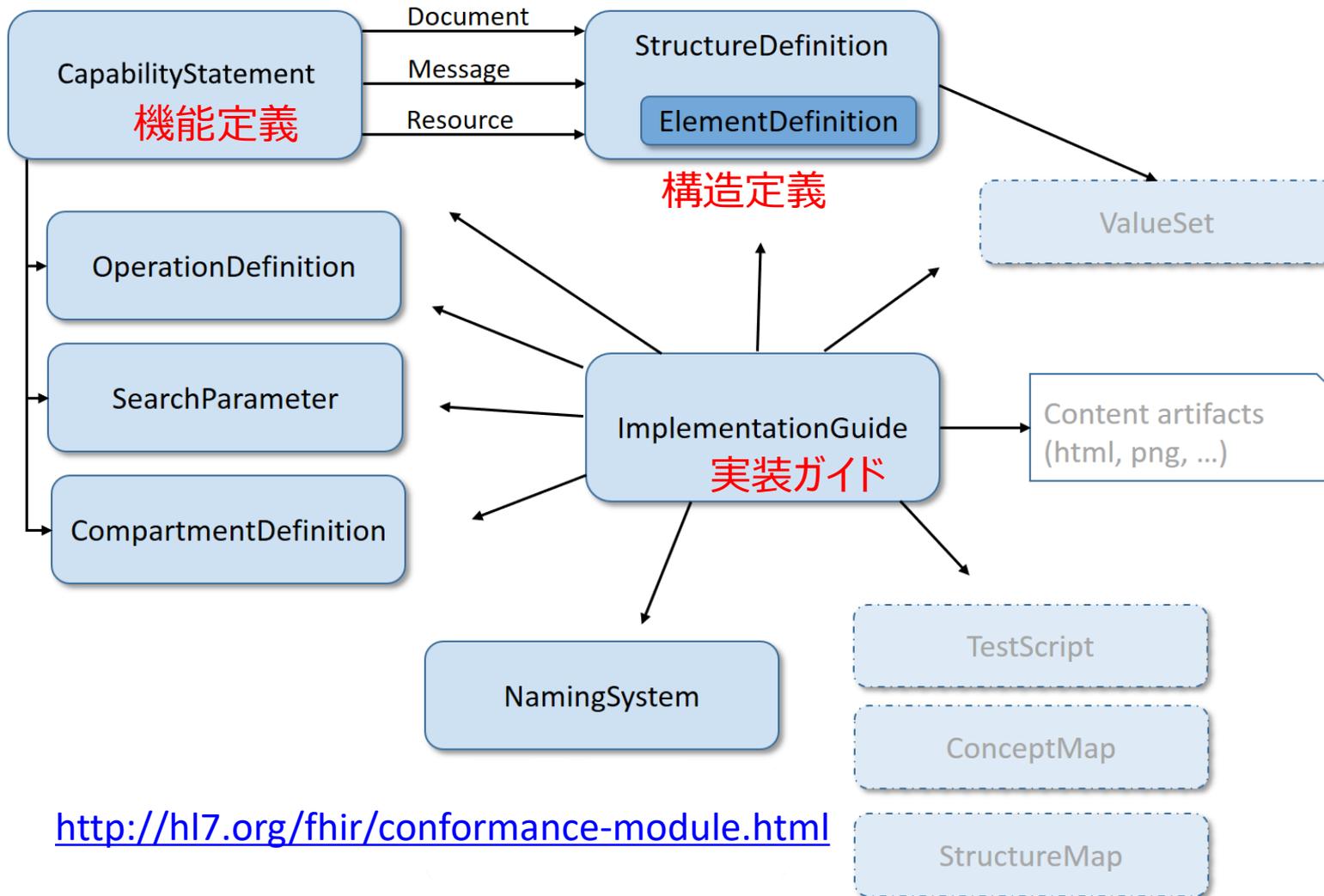


日本HL7協会第85回HL7セミナー

HL7 FHIR JP Core 実装ガイドの作り方 環境編

株式会社ファインデックス
宮川 力

実装ガイドの構造



<http://hl7.org/fhir/conformance-module.html>

HL7 FHIR JP Core Implementation Guide
1.1.1 - release

FHIRJP Guidances FHIRContents Security Artifacts Download

Table of Contents > HL7 FHIR JP Core 実装ガイド

HL7 FHIR JP Core Implementation Guide - Local Development build (v1.1.1). See the Directory of published versions

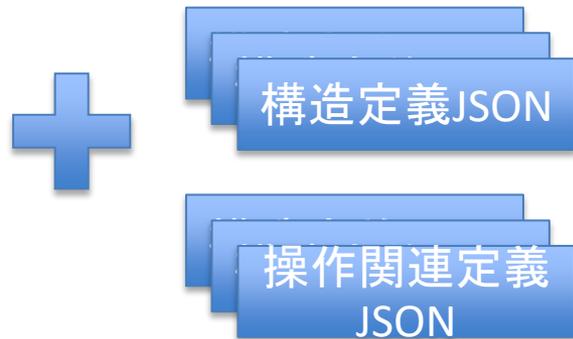
HL7 FHIR JP Core 実装ガイド

項目	内容
5.38.1	Resource Profile: JP Core Patient Profile
5.29.1	Resource Profile: JP Core MedicationRequest Profile
5.38.2	Structure Definition: JP Core MedicationRequest Profile

このドキュメントは日本HL7協会による公開バージョン履歴 (Version) History

Date	IG Version	FHIR Version	Description	Package
Current Versions				
2022-11-02	1.1.1	4.0.1	正式に公開された最新バージョンの実装ガイド	
R4 Sequence(current)				
2022-11-02	1.1.1-dev	4.0.1	最新版のプレ開発版	

パッケージファイル



- 人間にとって読みやすい自然言語記載された説明(HTML形式)
- 機械可読な定義情報(JSON,XML形式)
 - 構造定義
 - 操作関連定義

<https://jpfhir.jp/fhir/core/1.1.1/index.html>

<https://jpfhir.jp/fhir/core/index.html>

1. 構造化データやHTMLファイルを直接編集
 - バリデーション等がないため品質が心配。
 - リンク等の整合性を合わせるのは労力を必要とする。
2. 商用製品による編集
 - (Firely社のsimplifier.net[<https://simplifier.net/>]などが有名)
 - GUIを用いて初心者でも記載がしやすい
 - アップロードするとすぐにIGが更新される
 - 有償である（編集人数応じたライセンス）
3. Sushi & IG Publisherを用いた編集
 - 無償
 - FSHファイルを用いた少ない記述。
 - 自動デプロイまでの環境構築手間がかかる。

方法	構造化データやHTMLファイルを直接編集	商用製品	Sushi & IG Publisher
ツール利用コスト	○	×	○
学習コスト	△ (JSON構造理解)	○	△ (FSHの知識必要)
環境構築コスト	○	○	△ (git・自動デプロイ環境構築)
編集コスト	×	△ (一部JSON編集必要、リンク整合性)	○
バリデーション	×	○	○
全体コスト	×	×	○ (編集コスト膨大)

現在はSushi & IG Publisherが主流
商用ツールは初心者 & 少人数向き。

<https://simplifier.net/guide/jpcorev1/fhirjp>

HL7 FHIR JP Core 実装ガイド <Draft Ver.1> 2021.12.26

Copyright©2021 by FHIR® Japanese implementation research working group in Japan Association of Medical Informatics (JAMI). All rights reserved.

このドキュメントは日本医療情報学会(JAMI) NeXEHRs課題研究会「HL7®FHIR® 日本実装検討WG」で作成した実装ガイドのドラフトVer.1です。このバージョンは日本HL7協会による承認を受けていません。今後、予告なく内容に変更があります。実装や利用は全て自己責任で行ってください。

This Implementation Guide is for defining minimum requirement and constrains of conformance based on HL7 FHIR specification to access health and healthcare information in Japan. This is named as "JP Core Implementation Guide" (JP Core). JP Core is created and described through a lot of considerations and discussions in FHIR® Japanese implementation research working group under the Japan Association of Medical Informatics (JAMI). This version is only for public comments of Draft Version1 Release. Do not use for implementation of production operation system. Please use at your own risk and as it is.

Packageのダウンロード: [【zip版】](#) [【GitHubへ】](#)

概要

- 1. ガイダンス: JP Coreでの全体に関わる規則や注意事項を記載しています。
 - 1.1. 総合ガイダンス
 - 1.2. CardinalityとMust Supportの組み合わせ
 - 1.3. 欠損値の扱い
 - 1.4. 文字コード
 - 1.5. 検索
- 2. JP Core FHIRコンテンツ: JP Coreで利用するFHIRの詳細について記載をしています。
 - 2.1. Profiles (プロファイル)
 - 2.1.1. Administration (運営管理)
 - 2.1.1.1. JP Core Patient (患者) プロファイル
 - 2.1.1.2. JP Core Coverage (保険・公費) プロファイル
 - 2.1.1.3. JP Core Encounter (来院・入院) プロファイル
 - 2.1.1.4. JP Core Location (所在場所) プロファイル
 - 2.1.1.5. JP Core Organization (組織) プロファイル
 - 2.1.1.6. JP Core Practitioner (医療従事者) プロファイル
 - 2.1.1.7. JP Core PractitionerRole (医療従事者役割)プロファイル
 - 2.1.2. Medication (薬剤リスト)
 - 2.1.2.1. JP Core Medication (薬剤) プロファイル
 - 2.1.2.2. JP Core MedicationRequest (内服・外用薬剤処方) プロファイル
 - 2.1.2.3. JP Core MedicationRequest Injection (注射薬剤処方) プロファイル
 - 2.1.2.4. JP Core MedicationDispense (内服・外用薬剤処方調剤・払い出し記録) プロファイル

JP Core v1.0.5まで
simplifier.netを用いて作成
<https://simplifier.net/>

<https://jpfhir.jp/fhir/core/1.1.1/index.html>

JAMI HL7 FHIR JP Core ImplementationGuide 1.1.1 - release

FHIRJP Guidances FHIRContents Security Artifacts Download

Table of Contents > HL7 FHIR JP Core 実装ガイド

HL7 FHIR JP Core ImplementationGuide - Local Development build (v1.1.1). See the [Directory of published versions](#)

HL7 FHIR JP Core 実装ガイド

項目	内容
定義URL	http://jpfhir.jp/fhir/core/ImplementationGuide/hl7.fhir.jp.core
Version	1.1.1
Name	FHIRJPCoreImplementationGuide
Title	HL7 FHIR JP Core ImplementationGuide
Status	Active (2022-11-02)
Copyright	Copyright FHIR Japanese implementation research working group in Japan Association of Medical Informatics (JAMI) 一般社団法人日本医療情報学会NeXEHRs課題研究会FHIR日本実装検討WG

このドキュメントは日本医療情報学会NeXEHRs課題研究会「HL7®FHIR® 日本実装検討WG」で作成した実装ガイドのドラフトVer.1.1.1である。このバージョンは日本HL7協会による承認を受けていない。今後、予告なく内容に変更がある。また実装や利用は全て自己責任で行なうこと。

1.1 概要

ガイダンス: JP Coreでの全体に関わる規則や注意事項を記載している。

- 総合ガイダンス
- Must SupportとCardinality(多重度)のルール
- 欠損値の扱い
- 文字コード
- 検索
- 利用上の注意事項
- 更新履歴

1.2 JP Core FHIRコンテンツ

JP Core FHIRコンテンツ: JP Coreで利用するFHIRの詳細について記載をしている。

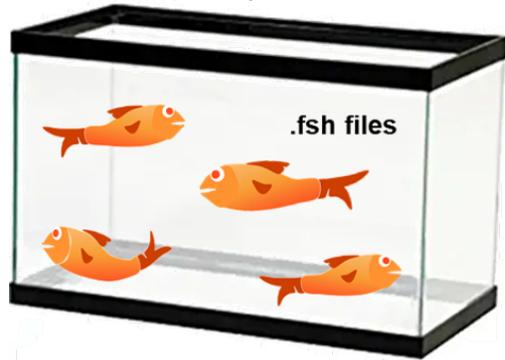
1.2.1 Profiles (プロファイル)

- Administration (運営管理)
 - JP Core Patient (患者) プロファイル
 - JP Core Coverage (保険・公費) プロファイル
 - JP Core Encounter (来院・入院) プロファイル
 - JP Core Location (所在場所) プロファイル
 - JP Core Organization (組織) プロファイル
 - JP Core Practitioner (医療従事者) プロファイル
 - JP Core PractitionerRole (医療従事者役割)プロファイル
- Medication (薬剤リスト)
 - JP Core Medication (薬剤) プロファイル
 - JP Core MedicationRequest (内服・外用薬剤処方) プロファイル
 - JP Core MedicationRequest Injection (注射薬剤処方) プロファイル
 - JP Core MedicationDispense (内服・外用薬剤処方調剤・払い出し記録) プロファイル

JP Core v1.1.0以降は
sushi & IG Publisherにて作成、
Visual Studioのプラグイン等をスペル
チェック、一括変換やエラー、警告の通知
等の機能が充実している

Shorthand

1 FSH files
/input/fsh



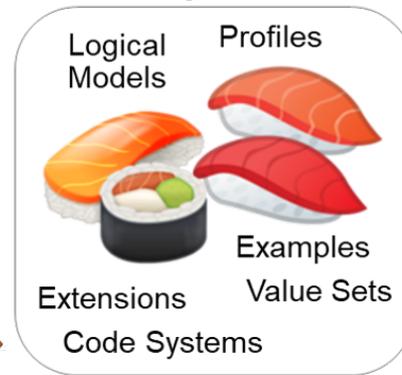
+

2 SUSHI configuration information
(sushi-config.yaml)



3

4 FHIR JSON
/fsh-generated



+

5 IG Data
(Pages, menus, images, etc.)
IG configuration Information
(package.json, ig.ini, IG resource)

HL7 FHIR IG Publisher

6



Implementation Guide

7

Runs SUSHI if /input/fsh exists

Credits: Sushi clipart from Google and WhatsApp rendering of Unicode 6.0 sushi emoji, Sushi menu from PNGWave, Non-Commercial Use, no attribution required (<https://www.pngsave.com/png-clip-art-oxer/>)

<http://hl7.org/fhir/uv/shorthand/overview.html>

- オープンソースとして公開されて、誰でも自由に利用可能。
- FHIR SHORTHAND形式という簡易な文法にて記述
- FHIR Shorthand形式から構造データ(JSON)を生成
- 上記JSONファイルとMarkdown形式で記載された自由記述部分より実装ガイド作成
- スクリプトにより、一連の流れを自動化が可能

① FISHファイル(.FSH)	FHIRリソース記述言語で記載されたソースファイル
② Configurationファイル	Sushiプログラム構成ファイル フォルダ構成やIG Publisher呼び出しの定義等を記載します。
③ Sushi	FSH→JSON変換プログラム FSHの整合性をチェックしながら、JSONファイルを生成します。
④ FHIR JSONファイル	Sushiによって生成されたFHIRの構造・操作関連定義ファイル。
⑤ IG Data & Configuration	構造・操作関連定義ファイル以外のHTML元となるMarkdownファイルやイメージファイル、またIGの構成ファイル等。
⑥ IG Publisher	④⑤を元にIG用のHTMLファイルやJSONファイルを配置します。
⑦ Implementation Guide	完成物である実装ガイド

ビルド環境作成

- OS
 - Windows10
LinuxやMacでも動作可能。
- インストールが必要なライブラリ
 - Node.js(npm含む)
 - Java SDK
 - Ruby(gem含む)
 - Jekyll
 - Visual Studio Code
 - FSH Extension
 - Markdown Extension
 - yaml Extension

環境変更によるリスクがある場合は、新たな環境を用意するようにしてください。



Node.js®はクロスプラットフォームに対応したオープンソースのJavaScript実行環境です。

New security releases to be made available June 20th, 2023

ダウンロード Windows (x64)

18.16.0 LTS

推奨版

20.3.0 最新版

最新の機能

[他のバージョン | 変更履歴 | APIドキュメント](#) [他のバージョン | 変更履歴 | APIドキュメント](#)

サポートされているバージョンに関する詳しい情報：リリーススケジュール。

<https://nodejs.org/ja/>

- 環境に合わせてNode.jsをインストールすること。
- LTSバージョンの最新がおすすめ。
- ダウンロード先
<https://nodejs.org/ja/>
- インストール方法はインターネットを参照のこと

※管理者権限にて実行すること。

- Node.jsインストール確認

コマンドプロンプトより下記を実行して表示されることを確認して下さい。

```
> node --version  
> npm --version
```

- Sushiのインストール

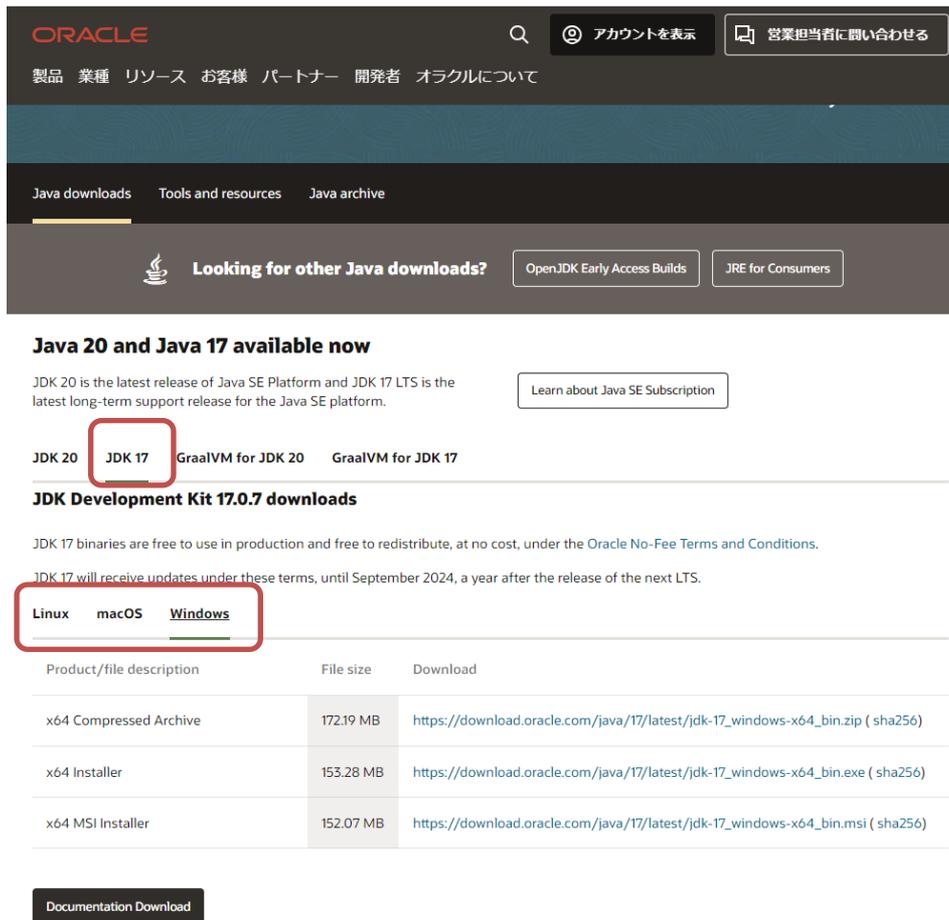
コマンドプロンプトより下記をsushiをインストールしてください。

```
> npm install -g fsh-sushi
```

- Sushiインストール確認

コマンドプロンプトより下記を実行して表示されることを確認して下さい。

```
> sushi -v
```



ORACLE

製品 業種 リソース お客様 パートナー 開発者 オラクルについて

Java downloads Tools and resources Java archive

Looking for other Java downloads? OpenJDK Early Access Builds JRE for Consumers

Java 20 and Java 17 available now

JDK 20 is the latest release of Java SE Platform and JDK 17 LTS is the latest long-term support release for the Java SE platform.

JDK 20 **JDK 17** GraalVM for JDK 20 GraalVM for JDK 17

JDK Development Kit 17.0.7 downloads

JDK 17 binaries are free to use in production and free to redistribute, at no cost, under the Oracle No-Fee Terms and Conditions.

JDK 17 will receive updates under these terms, until September 2024, a year after the release of the next LTS.

Linux macOS **Windows**

Product/file description	File size	Download
x64 Compressed Archive	172.19 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.zip (sha256)
x64 Installer	153.28 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.exe (sha256)
x64 MSI Installer	152.07 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.msi (sha256)

Documentation Download

- 環境に合わせてJava SDKをインストールをすること。
- 推奨バージョンは **ver17**のLTS version。(2023年6月現在)
<https://confluence.hl7.org/display/FHIR/IG+Publisher+Documentation#IGPublisherDocumentation-QuickStart>
- インストールを実行すると環境変数(PATH)が自動的に設定される。

- Jekyll(ジキル)とは
Jekyllは静的サイトの生成を行うための、Ruby製のツールです。
- インストール全般
<http://jekyllrb-ja.github.io/docs/installation/>
- Windows場合
<http://jekyllrb-ja.github.io/docs/installation/windows/>
 - Ruby+Developのインストール
<https://rubyinstaller.org/downloads/>
Ruby+Devkit 3.1.X (x64)をダウンロード後、インストールを管理者にて実行します。
途中で**ridk installステップ**を質問がありますが、選択し実行してください。
 - Jekyllのインストール
改めてコマンドプロンプトを管理者にて開く。※環境変数反映のため。

```
> gem install jekyll bundler
```
 - インストール確認

```
> jekyll -v
```

- 高機能なソースコードエディタ
 - 無償提供(MITライセンス)
 - Linux,Mac,Windows等でも動作可能
 - Extension導入で様々なプログラム言語の編集が可能
- インストーラーのダウンロード
<https://code.visualstudio.com/>
- インストール方法 (参考)
<https://www.javadrive.jp/vscode/install/index1.html>
管理者権限にて実行すること。

Visual Studio Code Docs Updates Blog API Extensions FAQ Learn

Version 1.63 is now available! Read about the new features and fixes from November.

Code editing. Redefined.

Free. Built on open source. Runs everywhere.

[Download for Windows](#)
Stable Build

Other platforms and Insiders Edition

By using VS Code, you agree to its [license and privacy statement](#).

The screenshot shows the Visual Studio Code interface with a code editor displaying a JSON file named `JP_Patient_withExtensions.fsh`. The file explorer on the left shows a project structure for `JP CORE DRAFT MAIN` with various FHIR extension files. The code editor shows the following content:

```

1 Extension: JP_Patient_Race
2 Description: "人種 (Race)に関する情報を格納するためのExtension。"
3 *meta.lastUpdated = "2021-11-30T10:09:36.662+00:00"
4 *url = "http://jp.fhir.jp/fhir/core/Extension/StructureDefinition/JP_Patient_Race"
5 *version = "1.0.0"
6 *date = "2021-11-30T09:57:31.0583757+00:00"
7 *context.type = #element
8 *context.expression = "Patient"
9 *short = "人種に関する情報"
10 *comment = "人種に関する情報を表現する拡張。 \n\nValueSetとして \n\nhttp://terminology.hl7.org/CodeSystem/v3-Race \n\nを使用"
11 *url = "http://jp.fhir.jp/fhir/core/Extension/StructureDefinition/JP_Patient_Race" (exactly)
12 *value[x] only Coding
13 *value[x].system = "urn:oid:2.16.840.1.113883.5.104" (exactly)
14
15 Profile: JP_Patient
16 Parent: Patient
17 Id: JP_Patient
18 Title: "JP_Patient"
19 Description: "このプロファイルはPatientリソースに対して、患者のデータを送受信するための基礎となる制約と拡張を定めたものである。"
20 // extension 参照宣言 基底拡張2個、内部拡張1個
21 * extension contains $patient-religion named religion 0..*
22 * extension contains $patient-birthPlace named birthPlace 0..1
23 * extension contains JP_Patient_Race named Race 0..*
24 //
25 *url = "http://jp.fhir.jp/fhir/core/StructureDefinition/JP_Patient"
26 *version = "1.0.0"
27 *date = "2021-12-01T03:29:13.2472819+00:00"
28 *short = "Information about an individual or animal receiving health care services. 医療サービスを受けている個人または動物"
29 *definition = "Demographics and other administrative information about an individual or animal receiving care or other h"
30 *extension ^slicing.discriminator.type = #value
31 *extension ^slicing.discriminator.path = "url"
32 *extension ^slicing.rules = #open
33 *extension ^min = 0
34 *extension[religion] only $patient-religion
35 *extension[religion] ^sliceName = "religion"
36 *extension[religion] ^short = "患者の宗教"
37 *extension[religion] ^definition = "The patient's professed religious affiliations. \n\n患者の公言された宗教的所属。"
38 *extension[religion] ^comment = "患者の宗教をValueSet(v3.ReligiousAffiliation)より選択する。輸血や食事で考慮が必要な場合がある"
39 *extension[religion] ^min = 0
40 *extension[religion] ^mustSupport = false
41 *extension[birthPlace] only $patient-birthPlace
42 *extension[birthPlace] ^sliceName = "birthPlace"

```

IntelliSense

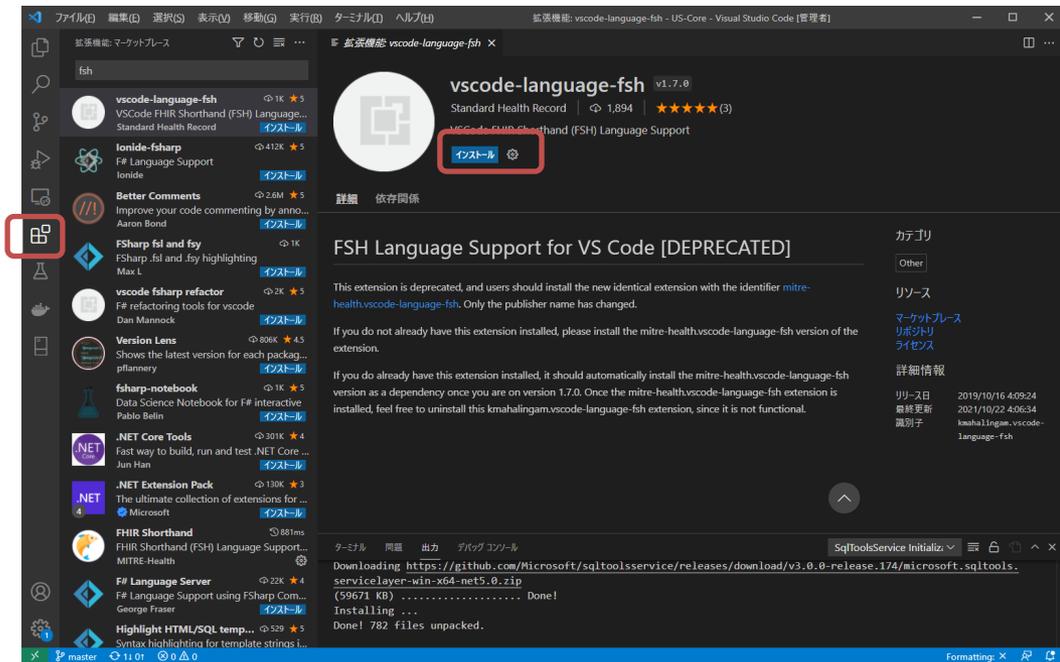
Run and Debug

Una Kravets
@una

VS @code does so many things right. I'm constantly impressed by the UX, and

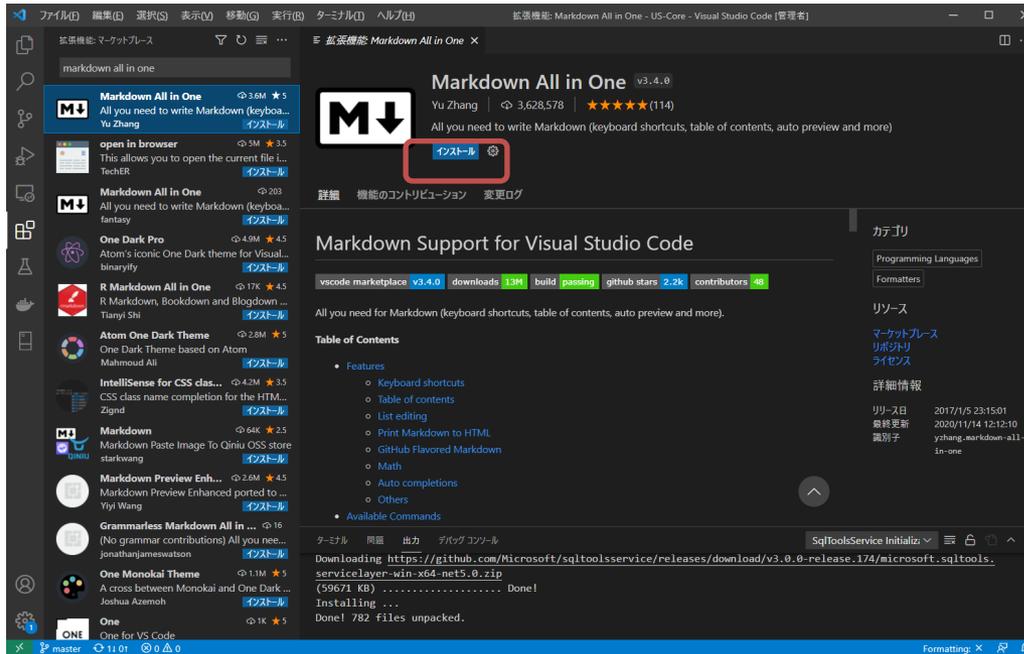
Jonathan Dunlap
@jdunlap

VS Code will have a remote development environment that can connect to a cloud

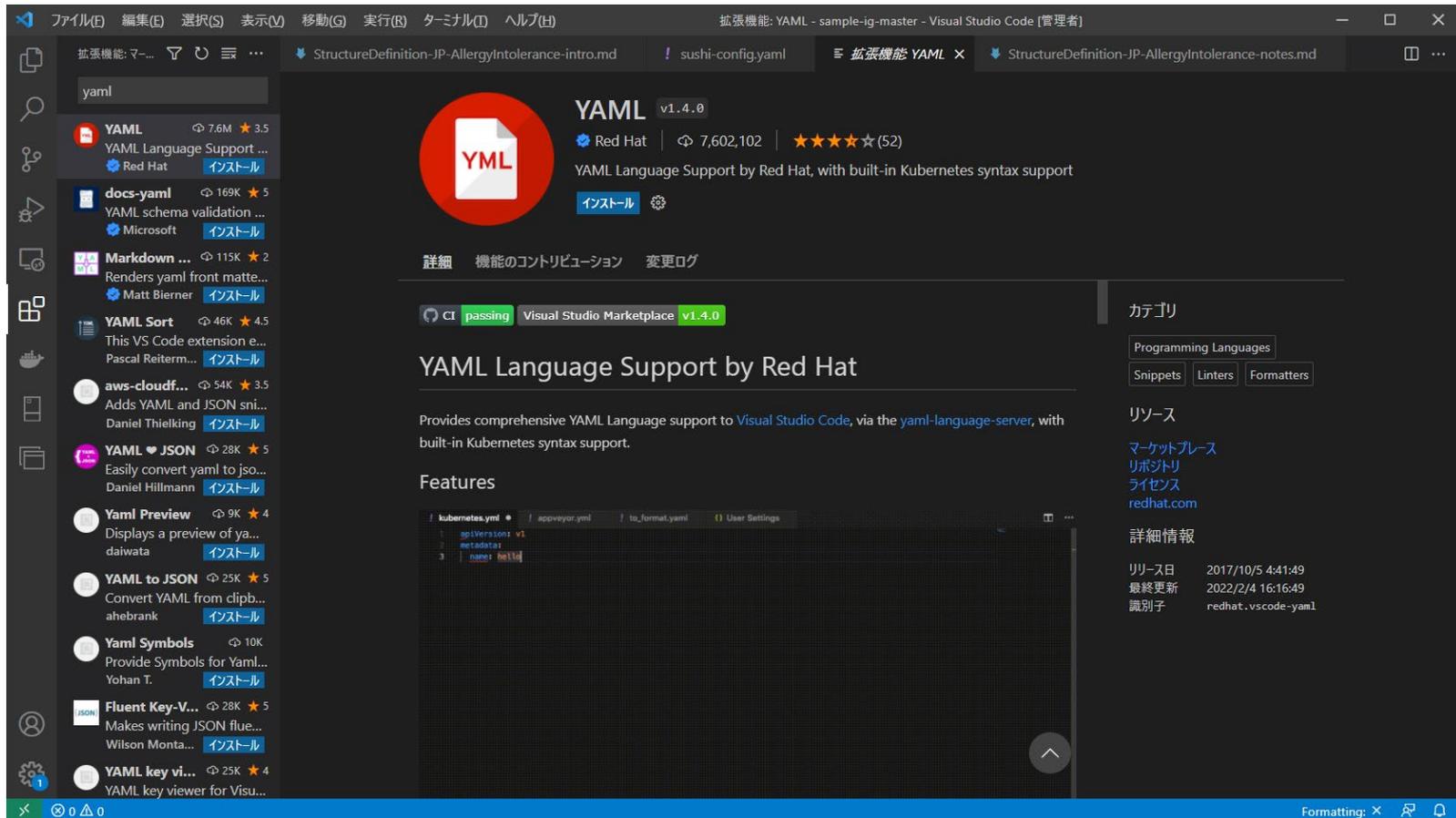


vscode-language-fsh : fshファイル編集用

画面左側のアイコンをクリックし、対象のExtensionを検索し、インストールを実施します。



前回同様にextensionの印スールを実施してください
 markdown all in one : marddownファイル編集用



前回同様にextensionの印スールを実施してください
 yaml :yamlファイル編集用

ビルド環境の確認

sushi -initの実行

```
> mkdir sushi-proj ↵
> cd sushi-proj ↵
> sushi -init ↵

-----
| This interactive tool will use your answers to create a |
| working SUSHI project configured with your project's   |
| basic information.                                     |
-----

Name (Default: ExampleIG): simple-sushi ↵
Id (Default: fhir.example): simple-sushi.example ↵
Canonical (Default: http://example.org): http://example.org ↵
Status (Default: draft): draft ↵
Version (Default: 0.1.0): 0.2.0 ↵
Publisher Name (Default: Example Publisher): hl7 Students ↵
Publisher Url (Default: http://example.org/example-publisher): http://example.org/hl7-Students ↵
Initialize SUSHI project in (対象フォルダ)¥sushi-proj¥simple-sushi? [y/n]: y ↵
```

- 対象フォルダを作成し、そのフォルダにカレントフォルダを移動後に、「sushi -init」を実行します。
- 初期設定項目に対する入力行くと、Nameで指定したフォルダ配下にSushi用の初期フォルダおよび初期ファイルが配置されます。

フォルダ構造の確認

```
> cd simple-sushi
> tree /F
対象フォルダ
├── .gitignore
├── ig.ini
├── sushi-config.yaml
├── _genonce.bat
├── _genonce.sh
├── _updatePublisher.bat
├── _updatePublisher.sh
├── input
│   ├── ignoreWarnings.txt
│   ├── fsh
│   │   └── patient.fsh
│   └── pagecontent
│       └── index.md
```

- 先程入力したnameのフォルダに移動し、内容を確認してください。
- 「input」-「fsh」フォルダがsushiのコンパイル対象ファイルになります。
- patient.fshファイルがサンプルファイルとして作成されています。

```
> sushi .
info Running SUSHI v2.2.6 (implements FHIR Shorthand specification v1.2.0)
(中略)
```

SUSHI RESULTS			
Profiles	Extensions	Logicals	Resources
1	0	0	0
ValueSets	CodeSystems	Instances	
0	0	1	
Exclamation point!		0 Errors	0 Warnings

```
>tree /F
(対象フォルダ)
. .gitignore
. .ig.ini
. sushi-config.yaml
. _genonce.bat
. _genonce.sh
. _updatePublisher.bat
. _updatePublisher.sh

fsh-generated
├── includes
│   └── menu.xml
└── resources
    ├── ImplementationGuide-simple-sushi.example.json
    ├── Patient-PatientExample.json
    └── StructureDefinition-MyPatient.json
```

- 同じフォルダにて「sushi .」を実行します。
- コマンド結果にsushiが実行され、profilesが1つ、instanceが1つ、変換されたこと、またエラーが出力されていないことを確認します。
- 実行後「fsh-generated」フォルダおよびその中にファイルが作成されていることを確認します。

```
> _updatePublisher.bat
Checking internet connection...
xxx.xxx.xxx.xxx からの応答: バイト数 =32 時間 =100ms TTL=121
We're online

IG Publisher is not yet in input-cache or parent folder.
Ok? (Y/N) y ↵
Will place publisher jar here: (ルート)sushi-proj¥simple-sushi¥input-
cache¥publisher.jar
Downloading most recent publisher to Input Cache - it's ~100 MB, so
this may take a bit

Updating scripts
Update scripts? (Y/N) y ↵
Updating _updatePublisher.sh
Updating _genonce.bat
Updating _gencontinuous.bat
Updating _genonce.sh
Updating _gencontinuous.sh
Updating _updatePublisher.bat

> dir input-cache /B
publisher.jar
```

- IGPublisherモジュールをダウンロードするため、「_updatePublisher.bat」を実行します。
- 途中質問の回答を求められますがすべてyで回答してください。
- 100MBのファイルになるため環境によっては時間がかかります。
- ダウンロード終了後「dir input-cache /B」を実行しファイルが生成されていることを確認します。

```
>_genonce.bat
(大量トレースログ表示)
>start .\output¥index.html
```



0.2.0 - ci-build

Home Artifacts

Table of Contents - Home

[simplsushi - Local Development build \(v0.2.0\). See the Directory of published versions ↗](#)

1 Home

Official URL: http://example.org/ImplementationGuide/simple-sushi.example	Version: 0.2.0
Draft as of 2022-02-24	Computable Name: simplsushi

simple-sushi

Feel free to modify this index page with your own awesome content!

IG © 2022+ hl7 Students ↗. Package simple-sushi.example#0.2.0 based on FHIR 4.0.1 ↗. Generated 2022-02-24
Links: Table of Contents | QA Report

0.2.0 - ci-build

Home Artifacts

Table of Contents

[simplsushi - Local Development build \(v0.2.0\). See the Directory of published versions ↗](#)

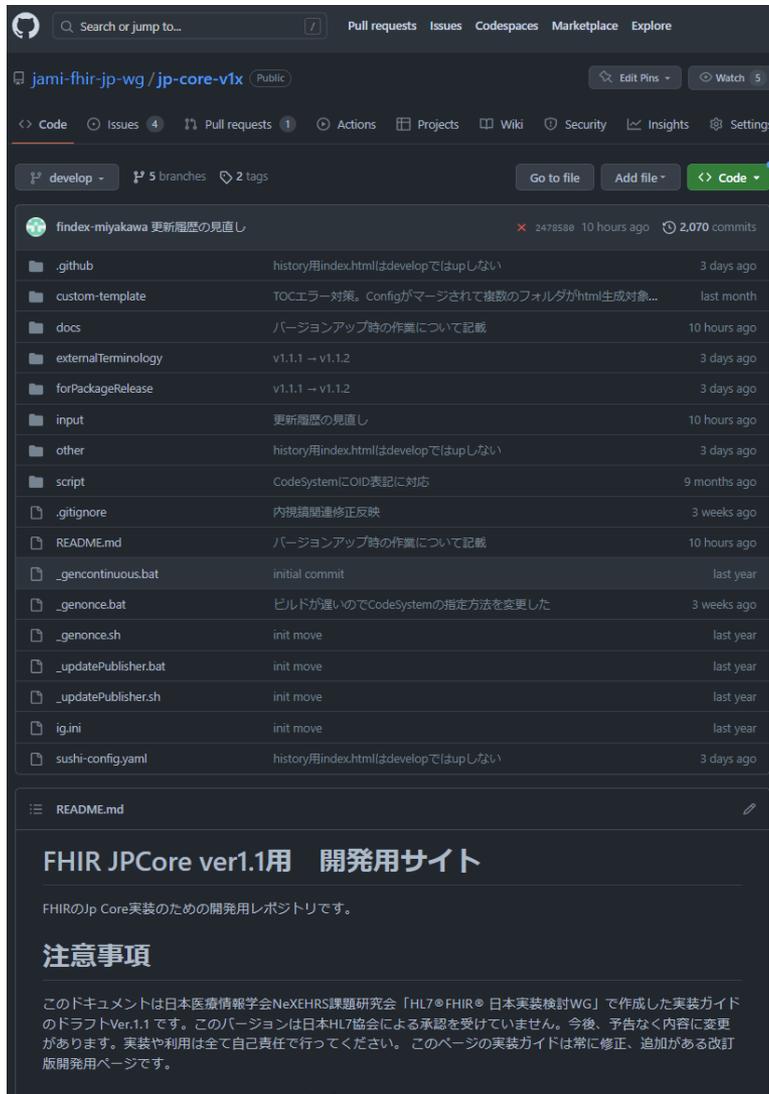
0 Table of Contents

- 0 Table of Contents
- 1 Home
- 2 Artifacts Summary
 - 2.1 MyPatient
 - 2.2 PatientExample

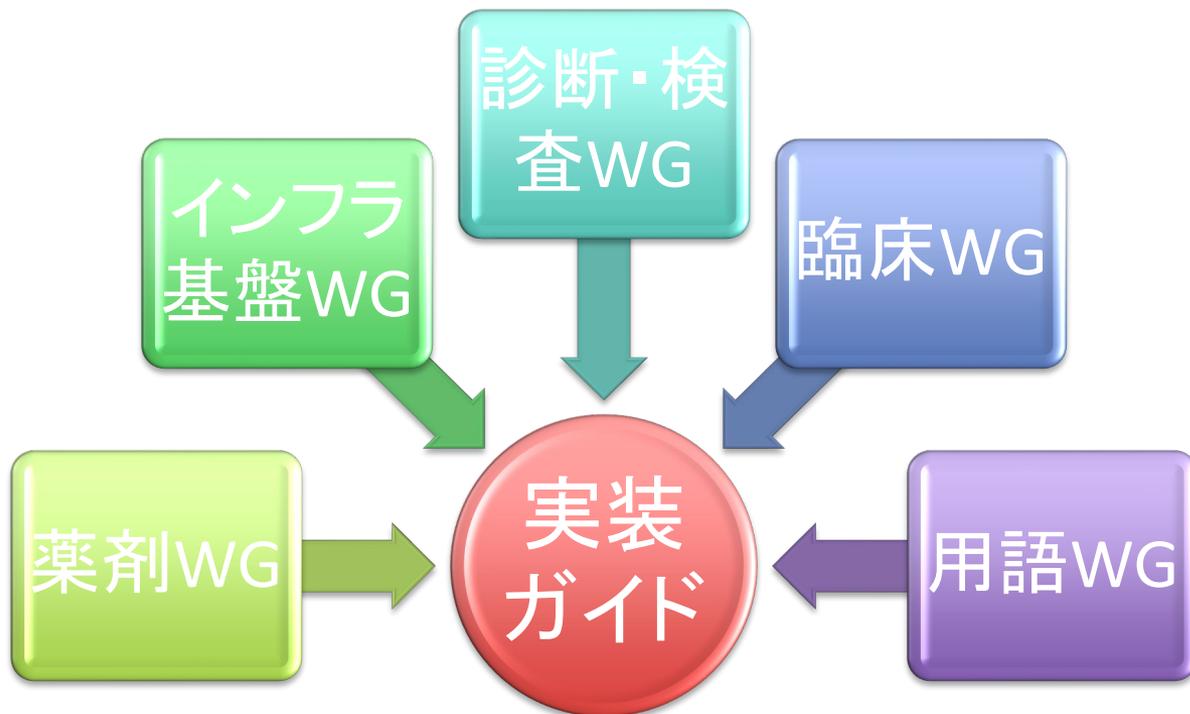
IG © 2022+ hl7 Students ↗. Package simple-sushi.example#0.2.0 based on FHIR 4.0.1 ↗. Generated 2022-02-24
Links: Table of Contents | QA Report

- IGPublisher実行するために「_genonce.bat」のコマンド発行します。内部的にsushiモジュールもコールされます。
- 実行すると画面にトレースログが表示されます。
- 処理終了した後に「.\output¥index.html」を開くと作成した実装ガイドのhtmlが表示されます。

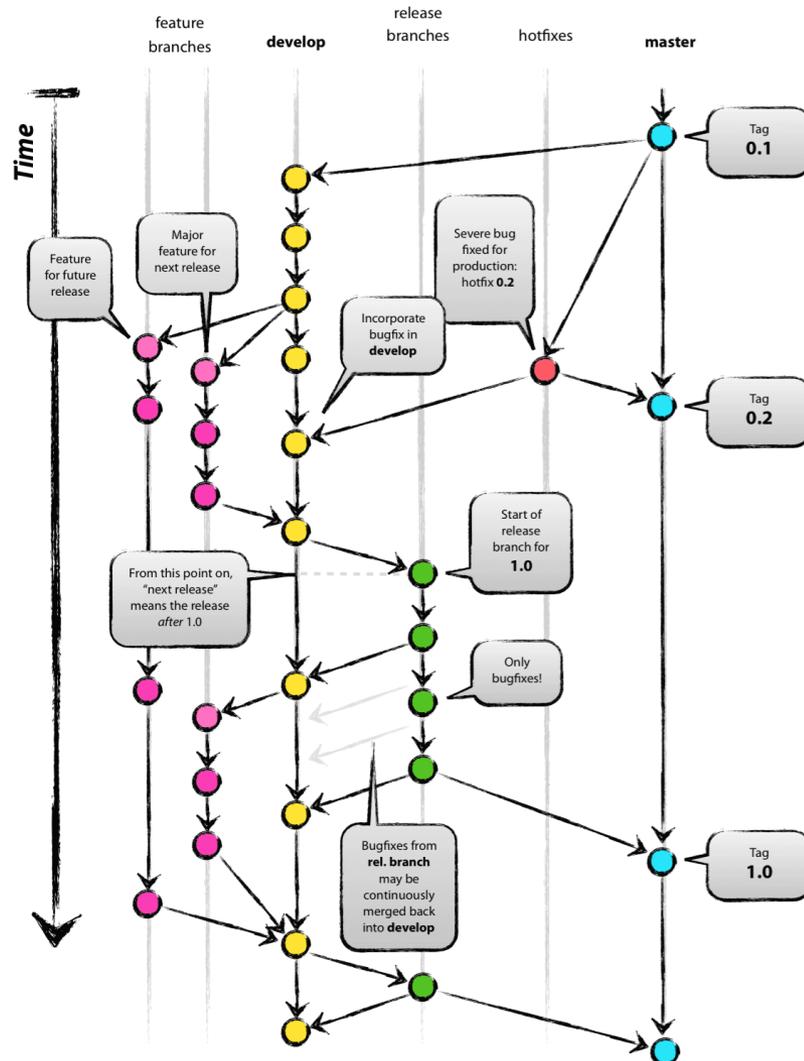
GIT共同作業環境 (参考)



- GitHub上構築
OpenSourceコミュニティにおけるデファクトスタンダード
- 実装ガイド作成時ルールや情報をReadme.mdに記載
- GitHubの機構を利用して複数人による共同編集環境を実現

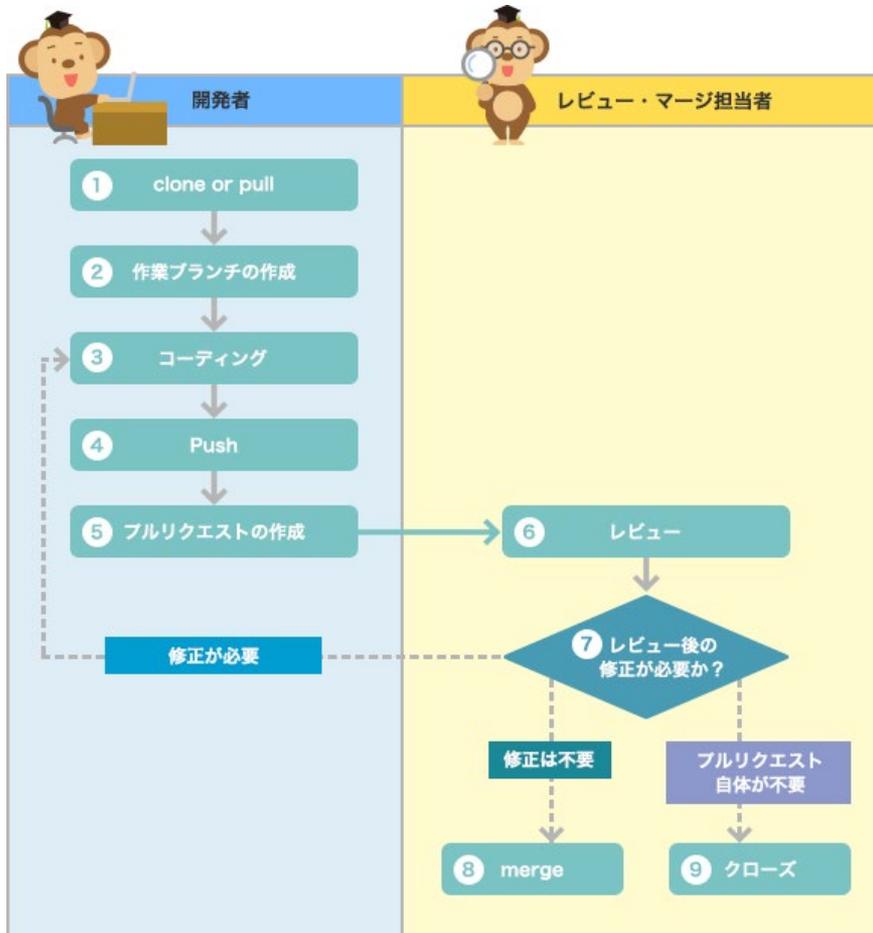


- 複数グループ同時並行で編集する必要あり
- タイミングを制御できないため、GitHubの分岐レポジトリの仕組みを利用

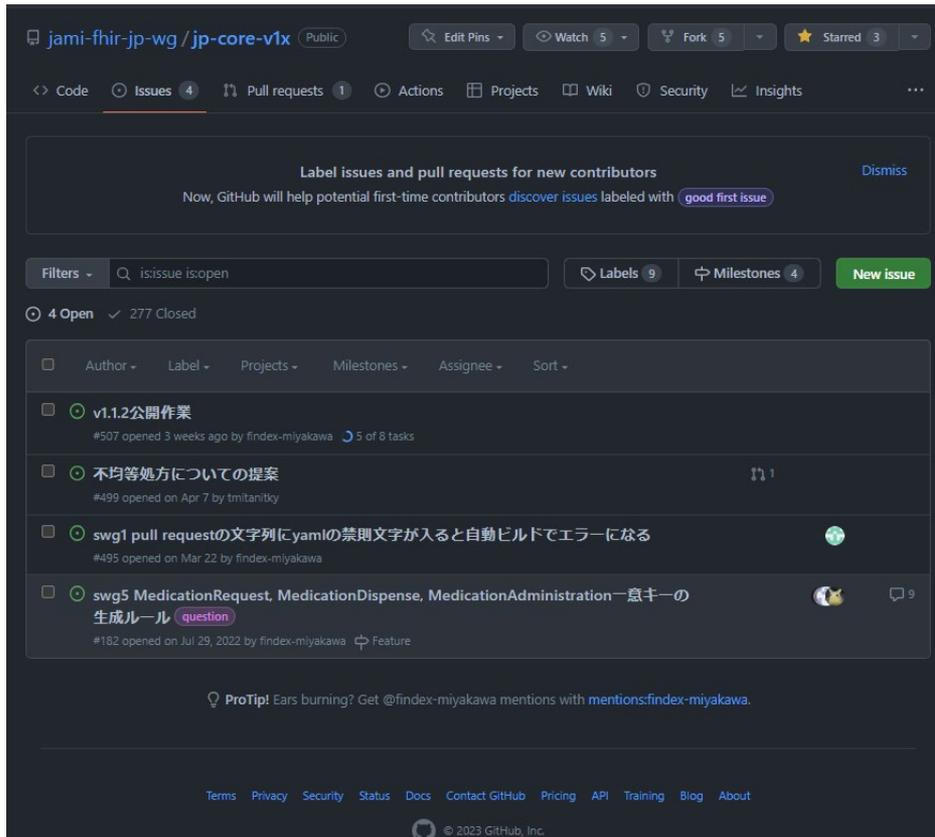


Gitにおけるリポジトリの分岐モデル。複数のFeatureブランチにて作成したものをdevelopへマージする際にPull Requestを作成しレビューをしてもらうもの。他にもいくつかのモデルがある。

- main(master)
 - ユーザにプロダクトとしてリリースするソースコードを管理するブランチ
 - タグでバージョンを管理
- develop
 - 実際に行う開発作業を行うブランチ
 - 新しい機能はfeatureブランチを作成して開発
- feature
 - developブランチを基盤に作成されるブランチ
 - 新しい機能を開発
 - 開発が終わったらdevelopブランチにマージ
- release
 - developブランチを基盤に作成されるブランチ
 - リリース直前にバグ修正などの微調整、つまりQAを行うブランチ
 - QAを完了したら、master, developブランチにマージ
- hotfix
 - リリースされたバージョンで発生したバグを速やかに修正するブランチ
 - 修正後すぐmaster, developブランチにマージ



- 開発者がFeatureブランチでの開発を終えて、developブランチにマージにする際に、レビューアに確認してもらう仕組み
- レビューアはPull Request画面よりGitHub上で差分ソース等を確認し、修正点があれば指摘をし、問題がなければDevelopブランチへのマージを実施する。



- 修正点（要望・不具合）を Git Issueにて管理
- git accountがあれば外部からの意見も受け入れ可能（設定による）
- ソースコミットやPull Requestのマージの際に Issue番号を記載すると自動的にCloseする



developへのPull
Request依頼

GitHub Actionの自
動ビルド起動

JP Core Implementation Guide

ビルド結果一覧

<https://jami-fhir-jp-wg.github.io/jp-core-v1xpages/index.html>

name version	branch	last published	user	IG Link waring
HL7-FHIR-JP-Core-R4 (1.1.2) develop	develop	23年06月17日(土) 14時32分 new	findex-miyakawa	📖 実装ガイド ⚠️ 警告 <input checked="" type="checkbox"/> QA e=90,w=216,h=58
HL7-FHIR-JP-Core-R4 (1.1.1) main	main	22年11月03日(木) 17時46分	findex-miyakawa	📖 実装ガイド ⚠️ 警告 <input checked="" type="checkbox"/> QA e=0,w=0,h=0
HL7-FHIR-JP-Core-R4 (1.1.0a) hotfix	hotfix/v110	22年11月03日(木) 16時13分	findex-miyakawa	📖 実装ガイド ⚠️ 警告 <input checked="" type="checkbox"/> QA e=0,w=0,h=0

Validation Results for FHIRJPCoreImplementationGuide
Generated Sun Jun 18 02:04:27 UTC 2023, FHIR version 4.0.1 for hl7.fhir.jp.core#1.1.2 (canonical = <http://jpfhir.jp/fhir/core> (history)). See [Full QA Report](#)

Quality Checks

Publisher Version: IG Publisher Version: v1.3.15, which is out of date. The current version is v1.3.16 [Download Latest](#)

Publication Code: n/a . PackageId = hl7.fhir.jp.core, Canonical = http://jpfhir.jp/fhir/core

Realm Check for n/a:

- n/a

Publication Request:

- This IG has never been published
- This IG has never been published, so the version should start with '0.' or include a patch version e.g. '-ballot'
- No publication request found

Suppressed Messages: 13 Suppressed Issues

Dependency Checks:

Package	Version	FHIR	Canonical	Web Base	Comment
hl7.fhir.jp.core	1.1.2	🔴	http://jpfhir.jp/fhir/core		
hl7.terminology.r4	5.0.0	🔴	http://terminology.hl7.org	http://terminology.hl7.org/5.0.0	Latest Release is 5.1.0
hl7.fhir.uv.extensions.r4	1.0.0	🟢	http://hl7.org/fhir/extensions	http://hl7.org/fhir/extensions/1.0.0	

Templates: hl7.fhir.jp.core.ig-template=null -> fhir.base.template=current

Dependent IGs: [no references](#)

Global Profiles: (none declared)

HTA Analysis: [All OK](#)

R5 Dependencies:

- <http://hl7.org/fhir/5.0/StructureDefinition/extension-NamingSystem-version>
- <http://hl7.org/fhir/5.0/StructureDefinition/extension-NamingSystem-uid>

Modifier Extensions: (none)

Previous Version Comparison: Unable to compare with previous version to find version history at <http://jpfhir.jp/fhir/core> (Problem #1 with package-list.json at <http://jpfhir.jp/fhir/core>: Invalid HTTP response 404 from <https://jpfhir.jp/fhir/core/package-list.json?nocache=1687048546995> (Not Found) (content in /tmp/http-log/fhir-2.log))

Comparison: n/a

Validation Suppressed: [ImplementationGuide/hl7.fhir.jp.core](#), [ImplementationGuide/hl7.fhir.jp.core](#)

Summary: errors = 90, warn = 216, info = 58, broken links = 0

Filename	Errors	Warnings	Hints
Build Errors	0	0	0
/home/runner/work/jp-core-v1x/jp-core-v1x/fsh-generated/resources/CapabilityStatement-jp-client-capabilitystatement	0	1	0
/home/runner/work/jp-core-v1x/jp-core-v1x/fsh-generated/resources/CapabilityStatement-jp-server-capabilitystatement	0	1	0
/home/runner/work/jp-core-v1x/jp-core-v1x/fsh-generated/resources/CodeSystem-jp-documentcodes-cs	0	1	0

警告用ページ

HL7 FHIR JP Core ImplementationGuide
1.1.2 - ci-build JP

FHIRJP | Guidances | FHIRContents | Security | Artifacts | Download

Table of Contents > HL7 FHIR JP Core 実装ガイド

HL7 FHIR JP Core ImplementationGuide - Local Development build (v1.1.2). See the [Directory of published versions](#)

HL7 FHIR JP Core 実装ガイド

項目	内容
定義URL	http://jpfhir.jp/fhir/core/ImplementationGuide/hl7.fhir.jp.core
Version	1.1.2
Name	FHIRJPCoreImplementationGuide
Title	HL7 FHIR JP Core ImplementationGuide
Status	Active (2023-06-26)
Copyright	Copyright FHIR Japanese Implementation research working group in Japan Association of Medical Informatics (JAMI) 一般社団法人日本医療情報学会NeXEHRS課題研究会FHIR日本実装検討WG

このドキュメントは日本医療情報学会NeXEHRS課題研究会 (FHIR) 日本実装検討WGで作成した実装ガイドのドラフトVer.1.1.2である。このバージョンは日本HL7協会による承認を受けていない。今後、予告なく公開される可能性がある。本ドキュメントは全て自己責任で行なうこと。

1.1 概要

ガイダンス: JP Coreでの全体に関する規則や注意事項を記述する

- 総合ガイダンス
- Must SupportとCardinality(多重度)のルール
- 欠損値の扱い
- 文字コード
- 検索
- 利用上の注意事項
- 更新履歴

開発途中
実装ガイド

まとめ

- 実装ガイドを作成するための環境について説明を行った
- ビルド環境をするにインストールと、編集をする際の便利なツールを説明した
- 編集人数が少ない場合には、自分の端末でビルドするので問題は無し。複数人で編集をするのであればGitHub等のバージョン管理を利用するとよい
- GitHubにGitHubPagesの機能があり、ビルドからHomeページ構築までを一気通貫でできる
- JP Coreのビルド手順等は公開されているので実際にビルドを試してみるのがよい



ご静聴ありがとうございました