Article
Lucas Enard · Jun 3, 2022   5m read

Questionnaire & Forms in FHIR : From creation to usage

This article will discuss FHIR Questionnaire and QuestionnaireResponse from the creation of the form to the upload on the server and how to fill them.

tl;dr :

- By using this online tool you can easily build your own form from scratch or using a template from an already existing one.
- By using this InterSystems local FHIR server you can easily store your FHIR resources and Questionnaire.
- By using this app you can manipulate, as if you were a Practitioner, Questionnaire and Response for each and every Patient on your FHIR server.

It is to be noted that the app doesn't communicate using the Content-Type 'application/json+fhir' but just the Content-Type 'application/json' so it won't work like this with our local InterSystems FHIR server.

That's why I created this GitHub repo holding a modified version of the app, working with Content-Type 'application/json+fhir', holding a local FHIR server and a link to the Questionnaire builder tool with some explaination.

After cloning the repo, by doing docker-compose up -d, npm ci, npm run build and then npm run start you will be greated with an access to the app, by selecting the FHIR server you want and the Patient you want to work with, you will be able to fill Questionnaire and save them to you server in 2 clicks.

End of tl;dr
What's next is the ReadMe of the GitHub.

1. App for FHIR forms using a local fhir server

This is an app mainly based on this repo that can be used to display

FHIR

SDC

Questionnaire

and collect data as FHIR QuestionnaireResponse resources.

By building it using docker-compose up -d you will have access to a [local FHIR server](#) that can then be used to test the app.

2. Requirements

The app relies on the [LHC-Forms](#) rendering widget for displaying forms. It has partial support for FHIR Questionnaires (versions STU3 and R4) and the [Structured Data Capture Implementation Guide](#).

This widget will be installed with the dependencies.

For some sample forms to try, this repository comes with some forms under e2e-test/data/R4 that are automatically loaded into the local FHIR server at build.

2.1. Add Node.js and npm to your path

The file bashrc.lforms-fhir-app specifies the version of Node.js we are using for development. Download that version of Node.js, and add its bin directory to your path.

2.2. Install Dependencies

By running this command you will be able to install everything needed for the app to work.

```
npm ci
```

3. Local FHIR server

If you don't have a FHIR server to try this app out, you can start and use a local FHIR server powered by InterSystems technologies by doing inside the fhir-form folder :

```
docker-compose up -d
```

After some wait, your local FHIR server is up and you can access it by using http://localhost:32783/fhir/r4

Note that this link is already registered in the app.

4. Using the app

To use the app you have to [build](#) it then [start](#) it.

You can now access any FHIR server of your choice using the menu of the app but if you want you can use this [local FHIR server](#)

4.1. Build the application

```
npm run build
```

This will create files for production in a "dist" directory, but will also copy some needed files into place from node_modules.

4.2. Run the Application

```
npm run start
```

will start an http server running at port 8000.

Now browse to the app at localhost:8000/forms-fhir-app/.

Here you can choose a server to connect to.

If you want to use the local FHIR server, start the local FHIR server then, on the app select the first choice http://localhost:32783/fhir/r4

5. FHIR form / questionnaire

5.1. Creating your own FHIR form

By using this online tool you can easily build your own form from scratch or using an already existing one.

We advise you to import one of the existing one in the e2e-tests/data/R4 folder and start from here to understand how the tool works.

5.2. Importing your FHIR form

Using the app, you can easily import your local forms and use them right away using the upload button.

If you are using the formbuilder tool, you can, if you have a FHIR server supporting the Content-Type 'application/json', export the form you are creating directly to the fhir server using the export button.

If your server doesn't support the Content-Type 'application/json' but only the Content-Type 'application/json+fhir' for example, as our local FHIR server you must export the form to a file, then on the app, upload the file to the server as the app communicate in Content-Type 'application/json+fhir'.

#FHIR #InterSystems IRIS for Health

Source URL:https://community.intersystems.com/post/questionnaire-forms-fhir-creation-usage