

Manual - Team 22

Team Name: Team 22

Team Members: Mehul Kamlesh Bhuvu, Santha Kumar Bobbili, Joshua Metcalf, Poonyaporn Nanthakijjar

Project Name: Dynamic FHIR Servers

Github Link: <https://github.com/inferno-community/dynamic-fhir-servers>

The fhir-api provides a dynamic server for handling get/search requests based on user provided json data in the form of the implementation guide that contains a capability statement

REST API:

GetById

Ability to search for specific resources by id and get data back based on example json data provided for that resource

Route:

GET /resource/id

Response:

resource JSON

Search

Ability to query with various valid parameter combinations and get data back based on example json data provided for that resource

Route:

GET /resource?queryParameters

Response:

resource JSON

Requires:

fhir-validator-wrapper running in the background found at:
<https://github.com/inferno-community/fhir-validator-wrapper>

Create

Route:

POST /resource BODY: Must contain "id", "resource_type", "resource"

Reponse:

created resource JSON

Delete

Route:

DELETE /resource/id

Reponse:

successfully deleted resource with specified id

Update

Route:

PUT /resource/id BODY: Must contain valid JSON to be stored in resource column

Reponse:

successfully updated resource with specified id

System dependencies

Requires PostGreSQL installed 1.2.3 or above

Requires Ruby version 2.7.2

Requires Rails 6.0.3.4

Configuration

Assumes default PostgreSQL password, can be modified in the Database.yml file as needed

Database creation

```
rake db:create
rake db:structure:load
# To load json example data from <Ruby Root>/db/seeds/package/example/
rake db:seed json_file='package'
# To load json data from Synthea from <Ruby Root>/db/seeds/synthea
rake db:seed json_file='synthea'
```

Running Locally

Initial Setup Commands

```
bundle install
```

Starting up server each time

```
rails s
```

Note: <https://github.com/inferno-community/fhir-validator-wrapper> must be running on port 4567 if using search commands