Instructions to Setup and Run

Da Vinci Bulk FHIR Server

Below are the instructions to setup and run the SUFU application

Prerequisites:

1. Java 8
2. Apache Tomcat 7 or 8
3. PostgresSql Database 10.x
4. Maven 3.3.x
5. GIT

# Clone the Repository

Clone the respository using the below command in command prompt

git clone <https://github.com/HL7-DaVinci/atr.git>

# Installation Instructions

## Postgres Configuration:

Load Schema and data into database

Create the database by running the below command in command prompt

$ createdb -h localhost -p 5432 -U postgres <database\_name>

Database file bulk-r4.sql will be in the cloned repository. Load Schema and data using the below command.

$ psql -U postgres -d <database\_name> -f <Path to cloned repository>/bulk-r4.sql

## Create Build:

### Build Bulk FHIR Server Backend Service:

Change the database configurations in the file application.properties located under src/main/resources

jdbc.url=jdbc:postgresql://localhost:5432/<database\_name>

jdbc.username=<username>

jdbc.password=<password>

Then navigate to SUFU backend service directory and run Maven build to build application war file.

$ mvn clean install

This will generate a war file under target/DavinciFHIR.war. Copy this to your tomcat webapp directory for deployment.

##### Start Tomcat Service

If the tomcat is started successfully then you should be able to access below endpoints

GET END POINTS:

1.http://localhost:<tomcatport>/<Service\_name>/fhir/Group/1/$export?\_type=Patient,Practitioner,Organization,Location,Coverage,RelatedPerson,PractitionerRole --Extracts all the Patients from the given Group.

2.http://localhost:<tomcatport>/<Service\_name>/fhir/Group/2/$export?\_type=Patient,Practitioner,Organization,Location,Coverage,RelatedPerson,PractitionerRole --Extracts all the Patients from the given Group.

Request Headers:

1. Prefer: respond-async

2. Content-Type: application/fhir+json

The above requests will provide the Content-Location in Response Headers

Response Headers:

1. Content-Location:http://localhost:<tomcatport>/<Service\_name>/bulkdata/<JobId>

### Polling to Content-Location

Method: GET

URL: [http://localhost:<tomcatport>/<Service\_name>/bulkdata/<JobId](http://localhost:%3ctomcatport%3e/%3cService_name%3e/bulkdata/%3cJobId)>

Response Body:

{

    "transactionStartTime": "Thu, 16 Jan 2020 07:16:42 UTC",

    "request": "http://localhost:<tomcatport>/<service\_name>/fhir/Group/1/$export",

    "secure": "false",

    "output": [

        {

            "type": "Patient",

            "url": "http://localhost:<tomcatport>/<service\_name>//bulkdata/download/41/Patient.ndjson"

        }

………

    ]

}

### Download the NDJson files:

Method: GET

URL: http://localhost:<tomcatport>/<service\_name>/bulkdata/download/41/Patient.ndjson

Response Body: <JSON Object with list of Patients>

### Update data in database:

Please follow the below instructions to create and update the records in database.

1. In Windows open the pg\_admin tool and select the database.
2. Expand <database\_name> 🡪 schemas 🡪 public 🡪 Tables. A list of tables will be displayed.
3. Right click on a table and select the View data option to view the data.
4. The Actual resource JSON data will be under the column ‘data’.
5. Double click on the row column ‘data’ and update the JSON object.
6. Click on Save button on top of the table to save the changes.