



PERÚ

Ministerio
de Salud



BICENTENARIO
DEL PERÚ
2021 - 2024

FHIR

Guías de Implementación

Oficina General de Tecnologías de la Información

NOVIEMBRE 2024

Ing. Jhon Llamas Mendoza

Guía Implementación

Las Guías de Implementación (IG) de FHIR son “un conjunto de reglas sobre cómo se resuelve un problema particular de interoperabilidad con estándares”.

Para definir estas reglas, los IG incluyen:

1. **Reglas** computables.
2. Versiones **legibles** por humanos de las reglas computables.
3. **Descripciones narrativas adicionales** de las reglas e información adicional sobre el problema que el IG está tratando de resolver.

La especificación básica FHIR es incompleta. Si bien define los elementos críticos y generalizables necesarios para una implementación FHIR, como la sintaxis y los recursos FHIR básicos, deja sin definir los detalles de implementación específicos del caso de uso.

Ejemplos de guías de implementación en FHIR:

- Guías para la implementación de registros de salud electrónicos (EHR).
- Guías para la integración de dispositivos médicos.
- Guías para la interoperabilidad en el contexto de la atención primaria o especializada.
- Guías para el intercambio de información de laboratorio, imágenes médicas o farmacovigilancia.

En resumen, las **guías de implementación de FHIR HL7** son documentos clave que proporcionan las instrucciones necesarias para adoptar FHIR de manera eficiente y efectiva, asegurando que las organizaciones puedan intercambiar información médica de manera interoperable y conforme a los estándares internacionales.

Herramienta	Descripción	Funcionalidades Clave	Ventajas
Simplifier.net	Plataforma para crear, gestionar y compartir guías de implementación FHIR.	- Crear y gestionar guías de implementación FHIR.	- Colaboración en tiempo real. - Validación y documentación fácil.
Firely Studio	Herramienta para desarrollo de perfiles FHIR y guías de implementación.	- Crear y validar perfiles FHIR. - Integración con FHIR IG Publisher y <u>Simplifier</u> .	- Soporte para varias versiones de FHIR. - Interfaz intuitiva.
FHIR IG Publisher	Herramienta para generar y publicar guías de implementación FHIR (IGs).	- Publicar guías en formatos HTML, XML y PDF. - Validación de perfiles y recursos.	- Automatiza la publicación de guías. - Soporta validación completa.
FHIR Shorthand (FSH)	Lenguaje para definir perfiles y guías de implementación FHIR de forma sencilla.	- Crear perfiles, recursos y guías de implementación usando sintaxis simplificada.	- Reducción de la complejidad al escribir perfiles. - Sintaxis legible.
SUSHI	Herramienta para convertir FSH a recursos FHIR estándar (XML/JSON).	- Convierte archivos FSH a recursos FHIR. - Facilita la creación de perfiles.	- Simplifica la creación de perfiles y recursos FHIR. - Evita trabajar con XML/JSON directamente.
FHIR Validator	Herramienta para validar que los recursos y guías FHIR cumplan con los estándares.	- Valida recursos FHIR y guías de implementación. - Asegura conformidad con el estándar.	- Validación robusta de recursos y perfiles. - Compatible con diferentes versiones de FHIR.
Postman (FHIR Collection)	Plataforma para probar <u>APIs</u> FHIR.	- Realizar pruebas de interoperabilidad entre sistemas FHIR.	- Pruebas fáciles de realizar. - Interfaz gráfica para <u>testing</u> .
FHIR TestScript	Herramienta para definir pruebas de interoperabilidad automatizadas de servicios FHIR.	- Crear y ejecutar pruebas automáticas sobre servicios FHIR.	- Automatización de pruebas. - Asegura la interoperabilidad de sistemas.
CIEL (Clinical Information Exchange Library)	Biblioteca de perfiles y guías de implementación FHIR predefinidos.	- Ofrece perfiles de interoperabilidad listos para usar.	- Proporciona perfiles validados. - Útil para sistemas de salud en desarrollo.

Guía Implementación FHIR –SUSHI

1. JAVA runtime : https://download.oracle.com/java/22/archive/jdk-22.0.1_windows-x64_bin.exe

- java version "22.0.1" 2024-04-16
- Java(TM) SE Runtime Environment (build 22.0.1+8-16)
- Java HotSpot(TM) 64-Bit Server VM (build 22.0.1+8-16, mixed mode, sharing)

- Verificación:

```
PS D:\> java -version
java version "22.0.1" 2024-04-16
Java(TM) SE Runtime Environment (build 22.0.1+8-16)
Java HotSpot(TM) 64-Bit Server VM (build 22.0.1+8-16, mixed mode, sharing)
```

2. Jekyll --> <https://jekyllrb.com/docs/installation/>

- Verificación:

```
PS D:\> jekyll -version
jekyll 4.3.3
```

3. Nodejs: <https://nodejs.org/en>

- Verificación (node y npm) :

```
PS D:\> npm --version
9.5.1
PS D:\> node --version
v18.16.0
```

4. Gestor Paquetes: npm install -g fsh-sushi

- Verificación:

```
PS D:\> sushi -v
SUSHI v3.11.1 (implements FHIR Shorthand specification v3.0.0-ballot)
```

5. Visual Studio Code: -- Extension FSH sushi - (opcional)

Vínculo de VS Marketplace: <https://marketplace.visualstudio.com/items?itemName=FHIR-Shorthand.vscode-fsh>

SUSHI es una herramienta de código abierto para escribir y generar perfiles FHIR. La herramienta toma como entrada archivos en un formato simplificado (shorthand) y los convierte en archivos de definición de recursos FHIR completos (que incluyen metadatos y restricciones específicas de los perfiles).

Crea un proyecto con *sushi init*

Define tus perfiles y extensiones en archivos *.fsh*

Genera los perfiles FHIR con el comando *sushi*.

Valida los perfiles usando herramientas de validación de FHIR.

Pública o utiliza los perfiles generados en un servidor FHIR.

```
mi-proyecto-fhir/  
|  
├── input/  
│   ├── StructureDefinition/  
│   └── ValueSet/  
|  
├── output/  
│   └── fhir/  
├── fsh.config  
└── README.md
```

[GitHub - HL7/fhir-shorthand: FHIR Shorthand](https://github.com/HL7/fhir-shorthand)

The screenshot shows a VS Code editor with a file explorer on the left and a terminal at the bottom. The file explorer shows a project structure for 'PRJ-FHIR' with files like 'patient.fsh', 'pagecontent', 'index.md', 'ignoreWarnings.txt', and various scripts. The 'patient.fsh' file is open in the editor, showing FSH code for a patient profile. The terminal shows the output of the 'sushi' command, including downloading extensions, loading packages, converting FSH to FHIR resources, and generating JSON output.

```
input > fsh > patient.fsh  
1 // This is a simple example of a FSH file.  
2 // This file can be renamed, and additional FSH files can be added.  
3 // SUSHI will look for definitions in any file using the .fsh ending.  
4 Profile: MyPatient  
5 Parent: Patient  
6 Description: "An example profile of the Patient resource."  
7 * name 1..* MS  
8  
9 Instance: PatientExample  
10 InstanceOf: MyPatient  
11 Description: "An example of a patient with a license to kill."  
12 * name  
13   * given[0] = "James"  
14   * family = "Pond"
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
info Downloaded hl7.fhir.uv.extensions.r4#5.1.0  
info Loaded package hl7.fhir.uv.extensions.r4#5.1.0  
info Checking /home/jmartin/.fhir/packages for hl7.fhir.r4.core#4.0.1...  
info Did not find hl7.fhir.r4.core#4.0.1 in /home/jmartin/.fhir/packages.  
info Downloading hl7.fhir.r4.core#4.0.1... https://packages.fhir.org/hl7.fhir.r4.core/4.0.1  
info Downloaded hl7.fhir.r4.core#4.0.1  
info Loaded package hl7.fhir.r4.core#4.0.1  
info Converting FSH to FHIR resources...  
info Converted 1 FHIR StructureDefinitions.  
info Converted 1 FHIR instances.  
info Exporting FHIR resources as JSON...  
info Exported 2 FHIR resources as JSON.  
info Assembling Implementation Guide sources...  
info Generated ImplementationGuide-demo.json  
info Assembled Implementation Guide sources; ready for IG Publisher.
```

SUSHI RESULTS

Profiles	Extensions	Logicals	Resources
1	0	0	0

Directory Structure

normal IG project structure

```
IG-root
├── ig.ini
├── input
│   ├── ignoreWarnings.txt
│   ├── images
│   │   ├── myDocument.pdf
│   │   ├── myGraphic.png
│   │   └── mySpreadsheet.xlsx
│   ├── includes
│   │   └── menu.xml
│   ├── resources
│   │   └── myOtherResources.json
│   └── pagecontent
│       ├── 1_mySecondPage.md
│       ├── 2_myThirdPage.md
│       ├── 3_myFourthPage.md
│       └── index.md
├── package-list.json
└── input-cache/publisher.jar
```

+ FSH



IG project structure + FSH

```
IG-root
├── ig.ini
├── sushi-config.yaml
├── input
│   ├── fsh
│   │   ├── file1.fsh
│   │   ├── file2.fsh
│   │   └── file3.fsh
│   ├── ignoreWarnings.txt
│   ├── images
│   │   ├── myDocument.pdf
│   │   ├── myGraphic.png
│   │   └── mySpreadsheet.xlsx
│   ├── includes
│   │   └── menu.xml
│   ├── resources
│   │   └── myOtherResources.json
│   └── pagecontent
│       ├── 1_mySecondPage.md
│       ├── 2_myThirdPage.md
│       ├── 3_myFourthPage.md
│       └── index.md
├── package-list.json
└── input-cache/publisher.jar
```



PERÚ

Ministerio
de Salud

FHIR – Guías de Países de la Región



BICENTENARIO
DEL PERÚ
2021 - 2024

Puedes encontrar varias guías de implementación en línea, algunas de las cuales se almacenan en repositorios públicos.

**MINSALUD**

Interoperabilidad de Historia Clínica - Colombia

0.4.2 - ci-build 

[Inicio](#) [Generalidades](#) [RDA](#) [Transacciones](#) [Terminología](#) [Perfiles](#) [Resumen](#)

Table of Contents > [Home](#)

Interoperabilidad de Historia Clínica - Colombia - Local Development build (v0.4.2). See the [Directory of published versions](#)

1 Home

Official URL: https://www.minsalud.gov.co/ihc/fhir/ImplementationGuide/h17.fhir.co.CoreCO	Version: 0.4.2
Draft as of 2023-10-31	Computable Name: CoreCOImplementationGuide

1.1 Antecedentes

Como antecedente de interoperabilidad en salud en Colombia, el Ministerio de Salud expidió la resolución 866 de 2021, en la cual se reglamenta el conjunto de datos mínimos relevantes para la interoperabilidad de la Historia Clínica en el país; y a su vez la ley 2015 de 2020, dispone que el Ministerio de Salud y Protección Social adoptará un plan de implementación de la Interoperabilidad de la Historia Clínica Electrónica -IHCE para el intercambio de los datos clínicos relevantes; de esa manera, el propósito de esta Guía de Implementación GI es desarrollar la especificación técnica, tomando como referencia el lenguaje común de intercambio que el Ministerio de Tecnología de Información y comunicaciones ha recomendado para la interoperabilidad de historia clínica en Colombia.

- Antecedentes
- Alcances
- Aviso legal



Guía de Implementación "cl core" FHIR R4, (Versión Evolutiva)

1.9.1 - draft 



[Inicio](#) [Objetivos](#) [Must Support](#) [Generalidades](#) [CapabilityStatement](#) [Artefactos](#) [Historial de cambios](#)

Table of Contents > [Inicio](#)

Guía de Implementación "cl core" FHIR R4, (Versión Evolutiva), published by HL7 Chile. This guide is not an authorized publication; it is the continuous build for version 1.9.1 built by the FHIR (HL7® FHIR® Standard) CI Build. This version is based on the current content of https://github.com/HL7Chile/clcore_ig/ and changes regularly. See the [Directory of published versions](#)

1 Inicio

Official URL: https://hl7chile.cl/fhir/ig/clcore/ImplementationGuide/h17.fhir.cl.clcore	Version: 1.9.1
Active as of 2024-10-01	Computable Name: clcore
Copyright/Legal: Usado con el permiso de HL7 International, todos los derechos reservados en los Licencias de HL7 Internacional.	



0.5.0 - ci-build



[Home](#) [Objetivos](#) [Actores](#) [Operaciones](#) [Artefactos](#) [Recomendaciones](#) [Terminología](#) [Seguridad](#) [Consentimiento](#) [Descargas](#)

Table of Contents > [Home](#)

FHIR_CORE_AR - Local Development build (v0.5.0). See the [Directory of published versions](#)

1 Home

1.1 Introducción

La Guía CORE-AR está basada en FHIR Versión R4 y define los requerimientos mínimos de conformidad para comunicarse con el Bus de Interoperabilidad y los requerimientos semánticos mínimos para acceder y contribuir a los registros compartidos de datos demográficos y clínicos de pacientes.


Estos perfiles son la fundación para futuras guías de implementación y fueron producidos en colaboración entre el Ministerio de Salud de la República Argentina / DNSIS y la Asociación HL7 Argentina.


1.2 Cómo leer esta guía


Esta guía está dividida en varias páginas que se muestran en la parte superior de la página en la barra de menú..

- Home: Provee la introducción acerca de esta guía.
- Objetivos: Describe los objetivos estratégicos y de interoperabilidad
- Actores: Describen a los actores (sistemas) participantes en las operaciones
- Operaciones: Casos de uso y descripción de las operaciones disponibles

- Introducción
- Cómo leer esta guía
- Acceso Directo a los Perfiles Definidos en esta Guía
- Requerimientos de Conformidad


International

FHIR Shorthand
3.0.0 - 2nd Mixed Normative-Trial Use 



[Inicio](#) [Overview](#) [Language Reference](#) [Quick Reference Card](#) [Change Log](#) [More ▾](#)

[Table of Contents](#) > [Overview](#)

FHIR Shorthand - Local Development build (v3.0.0) built by the FHIR (HL7® FHIR® Standard) Build Tools. See the [Directory of published versions](#)

2 Overview

NOTE: Information on this page is [informative content](#).

FHIR Shorthand (FSH) is a domain-specific language for defining FHIR artifacts involved in creation of FHIR Implementation Guides (IG). The language is specifically designed for this purpose, simple and compact, and allows the author to express their intent with fewer concerns about underlying FHIR mechanics. FSH can be created and updated using any text editor, and because it is text, it enables distributed, team-based development using source code control tools such as GitHub.

- FHIR Shorthand Language Basics
- Defining Items in FSH
- Line-by-Line Walkthrough
- FSH in Practice

2.1 FHIR Shorthand Language Basics

The complete FSH language is formally described in the [FHIR Shorthand Language Reference](#). Here we present just enough to get a taste of FSH.

- Grammar:** FSH has a [formal grammar](#) defined in ANTLR.
- Data types:** The primitive and complex data types and value formats in FSH are identical to the primitive types and value formats in [FHIR R4](#) and [FHIR R5](#). The types available for use when authoring are dependent upon the targeted version of FHIR.
- Whitespace:** Repeated whitespace has meaning within FSH files only within string literals and when used for [indenting rules](#). In all other contexts, repeated whitespace is not meaningful.
- Comments:** FSH uses `//` as leading delimiter for single-line comments, and the pair `/* */` to delimit multiple line comments.
- Asterisk Character:** A leading asterisk is used to denote FSH rules. For example, here is a rule to set an element named `active` to `true`:

```
* active = true
```

0.1.0 - ci-build

[Home](#) [Artifacts](#)

[Table of Contents](#) > [Artifacts Summary](#)

pruebaguia - Local Development build (v0.1.0) built by the FHIR (HL7® FHIR® Standard) Build Tools. See the [Directory of published versions](#)

2 Artifacts Summary

This page provides a list of the FHIR artifacts defined as part of this implementation guide.

- Contents:
- Structures: Resource Profiles
 - Example: Example Instances

2.0.1 Structures: Resource Profiles

These define constraints on FHIR resources for systems conforming to this implementation guide.

[MyPatient](#)

An example profile of the Patient resource.

2.0.2 Example: Example Instances

These are example instances that show what data produced and consumed by systems conforming with this implementation guide might look like.

[PatientExample](#)

An example of a patient with a license to krill.



PERÚ

Ministerio
de Salud



BICENTENARIO
DEL PERÚ
2021 - 2024



Gracias