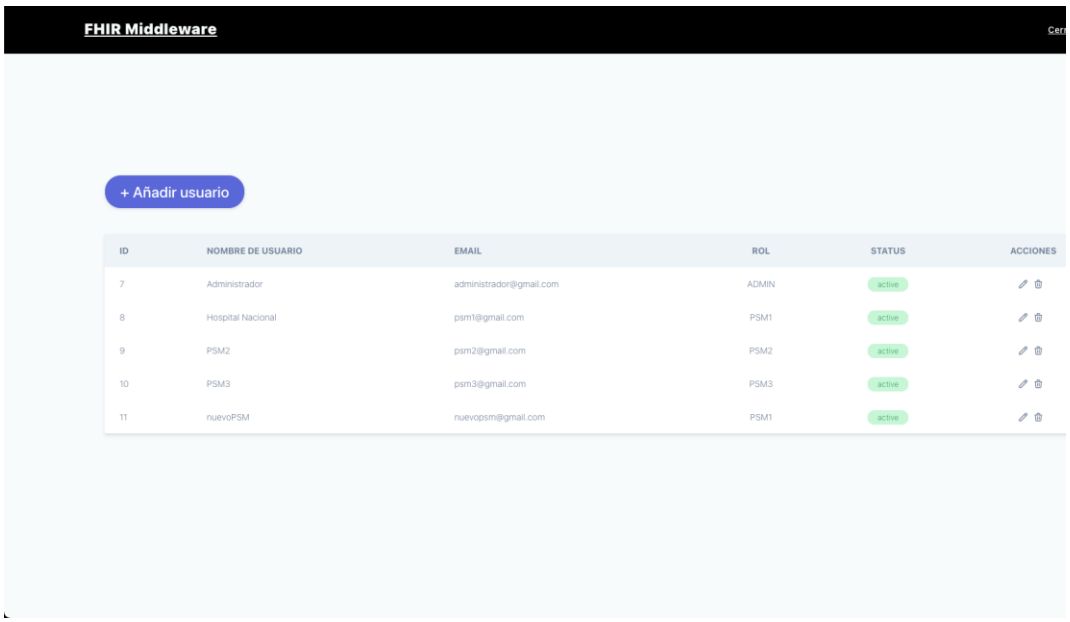
Una vez iniciado sesión la siguiente interfaz será lanzada:

Como se puede observar el middleware cuenta con tres usuarios previamente creados, uno para cada nivel de acceso y con los que se realizarán las pruebas.

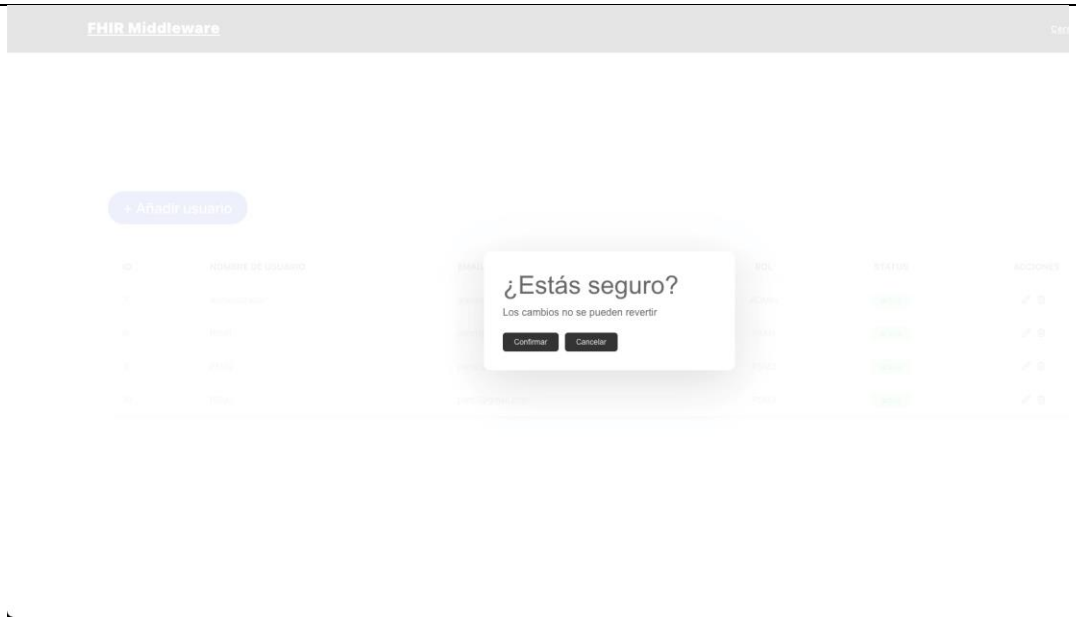
Para crear un nuevo usuario pulsamos el botón añadir e ingresamos los datos del nuevoPSM. Al finalizar la pantalla debería ser semejante a la siguiente imagen.

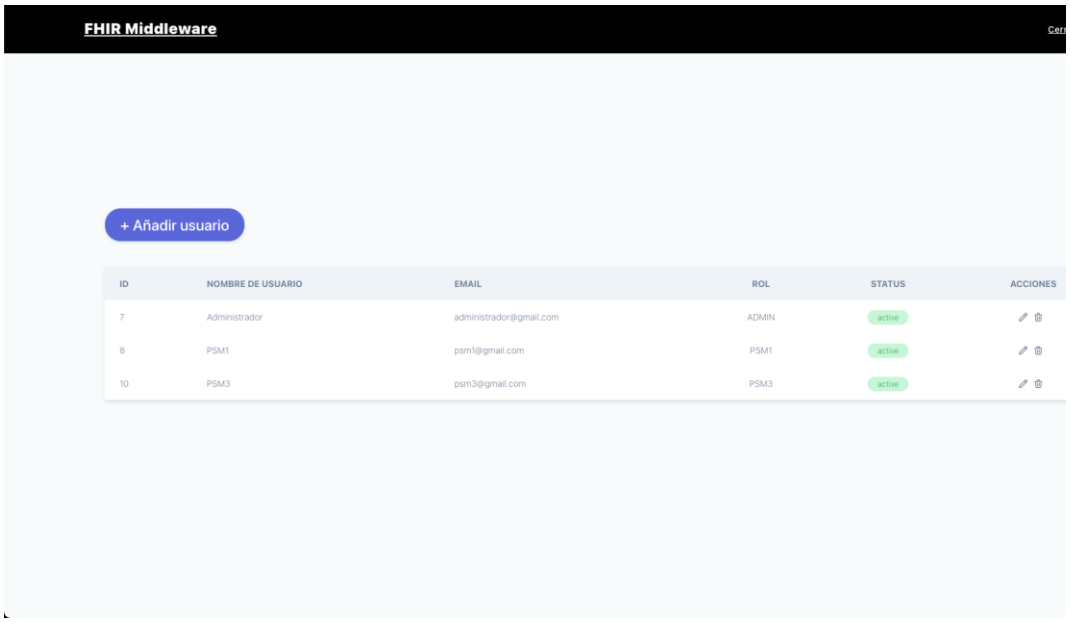


Para realizar la edición de los usuarios basta con hacer clic sobre el botón de edición y se mostrará el mismo formulario de creación pero esta vez con los datos del usuario que está siendo editado, al terminar de realizar los cambios de información para el usuario PSM1, la pantalla debería ser semejante a la siguiente imagen:



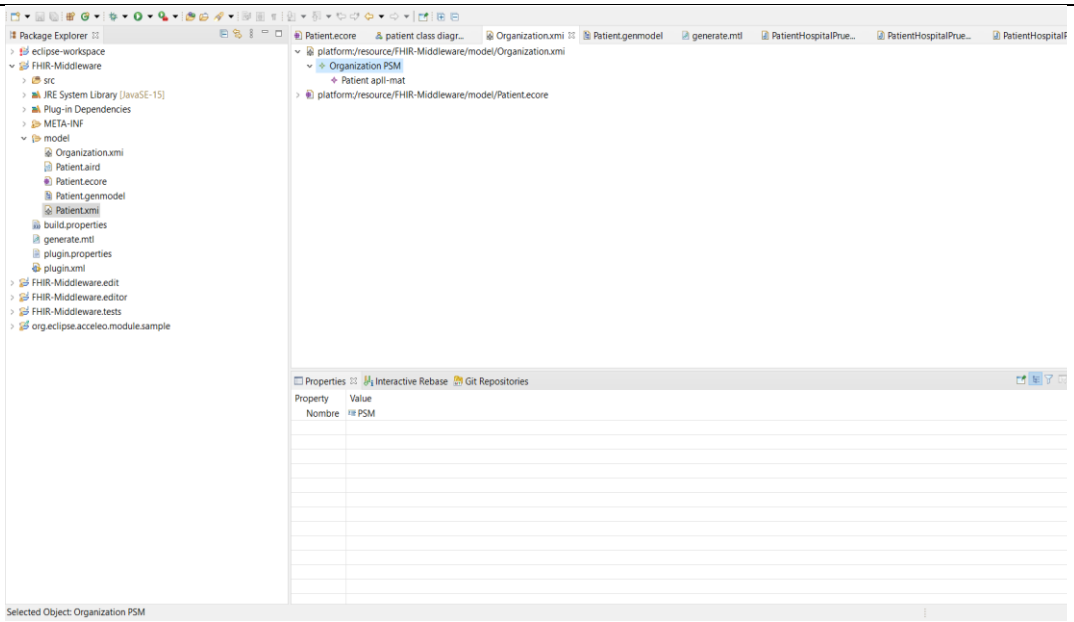
Finalmente para eliminar podemos hacer clic sobre el botón rojo con una x, confirmamos la eliminación del usuario y el resultado de esta operación debería presentar algo semejante a la siguientes imágenes:



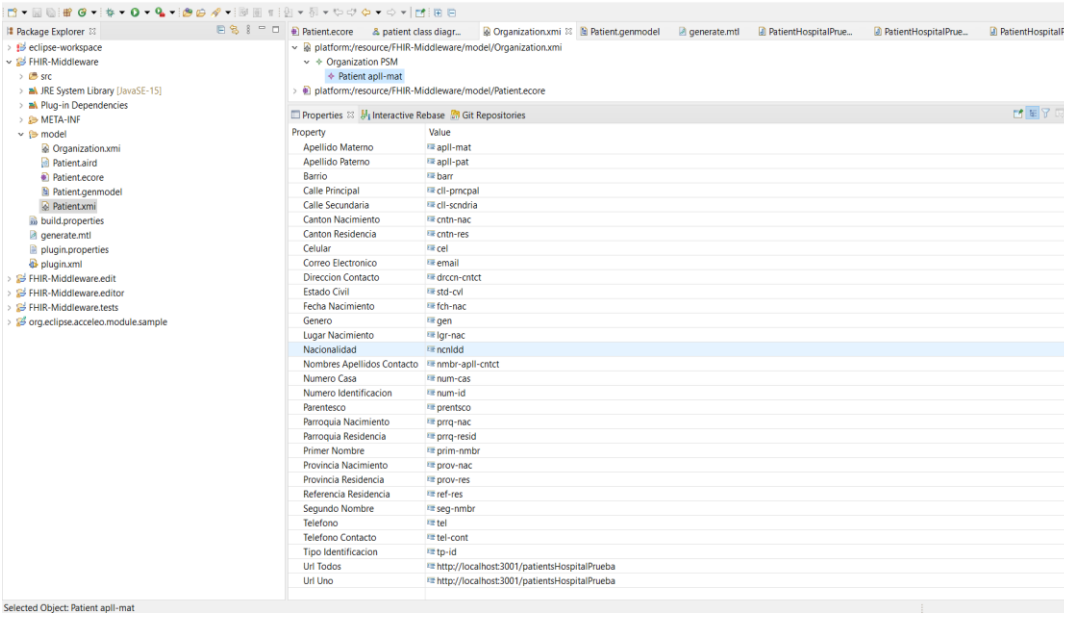


Para la segunda parte, abriremos el mde generado en eclipse modelling tools. Una vez dentro nos dirigiremos a la carpeta de FHIR middleware abriremos el ecore, una vez dentro y verificado que el paciente se encuentre en el ecore, Nos dirigimos a FHIR-middleware -> model -> Organization.xmi, lo abrimos desplegamos la sección de nombre platform:/resource/FHIR-Middleware/model/Organization.xmi

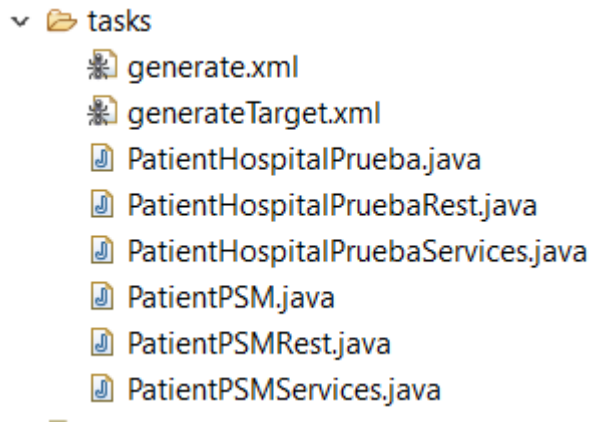
Luego cambiaremos el nombre de la organización PSM, haciendo clic en Organization, en la ventana de properties en la sección de abajo se muestran todas las propiedades, ahí es donde se verá la propiedad nombre y una columna Value vacía. El cambio debería verse así:



El mismo paso se hace para ingresar la tabla del alineamiento, pero haciendo clic sobre Patient, el resultado debería verse así:



Luego ejecutamos el proyecto y los resultados se presentan en la carpeta tasks dentro de org.eclipse.acceleo.module.sample:



Donde los 3 últimos archivos son los generados en este ejemplo. Posterior a eso se mueven estos archivos y se incluyen en el middleware. Para realizar esto se pueden utilizar los dos accesos directos que se encuentran en el escritorio. El acceso directo llamado FHIR-middleware-link redirige a la carpeta del proyecto, mientras que el acceso directo llamado FilesGenerated-link redirige al directorio de los archivos generados. El primer paso es dirigirse a src/main/java/com/fhir/model y crear una carpeta con el nombre del nuevo PSM en este caso, HospitalPrueba, luego se debe crear otra carpeta en con el mismo nombre pero en la ruta src/main/java/com/fhir/services. Luego el archivo modelo (archivo de tipo [nombreRecurso][nombrePSM].java) se debe mover al directorio src/main/java/com/fhir/HospitalPrueba. Luego el archivo extractor REST (archivo de tipo [nombreRecurso][nombrePSM]Rest.java) se debe mover a la ruta src/main/java/com/fhir/rest. Finalmente, el archivo de servicios (archivo de tipo [nombreRecurso][nombrePSM]Services.java) se debe mover a la ruta src/main/java/com/fhir/services/HospitalPrueba.

Para validar que el proceso fue exitoso se puede dirigir a la URL:

http://localhost:8081/middleware/fhir/Patient/2

Dado que este paciente existe únicamente en la base de datos incluida en este ejercicio si el resultado devuelve un XML con el contenido del paciente con id 3, el resultado es exitoso de lo contrario ha fracasado

Conclusiones

El proceso de gestión de usuarios es simple dada la simplicidad con la que estos se manejan. Mientras que el proceso de generación de archivos añade una complejidad de tiempo mayor sin embargo es un proceso relativamente corto comparado a lo que tomaría escribir los archivos de manera manual.

Anexo 5 – Ejercicio de evaluación para el middleware (desarolladores)

Ejercicio de evaluación para el middleware

¿Qué es FHIR?

El recurso de interoperabilidad Fast Healthcare es un estándar de datos desarrollado y alimentado por HL7 International. FHIR se creó teniendo en cuenta la complejidad de los datos de atención médica y adopta un enfoque moderno basado en Internet para conectar diferentes elementos discretos (Bresnick, 2016).

“La filosofía detrás de FHIR es construir un conjunto base de recursos que, ya sea por sí mismos o combinados, satisfagan la mayoría de los casos de uso comunes. Los recursos de FHIR tienen como objetivo definir el contenido y la estructura de la información para el conjunto de información central que comparten la mayoría de las implementaciones (Bresnick, 2016).

Los elementos de datos, o "recursos", tienen cada uno una etiqueta que actúa como un identificador único, al igual que la URL de una página web. (Bresnick, 2016)

Cuando pida algo en Amazon, por ejemplo, mire la línea de su navegador, explicó Micky Tripathi, director ejecutivo de Massachusetts eHealth Collaborative y presidente del grupo de trabajo de interoperabilidad eHI (Bresnick, 2016).

Si ha iniciado sesión y hace clic en algo, lo que verá es una URL que dice "https" y luego esta enorme serie de sinsentidos. Ese es un sistema de recuperación de consultas que se genera en su navegador y se envía a Amazon, y luego Amazon devuelve inmediatamente los resultados de forma segura (Bresnick, 2016).

En Internet, los usuarios de todo el mundo pueden acceder a la misma URL y completar las mismas tareas utilizando cualquier navegador estándar que se ejecute en cualquier dispositivo habilitado para la web, ya sea un teléfono inteligente, computadora de escritorio o tableta con Windows, Apple, Android o Linux (Bresnick, 2016).

FHIR espera hacer lo mismo: permitir a los desarrolladores crear aplicaciones de "navegador" estandarizadas que permitan el acceso a los datos sin importar qué "sistema operativo" EHR respalde la infraestructura del usuario (Bresnick, 2016).

La clave de esto es el recurso. Un recurso FHIR puede ser un paquete individual de información que incluye metadatos, texto o elementos de datos particulares, pero también puede agruparse en colecciones que crean documentos clínicos, similar al C-CDA, pero mucho más flexible (Bresnick, 2016).

Los recursos de FHIR se pueden utilizar para crear documentos que representen una composición: un conjunto de información coherente que es una declaración de información de atención médica, en particular que incluye observaciones y servicios clínicos. Un documento es un conjunto inmutable de recursos con una presentación fija que es autorizada y / o certificada por humanos, organizaciones y dispositivos (Bresnick, 2016).

Al crear una URL accesible y estándar para estos paquetes de información en lugar de simplemente pasar documentos individuales de un sistema a otro, varias aplicaciones diferentes pueden apuntar a la misma versión de los mismos datos cada vez (Bresnick, 2016).

La razón del porqué este tipo de estándares son construidos e implementados en cientos de sistemas de información médica es habilitar la interoperabilidad entre estos.

Por ejemplo un recurso de FHIR que representa a un paciente se vería tal cual se muestra en la siguiente URL: https://www.hl7.org/fhir/patient-example.xml.html

La lista completa de recursos FHIR se puede encontrar en la siguiente URL: Resourcelist - FHIR v4.0.1 (hl7.org)

Middleware FHIR

Para habilitar la interoperabilidad de los sistemas de los proveedores de servicios médicos en la república del Ecuador se desarrolló un middleware, todo esto mediante el uso de las normas internacionales HL7 y FHIR, y teniendo en cuenta ciertas lecciones aprendidas de trabajos previos relacionados con el área.

El funcionamiento del middleware es a través de consultas HTTP principalmente GET donde se solicita la información en específica relacionada a los datos médicos de un paciente en específico. Estas URL siguen los lineamientos sugeridos por FHIR para la construcción de las URLs y el formato en la que las consultas son

llevadas a cabo. Por ejemplo, si se requiere obtener la información concerniente a las vacunas del paciente con id 1 la URL, debería ser la siguiente:

http://localhost:8081/middleware/fhir/Immunization?patient:Patient=1

El middleware siempre responde en formato XML o JSON dependiendo del encabezado que se le envié, cabe aclarar que también requiere de un token obtenido con la autenticación del respectivo usuario.

El funcionamiento principal del middleware radica en la extracción de datos de diferentes bases de datos previamente definidas dentro del middleware, para convertirlas al estándar FHIR y servirlas a los distintos sistemas de información médica que las soliciten. De esta forma y con esta información los proveedores de servicios médicos (PSM) podrían transformarlas a una estructura que su sistema de información médica sea capaz de manejar.

Adicional a esto se presenta la métrica de transformación de datos que permitirá medir el ratio con el que el middleware transforma los datos a recursos FHIR

Tasa de transformación de datos: La métrica de tasa de transformación de datos es una ponderación de la cantidad de datos usados y transformados a recursos FHIR de una entidad en específico, versus la cantidad de datos provistos por la estructura interna de una base de datos en específico. En la ecuación (1) se presenta la fórmula para realizar el cálculo de esta métrica.

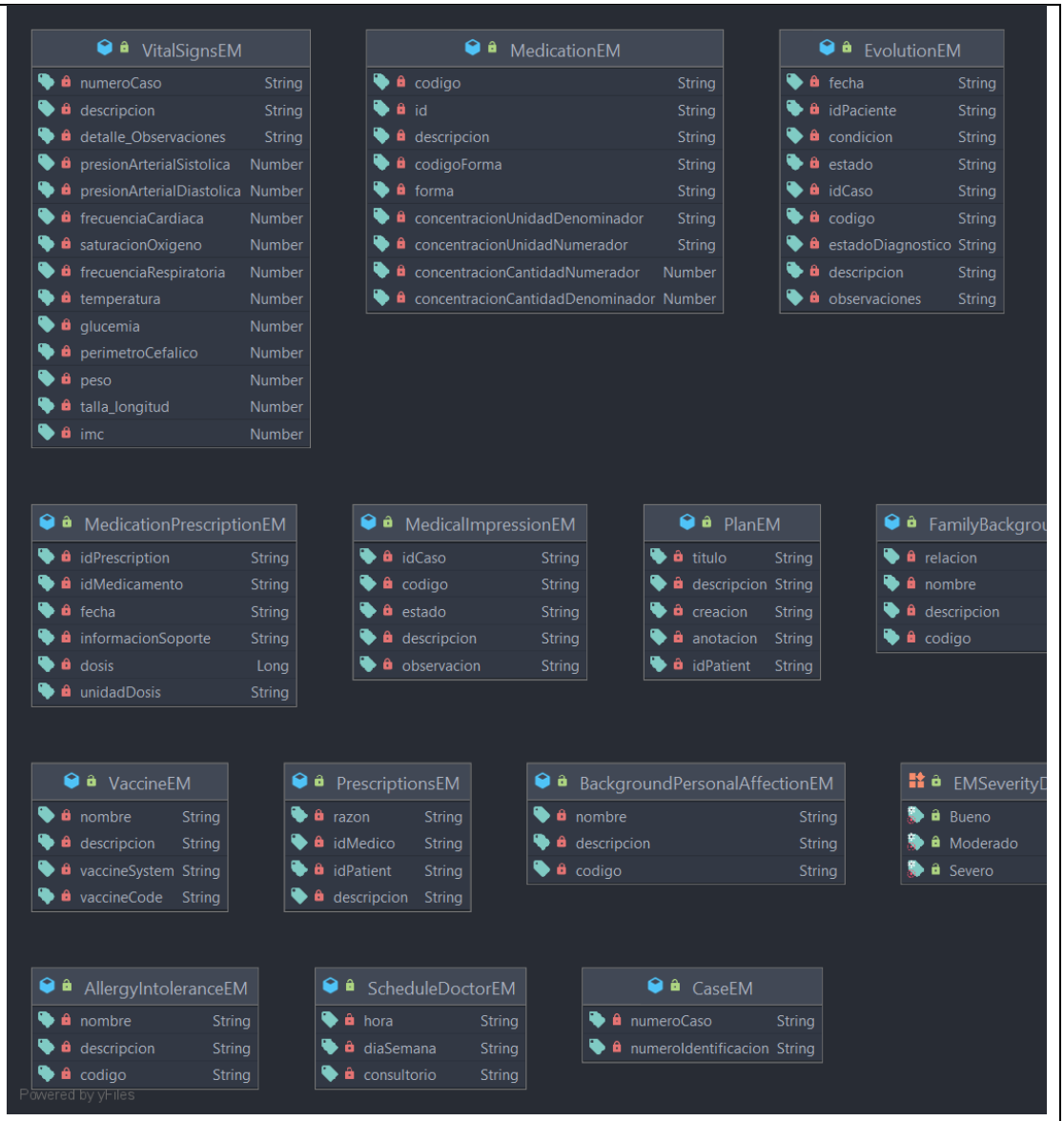
$$TTD = \frac{\text{número de datos incluidos en el recurso FHIR}}{\text{número de datos disponibles en la base de datos}}$$

Objetivos

El objetivo de este documento de capacitación es realizar un ejercicio donde se muestre el funcionamiento correcto del middleware FHIR para el caso en el que se requiere obtener el recurso del paciente.

Descripción del caso

Existe un PSM llamada *EM* que se encuentra actualmente dentro del middleware y tiene una estructura interna del paciente, propia, es decir, no tiene una relación directa con el estándar FHIR ni con ningún otro sistema de información médica. Las entidades que contiene este PSM se describen en la siguiente imagen:



Como se puede observar, para el estándar FHIR sería imposible poder determinar qué es lo que significa cada uno de los campos que se listan.

Pasos

Hora de inicio:	**hh:mm:ss**

1. El usuario Sistemas de los proveedores de servicios médicos, simulará el funcionamiento del middleware haciendo solicitudes a través de un cliente http, en este caso, hará una solicitud a la url, *http://localhost:8081/fhir/Condition?patient:Patient=$id&history=true*,

donde *$id* debe ser reemplazado por 1, la respuesta debería ser el recurso FHIR de tipo Bundle con los datos provistos por la base de datos del ejemplo.

Hora de fin:	**hh:mm:ss**

Hora de inicio:	**hh:mm:ss**

2. El usuario Sistemas de los proveedores de servicios médicos, simulará el funcionamiento del middleware haciendo solicitudes a través de un cliente http, en este caso, hará una solicitud a la url, *http://localhost:8081/fhir/Observation?subject:Patient=$id&identifier=$ident*, donde *$id* debe ser reemplazado por 1 y *$ident* por 1, la respuesta debería ser el recurso FHIR de tipo Bundle con los datos provistos por la base de datos del ejemplo.

Hora de fin:	**hh:mm:ss**

Hora de inicio:	**hh:mm:ss**

3. El usuario Sistemas de los proveedores de servicios médicos, simulará el funcionamiento del middleware haciendo solicitudes a través de un cliente http, en este caso, hará una solicitud a la url, *http://localhost:8081/fhir/MedicationAdministration?patient:Patient=$id&idPrescription=$ident*, donde *$id* debe ser reemplazado por 1 y *$ident* por 1, la respuesta debería ser el recurso FHIR de tipo Patient con los datos provistos por la base de datos del ejemplo.

Hora de fin:	**hh:mm:ss**

Hora de inicio:	**hh:mm:ss**

4. El usuario Sistemas de los proveedores de servicios médicos, simulará el funcionamiento del middleware haciendo solicitudes a través de un cliente

http, en este caso, hará una solicitud a la url, *http://localhost:8081/fhir/Patient/$id*, donde $id debe ser reemplazado por el 3, en este caso el acceso a la base de datos no cuenta con un paciente con ese id, para probar la resiliencia al error, la respuesta debería ser el error de que la no se encuentra el objeto en la base de datos consultada.

Hora de fin:	**hh:mm:ss**

Hora de inicio:	**hh:mm:ss**

5. Calcular el TTD de cada uno de los de los 4 pasos anteriores y anotarlos aquí (posteriormente se pedirá que se los ingrese en la encuesta).

Hora de fin:	**hh:mm:ss**

Los datos de prueba en este caso se describen en el siguiente JSON:


```
"VitalSignsEM": [
  {
    "id": 1,
    "numeroCaso": "1",
    "descripcion": "Signos vitales ejemplo 1",
    "detalle_Observaciones": "Detalle vitales ejemplo 1",
    "presionArterialSistolica": 20,
    "presionArterialDiastolica": 50,
    "frecuenciaCardiaca": 80,
    "saturacionOxigeno": 90,
    "frecuenciaRespiratoria": 20,
    "temperatura": 36,
    "glucemia": 40,
    "perimetroCefalico": 20,
    "peso": 70,
```

"talla_longitud": 1.7,
"imc": 20
}
],
"DoctorsEM": [
{
"id": 1,
"tipoIdentificacio": "RUC",
"numeroIdentificacion": "0106604419001",
"primerNombre": "Kevin Doctor",
"segundoNombre": "Alexander Doctor",
"apellidoPaterno": "Maxi Doctor",
"apellidoMaterno": "Jara",
"genero": "Masculino",
"estadoCivil": "Soltero",
"telefono": "4122240",
"celular": "0984601231",
"correoElectronico": "kevin.maxi@ucuenca.edu.ec",
"especializacion": "Cirujano"
}
],
"VaccinesEM": [
{
"id": 1,
"nombre": "Vacunación profiláctica contra resfriado común",
"descripcion": "Vacuna administrada por mi hospital",
"vaccineSystem": "http://hl7.org/fhir/sid/icd-10",

"vaccineCode": "Z25.8"
},
{
"id": 2,
"nombre": "Vacunación profiláctica contra el cólera con tifoidea-paratifoidea (cólera + TAB)",
"descripcion": "Vacuna administrada por mi hospital",
"vaccineSystem": "http://hl7.org/fhir/sid/icd-10",
"vaccineCode": "Z27.0"
},
{
"id": 3,
"nombre": "Vacunación profiláctica contra combinación de enfermedades NCOP",
"descripcion": "Vacuna administrada por mi hospital",
"vaccineSystem": "http://hl7.org/fhir/sid/icd-10",
"vaccineCode": "Z27.9"
}
],
"BackgroundPersonalEM": [
{
"id": 1,
"nombre": "Diarrea",
"descripcion": "Derivado de pediatría",
"codigo": "A09"
},
{
"id": 2,

"nombre": "Hepatitis tipo A",
"descripcion": "Derivado de emergencias",
"codigo": "B15.9"
},
{
"id": 3,
"nombre": "Dengue",
"descripcion": "Derivado de pediatria",
"codigo": "A91"
},
{
"id": 4,
"nombre": "Fiebre arenavirus",
"descripcion": "Derivado de emergencias",
"codigo": "A96"
}
],
"AllergyIntoleranceEM": [
{
"id": 1,
"nombre": "Dermatitis",
"descripcion": "Alergía niñez",
"codigo": "L23"
},
{
"id": 2,
"nombre": "Enteropatía",

"descripcion": "Alergía niñez",
"codigo": "k52.2"
},
{
"id": 3,
"nombre": "gastroenteritis",
"descripcion": "Alergía niñez",
"codigo": "k528"
}
],
"FamilyBackgroundEM": [
{
"id": 1,
"relacion": "Esposo",
"nombre": "cardiopatía",
"descripcion": "COV",
"codigo": "A91"
}
],
"DiagnosticEM": [
{
"id": 1,
"idCaso": 256,
"codigo": "A91",
"estado": "presuntivo",
"descripcion": "cardiopatía",
"observaciones": "Esta es una observación"

```
    }
  ],
  "EvolutionEM": [
    {
      "id": "1",
      "fecha": "2021-06-06",
      "idPaciente": "1",
      "condicion": "Vivo",
      "estado": "Bueno",
      "idCaso": 256,
      "codigo": "A91",
      "estadoDiagnostico": "presuntivo",
      "descripcion": "cardiopatía",
      "observaciones": "Esta es una observación"
    }
  ],
  "MedicationEM": [
    {
      "id": "1",
      "codigo": "261000",
      "descripcion": "Codeine phosphate",
      "codigoForma": "385022006",
      "forma": "Líquido oral",
      "concentracionUnidadDenominador": "mL",
      "concentracionUnidadNumerador": "mg",
      "concentracionCantidadNumerador": 10,
      "concentracionCantidadDenominador": 5
```

```
    }
  ],
  "PrescriptionsEM": [
    {
      "id": "1",
      "razon": "emergencia",
      "idMedico": "0106604416001",
      "idPatient": "0106604408",
      "descripcion": "Recetado después de una emergencia"
    }
  ],
  "MedicationPrescriptionsEM": [
    {
      "id": "1",
      "idPrescription": "1",
      "idMedicamento": "1",
      "fecha": "2021-07-05",
      "informacionSoporte": "Esta es la información de soporte",
      "dosis": 25,
      "unidadDosis": "mg/mL"
    }
  ],
  "PlansEM": [
    {
      "id": 1,
      "titulo": "Cuidado de la fiebre",
      "descripcion": "Plan para los cuidados de la fiebre",
```

"creacion": "27/06/2021",
"anotacion": "Esta es una nota adicional",
"idPatient": "0106604408"
},
{
"id": 2,
"titulo": "Cuidado de la gripe",
"descripcion": "Plan para los cuidados de la gripe",
"creacion": "27/06/2022",
"anotacion": "Esta es una nota adicional 2",
"idPatient": "0106604408"
}
]

El resultado obtenido debería ser idéntico a los archivos almacenados en la siguiente url:

https://drive.google.com/drive/folders/1aEb919SiSmMD80oNwM8KRaBdxub3Yzn_?usp=sharing

Una herramienta para comparar los XML sin necesidad de leerlos por completo se puede encontrar en:

XML diff - Compare xml online (extendsclass.com)

Al finalizar los ejercicios completa la siguiente encuesta:

- https://forms.gle/GWpmxYShi8FhFxn67

Anexo 6 – Ejercicio de evaluación para el middleware (Administradores)

Ejercicio de evaluación para el middleware

¿Qué es FHIR?

El recurso de interoperabilidad Fast Healthcare es un estándar de datos desarrollado y alimentado por HL7 International. FHIR se creó teniendo en cuenta la complejidad de los datos de atención médica y adopta un enfoque moderno basado en Internet para conectar diferentes elementos discretos (Bresnick, 2016).

“La filosofía detrás de FHIR es construir un conjunto base de recursos que, ya sea por sí mismos o combinados, satisfagan la mayoría de los casos de uso comunes. Los recursos de FHIR tienen como objetivo definir el contenido y la estructura de la información para el conjunto de información central que comparten la mayoría de las implementaciones (Bresnick, 2016).

Los elementos de datos, o "recursos", tienen cada uno una etiqueta que actúa como un identificador único, al igual que la URL de una página web. (Bresnick, 2016)

Cuando pida algo en Amazon, por ejemplo, mire la línea de su navegador, explicó Micky Tripathi, director ejecutivo de Massachusetts eHealth Collaborative y presidente del grupo de trabajo de interoperabilidad eHI (Bresnick, 2016).

Si ha iniciado sesión y hace clic en algo, lo que verá es una URL que dice "https" y luego esta enorme serie de sinsentidos. Ese es un sistema de recuperación de consultas que se genera en su navegador y se envía a Amazon, y luego Amazon devuelve inmediatamente los resultados de forma segura (Bresnick, 2016).

En Internet, los usuarios de todo el mundo pueden acceder a la misma URL y completar las mismas tareas utilizando cualquier navegador estándar que se ejecute en cualquier dispositivo habilitado para la web, ya sea un teléfono inteligente, computadora de escritorio o tableta con Windows, Apple, Android o Linux (Bresnick, 2016).

FHIR espera hacer lo mismo: permitir a los desarrolladores crear aplicaciones de "navegador" estandarizadas que permitan el acceso a los datos sin importar qué "sistema operativo" EHR respalde la infraestructura del usuario (Bresnick, 2016).

La clave de esto es el recurso. Un recurso FHIR puede ser un paquete individual de información que incluye metadatos, texto o elementos de datos particulares, pero también puede agruparse en colecciones que crean documentos clínicos, similar al C-CDA, pero mucho más flexible (Bresnick, 2016).

Los recursos de FHIR se pueden utilizar para crear documentos que representen una composición: un conjunto de información coherente que es una declaración de información de atención médica, en particular que incluye observaciones y servicios clínicos. Un documento es un conjunto inmutable de recursos con una presentación fija que es autorizada y / o certificada por humanos, organizaciones y dispositivos (Bresnick, 2016).

Al crear una URL accesible y estándar para estos paquetes de información en lugar de simplemente pasar documentos individuales de un sistema a otro, varias aplicaciones diferentes pueden apuntar a la misma versión de los mismos datos cada vez (Bresnick, 2016).

La razón del porqué este tipo de estándares son construidos e implementados en cientos de sistemas de información médica es habilitar la interoperabilidad entre estos.

Por ejemplo un recurso de FHIR que representa a un paciente se vería tal cual se muestra en la siguiente URL: https://www.hl7.org/fhir/patient-example.xml.html

La lista completa de recursos FHIR se puede encontrar en la siguiente URL: Resourcelist - FHIR v4.0.1 (hl7.org)

Middleware FHIR

Para habilitar la interoperabilidad de los sistemas de los proveedores de servicios médicos en la república del Ecuador se desarrolló un middleware, todo esto mediante el uso de las normas internacionales HL7 y FHIR, y teniendo en cuenta ciertas lecciones aprendidas de trabajos previos relacionados con el área.

Para poder acceder a todas las funcionalidades del middleware es necesario contar con un usuario y sus credenciales previamente creadas, por lo que una de las principales tareas del usuario Administrador del middleware es la gestión de los usuarios y los niveles de acceso. Otra tarea principal del administrador del middleware es la inclusión de nuevos PSM dentro del middleware, por lo que el mismo será el encargado de recibir la información correspondiente al alineamiento entre los atributos de la base de datos en específico y el modelo generado para la transformación. Con esta información realizará la generación de los archivos

necesarios a través del metamodelo construido para luego incluirlos dentro del middleware.

Objetivos

El objetivo de este documento de evaluación es realizar un ejercicio donde se evalúe el funcionamiento correcto del middleware FHIR para el caso en el que se requiere realizar la gestión de un usuario nuevo y la generación de los archivos necesarios para incluir a un PSM dentro del middleware.

Descripción del caso

1. **Gestión de usuarios**

Hora de inicio:	**hh:mm:ss**

Existe un nuevo usuario con nivel de acceso básico que requiere ser añadido al middleware para que pueda realizar las respectivas transacciones sus credenciales serían las siguientes:

Usuario: PSMHospitalHorizonte

Contraseña: PSMHospiHori123456

Hora de fin:	**hh:mm:ss**

Hora de inicio:	**hh:mm:ss**

Por otro lado, se requiere actualizar el nombre del usuario antiguoPSM a PSMHospitalOeste y cambiar su nivel de acceso a Básico.

Hora de fin:	**hh:mm:ss**

Hora de inicio:	**hh:mm:ss**

Finalmente se desea eliminar al usuario con id 3 de la lista de usuarios.

Hora de fin:	**hh:mm:ss**

2. **Generación de archivos**

Hora de inicio:	**hh:mm:ss**

Para la transformación de los datos se estableció una clase general con ciertos atributos necesarios para la transformación a FHIR, por lo que, el funcionamiento correcto del middleware depende de la alineación de los atributos de la entidad de la base de datos del PSM y el listado generado en la sección anterior.

Para la generación de los archivos se cuenta con la siguiente información de la entidad paciente obtenida del PSM que está solicitando su adición al middleware:

Atributo en el PSM	Propiedad
Apll_mat	Apellido Materno
Apll_pat	Apellido Paterno
barr	Barrio
Cll_prncpal	Calle principal
Cll_scndria	Calle secundaria
Cntn_nac	Cantón nacimiento
Cntn_res	Cantón residencia
cel	Celular
email	Correo Electrónico
Drccn_cntct	Dirección contacto
Std_cvl	Estado civil
Fch_nac	Fecha nacimiento
gen	Género
nac	Nacionalidad
Lgr_nac	Lugar de nacimiento
Fch_nac	Fecha de nacimiento
Nmbr_apll_cntct	Nombres apellidos contacto
Num_cas	Numero casa

Num_id	Numero identificación
prentsco	Parentesco
Prrq_nac	Parroquia nacimiento
Prrq_resid	Parroquia residencia
Prim_nmbr	Primer nombre
Prov_nac	Provincia nacimiento
Prov_res	Provincia residencia
Ref_res	Referencia residencia
Seg_nmbr	Segundo nombre
tel	Teléfono
Tel_cont	Teléfono contacto
Tipo_id	Tipo identificación
https://medical-data-server.herokuapp.com/patientsNuevoHospital	Url Todos
https://medical-data-server.herokuapp.com/patientsNuevoHospital	Url Uno

Nombre de la Organización: NuevoHospital

Hora de fin:	**hh:mm:ss**

Al finalizar los ejercicios completa la siguiente encuesta:

- https://forms.gle/MJbkHzQWLDLKFwHo7