Unsupported REST Module

This module will not be the preferred REST web service module.
For a more complete list of web service methods see this module's replacement: Webservices.rest Module

Overview

What This Module Does Do

The OpenMRS REST Module provides a limited mechanism for communicating with an OpenMRS system externally as a client via Web Services. In this case, the REST Module is implemented using REST (1).

The methods or functions currently exposed via the OpenMRS REST Module are as follows:

- patient - get 1 or more patients by patient identifier
- findPatient - get 1 or more patients by patient name
- Observations - get a list of observation by submitting the patient identifier and concept id
- Submitting an hl7 message - send the hl7 source name and hl7 message string via the HTTP PUT method
- User - get a user by user identifier or name
- findUser - get 1 or more patients by user roles, identifier and/or name

What This Module Does Not Do

Here's some caveats to usage of the REST Module:

- REST is not SOAP - If you need SOAP, you'll need to write a SOAP layer on top of this module, or convert your SOAP calls to REST in your client
- The entire OpenMRS API is not exposed here -- hopefully more of the API is exposed as time goes on
- Calls to the API are not Audited (as far as I can tell)

Access URLs

- /moduleServlet/restmodule/api/findPatient/
- /moduleServlet/restmodule/json/findPatient/
- /moduleServlet/restmodule/api/patient
- /moduleServlet/restmodule/json/patient
- /moduleServlet/restmodule/api/hl7
- /moduleServlet/restmodule/api/obs?pId=patientIdentifier&cId=conceptId
- /moduleServlet/restmodule/json/obs?pId=patientIdentifier&cId=conceptId
- /moduleServlet/restmodule/api/user/identifierOrName
- /moduleServlet/restmodule/json/user/identifierOrName
- /moduleServlet/restmodule/api/findUser/identifierOrName?role=RoleName&role=OtherRoleName
- /moduleServlet/restmodule/json/findUser/identifierOrName?role=RoleName&role=OtherRoleName

Example Usage

Example #1: Search for patients matching a search criteria

This call would do a Patient Query based on the search string "John Smith". A list of patients matching that search criteria would be returned.
<patientList>
  <patient birthdate="1975-01-01" birthdateEstimated="false" gender="M">
    <identifierList>
      <identifier preferred="1" type="OpenMRS Identification Number">100-8</identifier>
      <identifier type="Old Identification Number">asdf</identifier>
      <identifier type="Old Identification Number">100</identifier>
    </identifierList>
    <name>
      <prefix>pre</prefix>
      <givenName>test123</givenName>
      <middleName>middle</middleName>
      <familyName>test</familyName>
      <familyName2>test2</familyName2>
      <degree>deg</degree>
    </name>
    <addressList>
      <address preferred="1">
        <address1>555 Johnson Rd.</address1>
        <address2>Apt. 555</address2>
        <cityVillage>Indianapolis</cityVillage>
        <stateProvince>IN</stateProvince>
        <country>USA</country>
      </address>
      <address preferred="1">
        <address1>asdf</address1>
        <address2></address2>
        <cityVillage></cityVillage>
        <stateProvince></stateProvince>
        <country></country>
      </address>
    </addressList>
  </patient>
</patientList>
Example #2:

This call does a Patient Lookup based on the OpenMRS Identifier "OpenMRS Identification Number". In this case, the Patient with the ID Number of "100-3" would be returned.
<patientList>
  <patient birthdate="1975-01-01" birthdateEstimated="false" gender="M">
    <identifierList>
      <identifier preferred="1" type="OpenMRS Identification Number">100-8</identifier>
      <identifier type="Old Identification Number">asdf</identifier>
      <identifier type="Old Identification Number">100</identifier>
    </identifierList>
    <name>
      <prefix>pre</prefix>
      <givenName>test123</givenName>
      <middleName>middle</middleName>
      <familyName>test</familyName>
      <familyName2>test2</familyName2>
      <degree>deg</degree>
    </name>
    <addressList>
      <address preferred="1">
        <address1>555 Johnson Rd.</address1>
        <address2>Apt. 555</address2>
        <cityVillage>Indianapolis</cityVillage>
        <stateProvince>IN</stateProvince>
        <country>USA</country>
      </address>
      <address preferred="1">
        <address1>asdf</address1>
        <address2/>
        <cityVillage/>
        <stateProvince/>
        <country/>
      </address>
    </addressList>
  </patient>
</patientList>
Example #3: Request for a list of observations for a given patient and concept
<?xml version="1.0" encoding="UTF-8"?>
<obsList version="1.0">
  <obs uuid="1b6898bf-bc11-4b79-9f30-8a3c0c4dd10a" patientId="3" obsDatetime="2010-02-15" isObsGrouping="false" isComplex="false">
    <concept id="5090">HEIGHT (CM)</concept>
    <valueNumeric>183.0</valueNumeric>
    <encounter id="1" type="ADULTINITIAL"/>
    <comment>Too tall</comment>
    <location id="1"/>
  </obs>
  <obs uuid="1b6808bf-bc17-4b98-9f30-8a3c0c4dd10t" patientId="3" obsDatetime="2010-02-15" isObsGrouping="false" isComplex="false">
    <concept id="5090">HEIGHT (CM)</concept>
    <valueNumeric>200.0</valueNumeric>
    <encounter id="1" type="ADULTINITIAL"/>
    <comment>Too obese</comment>
    <location id="1"/>
  </obs>
</obsList>
Note: The parameters at the end of the request URL i.e pid and cid are patientId and conceptId respectively. If you wish to view the JSON output in your browser, you will need to install a json viewer plugin for your respective browser if it exists, otherwise you will be prompted to save the file.

Example #4: Sending an hl7 message using rest

http://myhost:serverport/openmrs/moduleServlet/restmodule/api/hl7?message=my_hl7_message_string&source=myHl7SourceName

Example #5: Getting a patient based on the user identifier

/api/user/1

```xml
<?xml version="1.0" encoding="UTF-8"?>
<userList version="1.0">
  <user uuid="c6ef8bd8-fd7e-11df-8272-0800271bba91" userId="1" systemId="admin" username="" birthdateEstimated="false" gender="">
    <addressList/>
    <name>
      <givenName>Super</givenName>
      <familyName>User</familyName>
    </name>
    <roleList>
      <role>System Developer</role>
      <role>Provider</role>
    </roleList>
  </user>
</userList>
```

/json/user/4

```json
[
  {
    "addressList": [],
    "userId": "4",
    "systemId": "4-2",
    "username": "ajanthan",
    "gender": "M",
    "uuid": "c6ef8bd8-fd7e-11df-8272-0800271bba91",
    "name": {
      "givenName": "aja",
      "familyName": "bala"
    },
    "roleList": {
      "System Developer",
      "Provider"
    }
  }
]
```

Resources

- View REST module source
- Download REST module

Tidbits

- Calls to the REST library are restricted by the IP source address of the calling client. By default, this IP list is limited to calls from localhost (127.0.0.1). If you want to modify this list to allow other clients to call the REST layer, simply update the appropriate property in the setting (formerly global property 1.8 and below) table in the database.
- Passing connection credentials: ...

Release Notes

1.4.4.2
• Added findUser to find users by role, name and/or id
• Cleaned up rendering process

1.4.4.1
• Added ability to fetch users by id - TRUNK-351
• Clarified some of the hl7 message fetch errors

1.4.4
• URLs no longer need to have a trailing backslash
• Fixes to other minor bugs

1.4.3
• Changed found identifiers for patients to only include active (non-voided) identifiers

1.4.2
• Added ability to receive and process hl7 messages in a restful way

1.4.1
• Fixed error during authentication when the username or password isn't provided

1.4
• Added support for Obs(Observations) fetching.
• Version numbers have been added to the xml output for both patients and obs.

1.3.3
• Fixed json for patients without any addresses

1.3.1/1.3.2
• Fixed json for multiple patients
• Removed trailing commas from json

1.3
• Added json functionality for patient searching

1.2.3
• Fixed module for v1.5 of OpenMRS