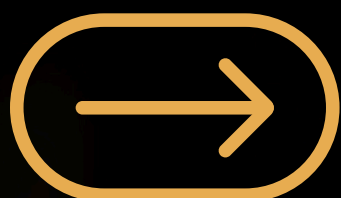


Tutorial – FHIR Implementation Guides with FHIR Shorthand (FSH) language





Introduction to FHIR

Fast Healthcare Interoperability Resources (FHIR) is the latest standard created by Health Level 7 to address the difficulty of data exchange between different health information systems.

What makes it different from its predecessors is that **it adapts to new architectural paradigms and web development technologies.**

It has met with great success, as evidenced by its adoption in interoperability improvement strategies by:

➡ **Private Companies**

➡ **Public organizations** -  [New HL7 Europe FHIR Implementation Guides to support the European Health Data Space](#)

<https://www.hl7.org/fhir>



Anatomy of a FHIR implementation guide

A FHIR Implementation guide is a set of documentation, FHIR resources and examples created to solve a specific use case within the healthcare domain.



PRACTICAL EXERCISE: visit the [HL7 FHIR Genomics Reporting Implementation Guide](#). Its use case is the standardization of the exchange of a patient's genetic information so that it can be accessed in structured data format.

In this guide we find **3 fundamental elements**:

- ✓ [documentation](#) (in markdown).
- ✓ [FHIR profiles](#) (in JSON or XML) – special FHIR resources which are used to validate the agreed FHIR structure and terminology.
- ✓ [examples of FHIR resources adapting to the established FHIR profiles](#) (in JSON or XML).

To learn more about HL7 FHIR Genomics Reporting Implementation Guide read:

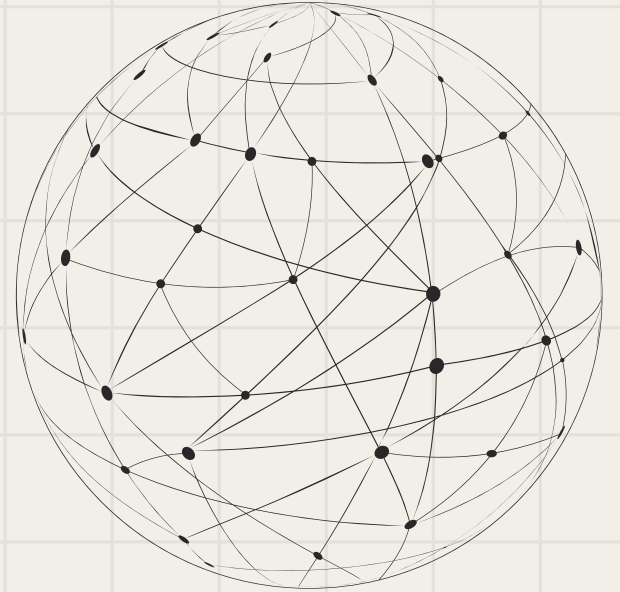
- [analysis done by the electronic Medical Records and Genomics \(eMERGE\)](#) mapping fields from different genetic reports to the guide's profiles.
- read a scientific article about this work here: [Genomic considerations for FHIR®; eMERGE implementation lessons](#)



Here a [registry of FHIR interoperability guides](#)



How FHIR implementation guides are generated



FHIR Implementation Guides are almost always presented in a standardized manner following templates that allow implementers not to get lost among so much documentation created by different organizations and vendors.

The screenshot shows the HL7 CH Core (R4) FHIR Implementation Guide page. The header includes the HL7 logo and 'CH Core (R4) 6.0.0 - trial-use'. The main content area has a red navigation bar with links: Home, Guidance, Profiles, Extensions, and Artifacts. Below the navigation bar, there's a 'Table of Contents' section. A yellow banner states: 'This page is part of the CH Core (R4) (v6.0.0: STU 6) based on FHIR (HL7® FHIR® Standard) R4. This is the current published version. For a full list of available versions, see the Directory of published versions'. The '1 Home' section contains a table with the following information:

Official URL: http://fhir.ch/ig/ch-core/ImplementationGuide/ch.fhir.ig.ch-core	Version: 6.0.0
Active as of 2025-12-16	Computable Name: CH_Core
Copyright/Legal: CC0-1.0	

Below the table, the '1.1 Introduction' section is visible, starting with 'This implementation guide is provided to support the use of FHIR® in Switzerland.' A sidebar on the right lists the table of contents: Introduction, Relation to the Swiss EPR, Scope, Governance, Collaboration, Safety Considerations, IP Statements, Cross Version Analysis, Dependency Table, and Globals Table.

Explore [official documentation Publisher](#) for generating FHIR interoperability guides

Source: <https://fhir.ch/ig/ch-core/index.html>

To generate these implementation guides, an **open-source tool maintained by the FHIR community itself is used (Publisher)**, which generates this web page ready to deploy from the 3 fundamental elements of the guides we listed earlier.



Scalable FHIR Implementation Guides



A FHIR interoperability guide is a software project.

To ensure that all developments derived from it are robust and maintainable over time, **its components must be managed using source code control.**

This allows implementers:

- ✓ following the guide to track its evolution
- ✓ adapt to new version changes
- ✓ and maintain a clear lineage between the guide and the implementations.



FHIR profiles in a implementation guide

The FHIR profile is the most important element of an implementation guide. It defines the syntactic and semantic structure of a dataset for a specific use case.



PRACTICAL EXERCISE: Check [Matchbox FHIR resource validation tutorial](#) to understand how the FHIR profiles are used.



EXAMPLE: US CORE PATIENT

Name	Flag	Card	Type	Description & Constraints
Patient		0..*	Patient	care services
us-core-race		0..1	(Complex)	US Core Race Extension URL: http://hl7.org/fhir/us/core/StructureDefinition/us-core-race
us-core-ethnicity		0..1	(Complex)	US Core ethnicity Extension URL: http://hl7.org/fhir/us/core/StructureDefinition/us-core-ethnicity
us-core-birthsex		0..1	code	Extension URL: http://hl7.org/fhir/us/core/StructureDefinition/us-core-birthsex Binding: Birth Sex (required): Code for sex assigned at birth
Identifier	S	1..*	Identifier	An identifier for this patient
system	S	1..1	uri	The namespace for the identifier value
value	S	1..1	string	The value that is unique within the system.
name	S	1..*	HumanName	us-core-8: Either Patient.name.given and/or Patient.name.family SHALL be present or a Data Absent Reason Extension SHALL be present.
family	S	0..1	string	Family name (often called "surname")
given	S	0..*	string	Given names (not always "first"). Includes middle names
suffix	S	0..*	string	Parts that come after the name
period		0..1	Period	Time period when name was/is in use
telecom		0..*	ContactPoint	A contact detail for the individual
system	S			Binding: ContactPointSystem (required): Telecommunications form for contact point.
value	S	1..1	string	The actual contact point details
use	S	0..1	code	home work temp old mobile - purpose of this contact point Binding: ContactPointUse (required)
gender	S	1..1		Binding: AdministrativeGender (required)
birthDate	S	0..1	date	The date of birth for the individual
address	S	0..*	Address	An address for the individual
line	S	0..*	string	Street name, number, direction & P.O. Box etc.
city	S	0..1	string	Name of city, town etc.
state	S	0..1	string	Binding: USPS Two Letter Alphabetic Codes (extensible): Two Letter USPS alphabetic codes. US Zip Codes
postalCode	S	0..1	string	
period	S	0..1	Period	Time period when address was/is in use
communication		0..*	BackboneElement	A language which may be used to communicate with the patient about his or her health
language	S	1..1	CodeableConcept	The language which can be used to communicate with the

Parent Resource / Profile

Extension

Invariant

Must Support Flag

Cardinality Constraint

Value Set Binding
Short Description

Source:
<https://confluence.hl7.org/spaces/FHIR/pages/265100106/FSH+Tooling+Community+Kick+Off+Webinar>



Maintaining FHIR profiles over the time

Maintaining the FHIR profiles in JSON or XML has been quite a challenge for FHIR experts. **Over time, FHIR profiles have been maintained:**

- Directly editing the JSON or XML by hand.
- Excel spreadsheets that later generate the json or xml structures.
- graphical interfaces that modifies the json or xml directly.
- **FHIR Shorthand language** – a programming language that through a Javascript library called **sushi** generates the json or xml structures.


Source: <https://build.fhir.org/ig/HL7/fhir-shorthand>

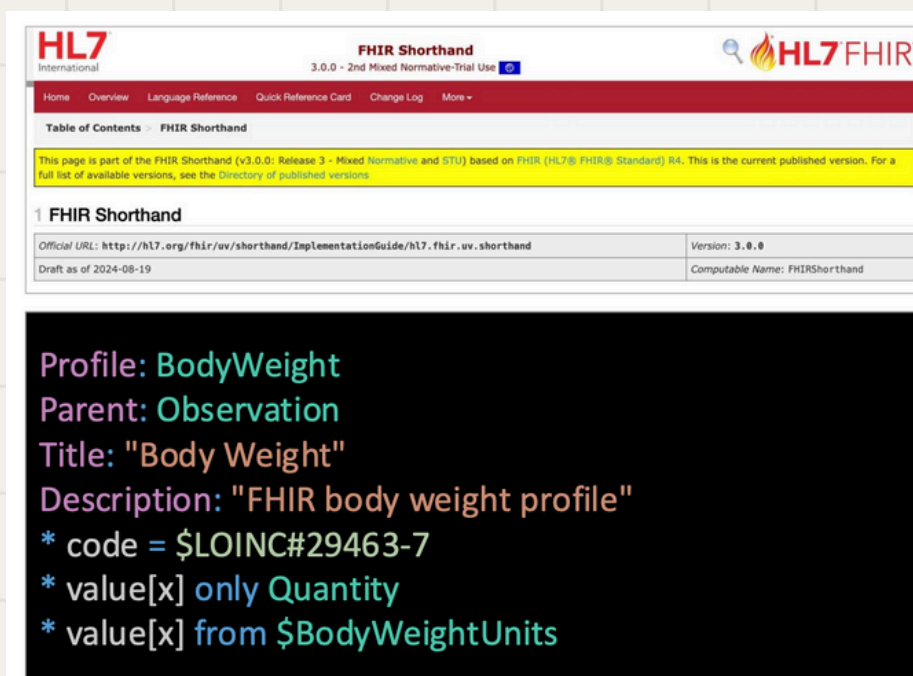


FHIR Shorthand language

Allows managing implementation guides like any software project:

- ✓ it reduces the complexity of managing FHIR profiles in JSON and XML
- ✓ enabling the introduction of CI/CD and source code control
- ✓ not dependent on OS

 Here a [registry of FHIR interoperability guides using FSH](#)



The screenshot shows the HL7 FHIR Shorthand web interface. The header includes the HL7 logo and navigation links. The main content area displays a FHIR profile definition for 'BodyWeight' under the 'Observation' parent. The profile title is 'Body Weight' and the description is 'FHIR body weight profile'. The profile includes a code from the LOINC system and specifies that the value is a quantity from the BodyWeightUnits.


```
Profile: BodyWeight
Parent: Observation
Title: "Body Weight"
Description: "FHIR body weight profile"
* code = $LOINC#29463-7
* value[x] only Quantity
* value[x] from $BodyWeightUnits
```

 Concise, readable, understandable

 Rapid changes via text operations

 Collaborative authoring w/ source code control

 Error checking & application of best practices

 Combine with other profiling approaches

HL7 Balloted Standard, built into IG Publisher

Source:

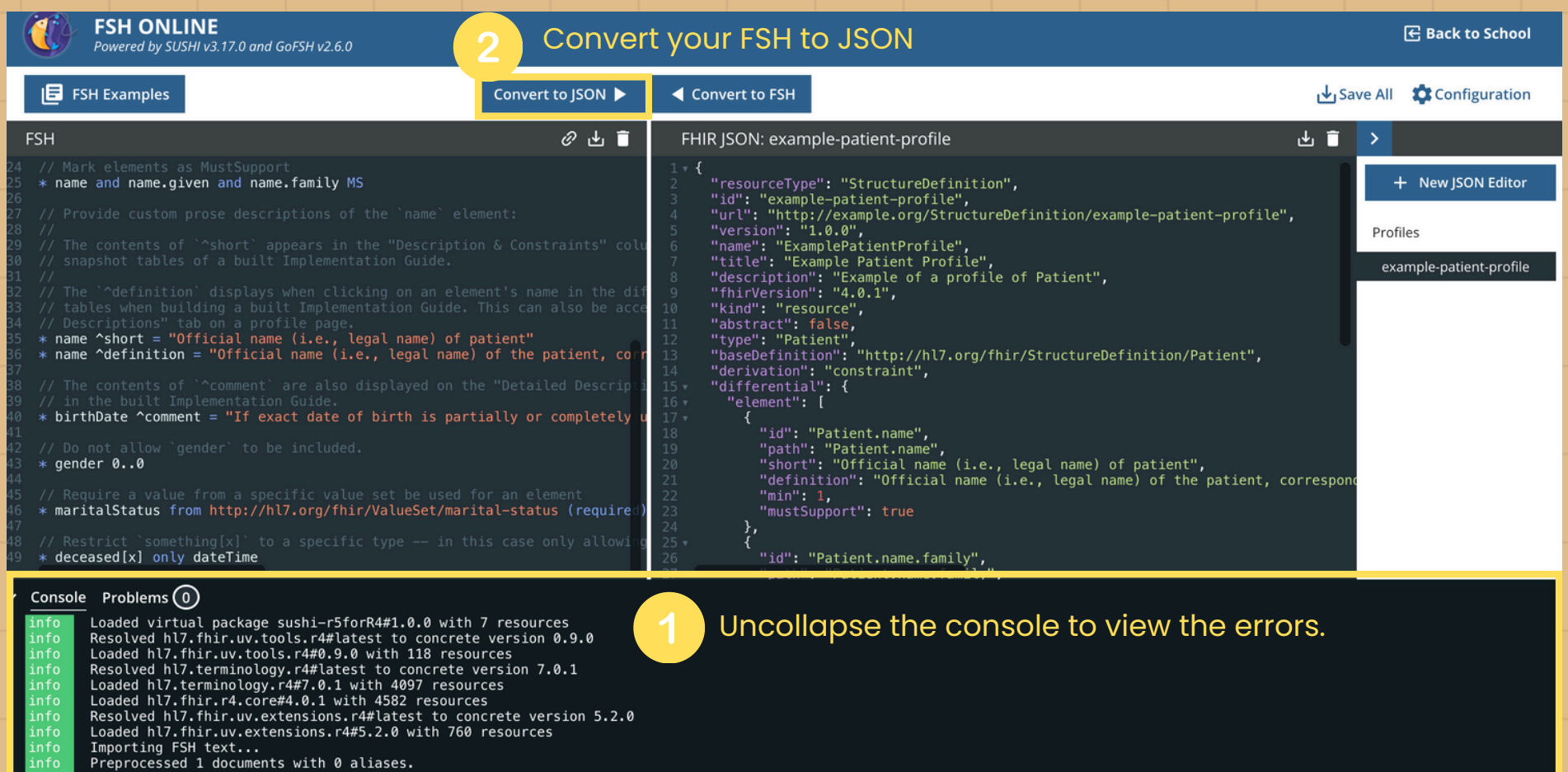
<https://confluence.hl7.org/spaces/FHIR/pages/265100106/FSH+Tooling+Community+Kick+Off+Webinar>



Start with FHIR Shorthand language

The go-to resource for learning the FHIR Shorthand language is the official [FSH School website](#) and [installing FHIR Shorthand](#) is quite straightforward.

If you want to start learning the language **without any installation**, I recommend using [FSH Online tool](#). **In our tutorial, we will use this tool, and the following image shows the features you should learn to use marked in yellow.**



Source:

<https://fshonline.fshschool.org/#!/>

LET'S GO!





Create a FHIR profile

PRACTICAL EXERCISE: Using [FSH Online](#), define a Practitioner profile named MyPractitioner. This profile must support the name, identifier, and qualification (to specify the specialty) fields. The identifier element should be sliced to support various types (e.g., Passport, National ID), but it must mandate exactly one 'License Number' with the fixed system `http://example.com/myLicenseNumberIdentifier`. Finally, generate a compliant Instance of this profile.

click here to see the [PROFILE SOLUTION](#)

click here to see the [INSTANCE SOLUTION](#)

click here to see what happens when creating an instance that does not follow the rules ([WRONG INSTANCE](#))

3

EXTRA TIP: Go to the left corner of the FSH Online, click "FSH Examples" and inspire on the examples to do the exercises!



FSH Examples

In this exercise you have learned to:

- ✓ define supported fields for your profile with respect to a reference FHIR resource,
- ✓ set a field as required,
- ✓ link a field to a specific text pattern,
- ✓ and to perform the technique called FHIR slicing*.

FHIR SLICING is a mechanism that allows you to take a repeating element (such as a list of identifiers) and "segment" it into specific groups based on defined criteria. What is it for? Imagine you have the Patient.identifier element. By default, it is just a list of IDs. With slicing, you can say: "I want this list to mandatorily have one element that is a National ID, and optionally, other elements that are Passports." You are defining different rules for different elements within the same list.





Make your FHIR profile work with terminologies

PRACTICAL EXERCISE: Using [FSHOnline](#), create a FHIR value set with the following SNOMED codes (system = <http://snomed.info/sct>):

- 394579002 "Cardiology"
- 394588006 "Pediatric oncology"
- 394593009 "Medical oncology"

and bind it to the field qualification in required mode. Finally, generate a compliant Instance of this profile.

click here to see the [PROFILE SOLUTION](#)

click here to see the [INSTANCE SOLUTION](#)

click here to see what happens if we create an instance that does not follow the rules ([WRONG INSTANCE](#)) 😊 do not panic !

Even though we are using a code that doesn't exist in SNOMED and isn't included in our ValueSet, FHIROnline doesn't complain. Why? Because to perform terminology validation, you need the FHIR validator, not SUSHI. In the following link, I'm sending you an [example of a validation in Matchbox](#) where a code is used that exists neither in SNOMED CT nor in the required ValueSet.



Feature	SUSHI Validation	FHIR Validator
Primary Goal	Validates FSH (Shorthand) syntax and logic.	Validates FHIR Resources (JSON/XML) against profiles.
Input Type	.fsh files (Shorthand code).	.json or .xml (Patient, Observation, etc.).
Validation Level	Basic (Paths, cardinalities, syntax).	Deep (Terminology, FHIRPath, slicing).
When to Use	During Development (authoring profiles).	During Testing/Production (checking data).
Core Engine	TypeScript-based compiler.	Java-based reference validator.
Example	FSH Online tool	Matchbox

In this exercise you have learned to:

- ✓ create a value set,
- ✓ bind a value set to a field of a profile,
- ✓ and differences between Sushu Validation and FHIR Validation.





Customize your FHIR profile with extensions

PRACTICAL EXERCISE: Using [FSHOnline](#), create a FHIR extension where I can place the years of experience of the Practitioner. Finally, generate a compliant Instance of this profile.

click here to see the [PROFILE SOLUTION](#)

click here to see the [INSTANCE SOLUTION](#)

The creation of both new profiles and new extensions should be thought through first. Someone might have had the same use case before you.



Here a [registry of FHIR extensions](#)

In this exercise you have learned to:

- ✓ create an extension,
- ✓ link the extension to a specified FHIR profile,
- ✓ and where to find extensions.





Create your FHIR profile inheriting from another FHIR profile

PRACTICAL EXERCISE: Using [FSHOnline](#), go to [Patient resource of International Patient Summary \(IPS\) FHIR Implementation Guide](#), copy the json resource and transform it to .fsh. Add you own profile called My Patient, inherit from IPS Patient and determine as must support language, gender, name and birthdate.

click here to see the [PROFILE SOLUTION](#)

click here to see the [INSTANCE SOLUTION](#)

click here to see what happens if [we validate our instance directly inheriting from International Patient Summary Implementation Guide](#) against a FHIR Validator.



🔍 We have just defined the attributes that we are going to populate for our use case and our profile is compliant and at the same with this known Implementation Guide.

In this tutorial
we recycle
FHIR profiles



In this exercise you have learned to:

- ✓ create a profile from another FHIR profile
- ✓ and to follow another FHIR profile at the same time that we add our own specifications.





Customize your FHIR profile with invariants

PRACTICAL EXERCISE: go to the [Patient resource from International Patient Summary in fsh](#), scroll down and identify one invariant and how you can link that invariant to a profile. That is an invariant

```
Invariant: ips-pat-1
Description: "Patient.name.given, Patient.name.family or Patient.name.text SHALL be present"
* severity = #error
* expression = "family.exists() or given.exists() or text.exists()"
* xpath = "f:given or f:family or f:text"
```

This invariant tells you that if you populate the name field you have to populate the field family, given or text.

check this example that has just populated name.use in FHIROnline tool, sushi validation is not complaining because is complaint with FHIR but...

click here to [validate the instance in the FHIR Validator](#).

FHIR INVARIANTS or constraints are formal rules that define business logic or clinical requirements that cannot be captured by simple data structures or cardinality alone. FHIRPath is the standard language used to write the logic for invariants.

Explore [official documentation of FHIRPath](#)

In this exercise you have learned to:

- ✓ recognize an invariant,
- ✓ link an invariant to a profile,
- ✓ and validate an invariant.



🌟 Extra points section: let's build our Implementation Guide with FSH profiles

- Install [Publisher requirements](#)
- Install [sushi requirements](#)
- Open your terminal:

```
>sushi init  
>cd ExampleIG  
>./_updatePublisher.sh  
>./_genonce.sh
```

In sushi init leave everything by default
In ./updatePublisher.sh say YES !

- Go to output folder and open index.html 😊
- Take one of the .fsh files of the tutorial, create a new .fsh file and place it in input/fsh folder. Run `>./_genonce.sh` and open /output/index.html.

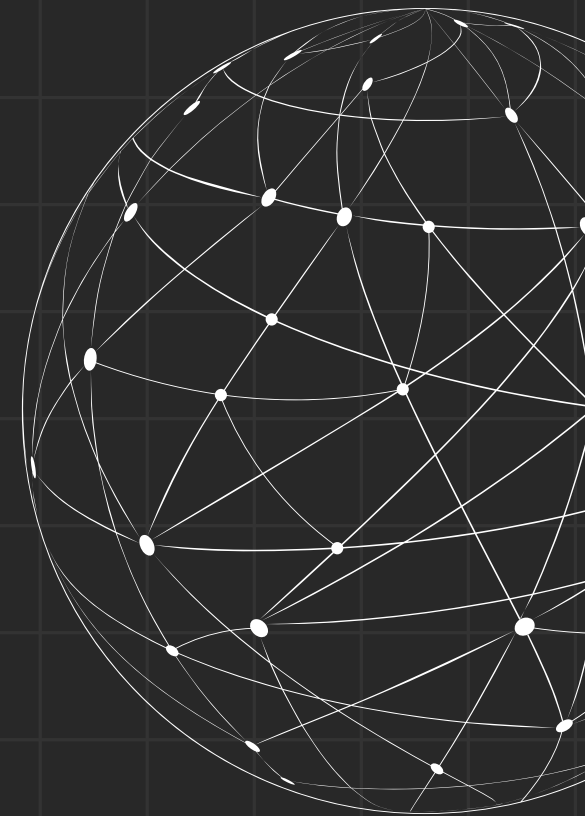
In this exercise you have learned to:

🔥 build a FHIR Implementation Guide with FHIR Shorthand language FHIR profiles.



From theory to practice: sharing direct lessons to drive digital health.

More information at yolandasabuco.io



Was this post
Helpful?

Contact me:



yolandasabuco.io



[yolanda-sabuco-garcia](https://www.linkedin.com/in/yolanda-sabuco-garcia)



[yoliyu](https://github.com/yoliyu)



info@yolandasabuco.io

