

Article

[Lorenzo Scalese](#) · Oct 10, 2022 3m de lecture

Création et validation de n'importe quelle ressource HL7 FHIR en utilisant le schéma FHIR à l'aide d'IntelliSense et de la fonctionnalité de complétion automatique dans VS Code

L'interopérabilité des soins de santé permet d'améliorer les soins aux patients, de réduire les coûts des prestataires de soins et de fournir une image plus précise aux prestataires. Cependant, avec un si grand nombre de systèmes différents, les données sont formatées de nombreuses manières différentes. De nombreuses normes ont été créées pour tenter de résoudre ce problème, notamment HL7v2, HL7v3 et CDA, mais toutes présentent des inconvénients.

FHIR (Fast Healthcare Interoperability Resources), ou Ressources rapides d'interopérabilité des soins de santé, est un nouveau format pour les échanges des informations médicales qui vise à résoudre ces problèmes. Il est développé par Health Level Seven International (HL7), une organisation qui a également développé HL7v2, HL7v3 et CDA.

Aujourd'hui nous allons explorer comment créer et valider une ressource FHIR en utilisant le schéma FHIR à l'aide d'IntelliSense et de la fonctionnalité de complétion automatique dans VS Code.

Etape 1 : Téléchargement du fichier de schéma JSON pour la validation des ressources sur le site officiel de FHIR <https://www.hl7.org/fhir/>.

hl7.org/fhir/documentation.html

Study on Surah Al-... Mobile WiFi Point Of Sale (POS)... Become our Partner GoMart.pk - Online... powerbuilder PowerBuilder | App... Learn web develop... Other bo

HL7 FHIR Release 4

Home Getting Started Documentation Resources Profiles Extensions Operations Terminologies

Table of Contents > Documentation Index

This page is part of the FHIR Specification (v4.0.1: R4 - Mixed Normative and STU). This is the current published version. For a full list of available versions, see the [Directory of published versions](#)

1.1 Documentation Index

FHIR Infrastructure Work Group	Maturity Level: N/A	Standards Status: Informative
--	---------------------	-------------------------------

This page provides an index to the key commonly used documentation pages for FHIR.

- Framework**
 - Conformance Rules [N](#)
 - Resource Life Cycles
 - References between Resources [N](#)
 - Compartments
 - Narrative [N](#)
 - Extensibility [N](#)
 - Formats: [N](#) XML [N](#), [N](#) JSON [N](#), & [N](#) RDF
 - Terminologies [N](#) (Code Systems, Value Sets)
 - FHIRPath [N](#)
 - Mappings to other standards
- Version Management**
 - Change Management & Versioning [N](#)
 - Managing Multiple FHIR Versions
 - Version History
 - Differences to Release 3
- Exchanging Resources**
 - RESTful API (HTTP) [N](#)
 - Search [N](#) (Search Param Registry)
 - Operations [N](#)
 - Asynchronous Use
 - Using GraphQL
 - Documents
 - Messaging
 - Services
 - Persistence/Data bases
- Base Types**
 - Data Types (Base) [N](#)
 - Metadata Types [N](#)
 - Resource [N](#)
 - DomainResource [N](#)
 - Element [N](#)
- Adopting & Using FHIR**
 - Profiling FHIR [N](#)
 - FHIR Workflow
 - Downloads - Schemas, Code, Tools
 - Managing Multiple FHIR Versions
 - Validating Resources
 - Best Practices for Implementers
 - Mapping Language (tutorial)
 - Testing Implementations
- Safety & Security**
 - Security, Security Labels & Signatures
 - Clinical Safety
- Implementation Advice**
 - Managing Resource Identity
 - Guide to Resources

Étape 2: Création d'un dossier (dans cet exemple, j'utilise le dossier Patient et la ressource Patient) et copiage du fichier fhir.schema.json extrait dans le même dossier, puis ouverture du dossier à partir du code VS.

Étape 3: Configurez le code VS pour reconnaître le schéma FHIR en modifiant le fichier setting.json. Appuyez sur CTRL+SHIFT+P et tapez les paramètres de l'espace de travail JSON

Étape 4: Création d'un nouveau fichier patient.fhir.json dans le même dossier. Appuyez sur Ctrl+Espace et vous obtiendrez tous les attributs des ressources FHIR à travers IntelliSense

Ajoutez le type de ressource Patient et tous les attributs liés à la ressource Patient vont apparaître dans l'IntelliSense.

VS Code validera automatiquement la structure et la syntaxe de la ressource.

Avec l'aide d'IntelliSense et de la fonction de complétion automatique, nous avons créé et validé notre ressource patient.

Step 5: Affichez la ressource créée dans le serveur FHIR d'InterSystems en utilisant l'API Rest à partir de postman

Récupérer la ressource patient créée en utilisant la méthode "Get"

Félicitations, nous avons créé, validé notre ressource patient et réussi à l'envoyer et la récupérer sur le serveur FHIR d'InterSystems en utilisant postman.

De cette façon, nous pouvons facilement créer et valider n'importe quelle ressource FHIR.

[#FHIR](#) [#REST API](#) [#Caché](#) [#Ensemble](#) [#InterSystems IRIS for Health](#) [#VSCode](#)

URL de la source:<https://fr.community.intersystems.com/post/cr%C3%A9ation-et-validation-de-nimporte-quelle-ressource-hl7-fhir-en-utilisant-le-sch%C3%A9ma-fhir-%C3%A0>