

# HTML Form Entry REST documentation

The two main objectives of the project were to create wrapper rest controllers over the existing HFE module and to create a simple interface to test out the new rest module

Installation:

The installation is very simple just like any other OpenMRS module. The simplest way to load it is via module management in the administration section of the UI. The HFE module is a prerequisite for the same.

Brief Documentation of the APIs that were built:

API for Html Form management: (creating, viewing, editing html forms)

Url : `openmrs/ws/rest/v1/htmlformentryrest/htmlform?id=<html_form_id>`

GET:

Expected Response:

```
{
  "id": 1,
  "previewHtml": "<htmlform formEncounterType=\"67a71486-...\"
}
```

POST:

Body:

```
"form":{
  "name":"test",
  "description":"testing form via rest",
  "version":"1.0",
  "encounterType":"7",
  "creator":{
    "personName":"piyush"
  },
  "changedBy":{
    "personName":"piyush"
  },
  "published":"checked"
},
"xmlData":"<htmlform ..</htmlform>"
}
```

Note: the xmlData should have escaped " or any other characters which might invalidate the json structure. The response will be a simple success or failure message.

List of Html Forms available in the system

Url: <http://localhost:8080/openmrs/ws/rest/v1/htmlformentryrest/htmlformslist?username=admin&password=Admin123>

GET:

Result shall contain a list of html form names with their ids.

Eg.

[

```
{
  "Name" = "form1",
  "id" =1
}, ...
]
```

HtmlFormEntry:

localhost:8080/openmrs/ws/rest/v1/htmlformentryrest/htmlformentry?encounterId=5624&mode=EDIT&username=admin&password=Admin123

POST:

The body can be of the form as shown below which is just a concatenation of the form fields w1, w2.... etc and some other fields like personId, htmlFormId, formModifiedTimestamp, encounterModifiedTimestamp, encounterId (content type should be set to multipart/form-data)

eg.

personId=37&htmlFormId=1&formModifiedTimestamp=1528086638000&encounterModifiedTimestamp=1533134156000&encounterId=5624&closeAfterSubmission=&hasChangedInd=true&w1=2018-08-01&w3=1&w5=5&w8=100&w10=56&focus-in-bmi-question=&w12=37&w14=70&w16=50&w18=100&w20=80&w22=99

OR

All the same fields can be given in a json format like so: (content type should be set to Application/json as a header)

Body:

```
{
  "personId": 123,
  "w1" : "abc", ...
}
```

GET:

Shall return a body similar to below.

```
{
  "patientPersonName": "John Taylor",
  "GuessingInd": "false",
  "encounterDatetime": 1512225216000,
  "encounterId": 396,
  "patient.personId": 37,
  "encounterTypeName": "Visit Note",
  "htmlFormId": 2,
  "encounterModifiedTimeStam": 1533570637000,
  "hasChangedId": "false",
  "htmlToDisplay": "<htmlform ...</htmlform>",
  "encounterFormName": "Visit Note",
  "formName": "Visit Note",
  "formModifiedTimeStam": 1533563876000,
  "fieldAccessorJavascript": "propertyAccessorInfo.....",
  "encounterLocationName": "Unknown Location",
  "context.mode": "VIEW"
```

}

Deleting an Encounter related to a htmlform:

URL: localhost:8080/openmrs/ws/rest/v1/htmlformentryrest/encounter?encounterId=123

DELETE - a simple DELETE http call at the above url will delete the encounter relating to the htmlformentry.