Informatica Cloud Application Integration

AllScripts Practice Management Connector Package

October 2018

Contents

Copyright:

Overview:

Allscripts Healthcare Solutions provides physician practices, hospitals, and other healthcare providers with practice management and electronic health record technology.

This AllScripts Service Connector allows Informatica Cloud Application Integration (CAI) developers to incorporate access to AllScripts Electronic Health record system in their real-time integration scenarios.

This AllScripts Connector and associated assets demonstrate how to authenticate against AllScripts and retrieve patient details based on search parameters. Additional functionality is available for the reader to explore on their own.

Prerequisites:

- An AllScripts account with API access and the following login credentials:
  - User Name
  - User Password
  - Application Name
  - Application User ID
  - AllScripts Practice Management URL

- Appropriate role/permissions on Informatica Cloud Application Integration to perform an IMPORT operation

Use Case: Retrieve patient information based on first and last names

With credentials embedded in the Connection, supply

a. Inputs:
    i. First Name
    ii. Last Name

b. Outputs
    i. A list of patients and related information whose first and last names either equal or start with the provided first and last name inputs.

Quick Setup:

1. Download the AllScripts Connector package from the Informatica CAI GitHub Repository.

2. Import the “AllScripts PM Service Connector Package 1018.zip” package into your CAI workspace.

3. This creates a project named “AllScripts Practice Management”, with 3 assets in it.
    a. "AllScripts PracticeManagement 1402" Service Connector
    b. “AllScripts-PracticeManagement-1402-actions” Connection
    c. “Search Patients” Process

4. Go to the “AllScripts Practice Management” project

5. Publish the "AllScripts PracticeManagement 1402" Service Connector

6. Open the “AllScripts-PracticeManagement-1402-actions” Connection
    a. Under Properties, enter the required information. Note that once you enter the data, the first four fields will be displayed as bullets. ••••• That is because those fields have been defined as being encrypted in the service connector.
        i. Username
        ii. Password
        iii. Appname
        iv. AppUserID
        v. BaseURL - The baseURL is the AllScripts Practice Management URL. In the case of accessing a Unity PM sandbox, it would like “http://<your hostname>/UnityPM/unityservice.svc/json

    b. Save the Connection

    c. Publish the Connection

7. Publish the “Search Patients” Process

8. Test the “Search Patients” process in your REST tool of choice (Postman, RESTed, etc.) We’ll use Postman here.
    a. Create a new request.
    b. Define a POST operation.
    c. The Request URL is the Service Address from the just-published “Search Patients” process
        i. Actions-> Properties Detail…-> Service URL-> Copy
    d. The request body is of type “application/json”
    e. Specify your input search variables:

```
{
```

```
   "First Name": "<First name or first letters of first name>",
   "Last Name": "<Last name or first letters of last name>",
}
```

f. In our example, we'll use

```
{
   "First Name": "John",
   "Last Name": "M",
}
```

g. Invoke the operation ("Send", in the case of Postman)

h. Assuming there are records in your system that match the search criteria, you should receive a response that looks like this. Note that your search may yield more than one patient, depending upon your search criteria and the data in your system.

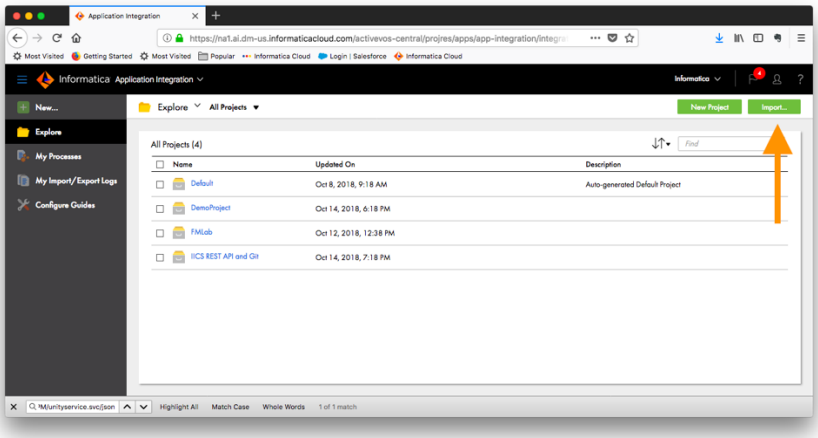
i.

```
{
    "List of Patients": [
        {
            "dateofbirth": "01/01/1980",
            "firstname": "john",
            "WorkPhone": null,
            "gender": "M",
            "ZipCode": "06030",
            "MRN": "311",
            "middlename": null,
            "City": "nq. gvfashua",
            "lastname": "mathew",
            "ssn": "543189876",
            "AddressLine1": "suite 600",
            "AddressLine2": null,
            "HomePhone": null,
            "State": "NH",
            "PhoneNumber": null,
            "age": "38y",
            "OtherNumber": null,
            "ID": "326",
            "CellPhone": null
        }
    ],
    "Count": 1
}
```

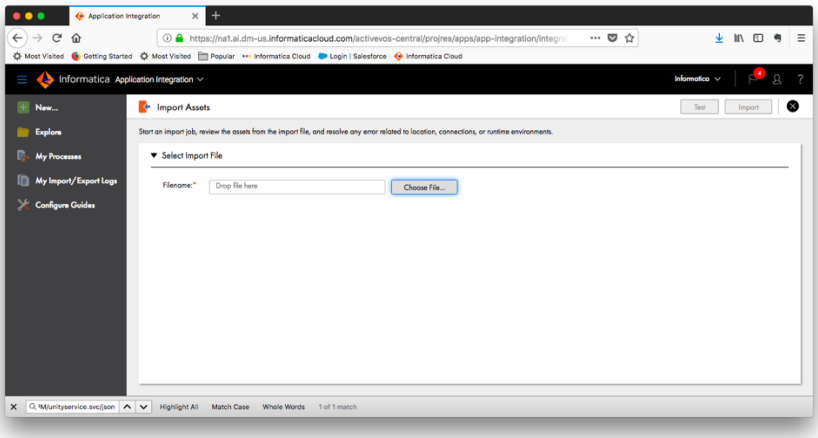
Detailed Setup:

1. Download the AllScripts Connector package from the Informatica CAI GitHub Repository.

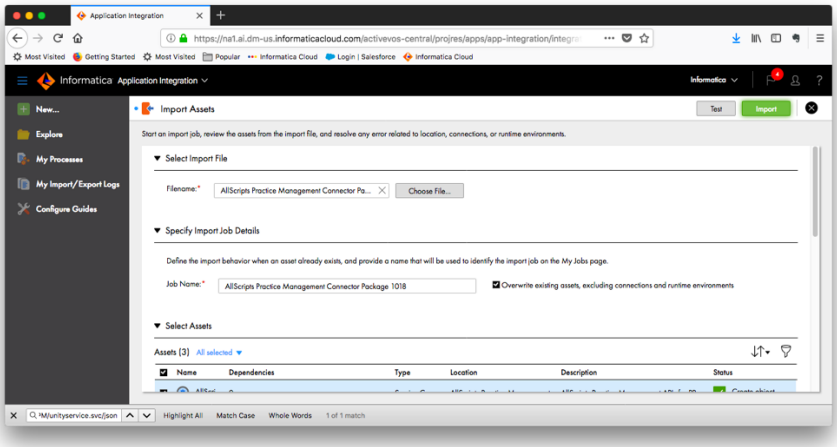
2. Import the “AllScripts PM Service Connector Package 1018.zip” package into your CAI workspace.
    a. Start the import process.



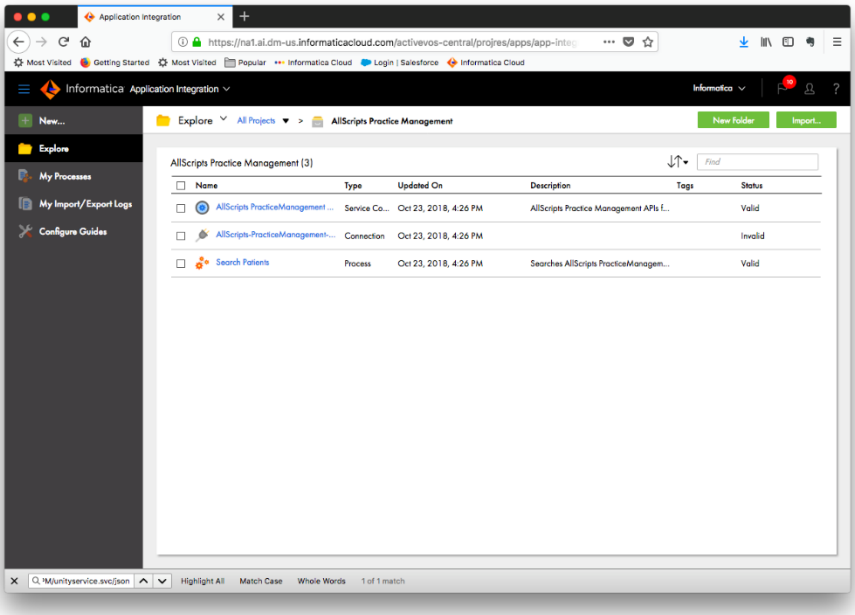
    b. Select the file that was just downloaded.



c. Complete the import process.

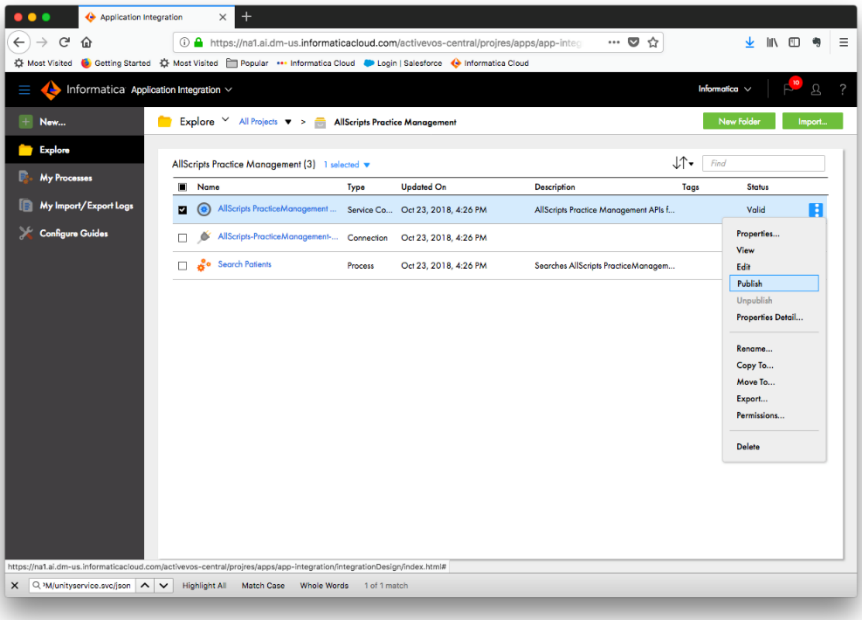


3. This creates a new project named “AllScripts Practice Management”, with 3 assets in it.
   a. "AllScripts PracticeManagement 1402" Service Connector
   b. “AllScripts-PracticeManagement-1402-actions” Connection
   c. “Search Patients” Process

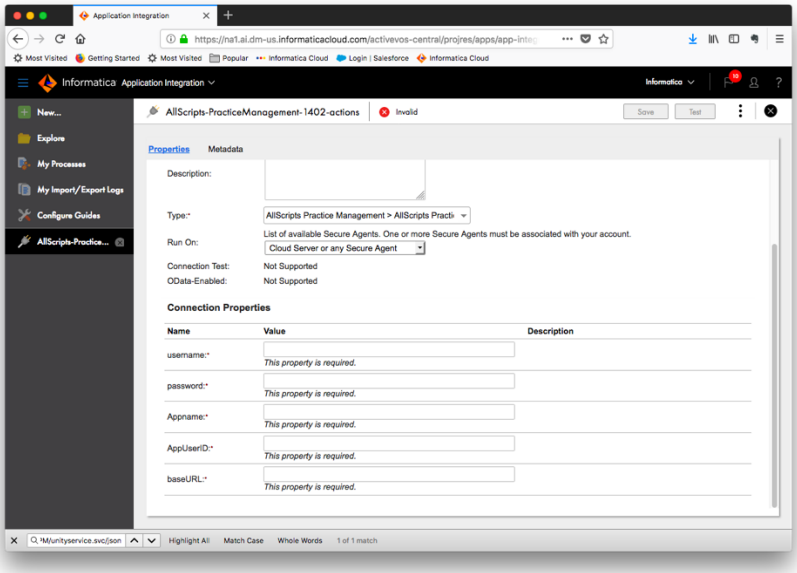


4. Go to the “AllScripts Practice Management” project.

5. Publish the "AllScripts PracticeManagement 1402" Service Connector



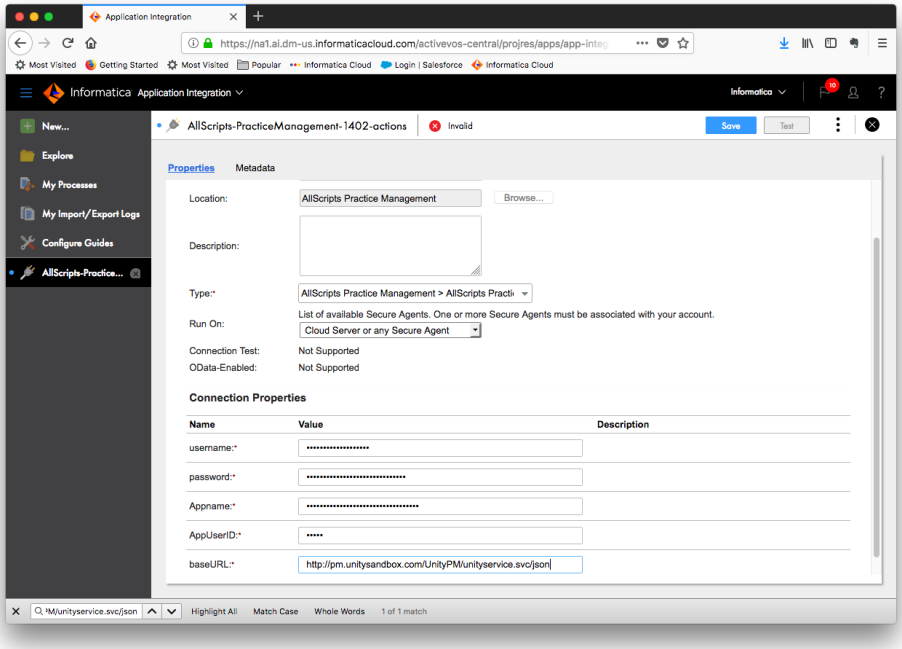
6. Open the “AllScripts-PracticeManagement-1402-actions” Connection



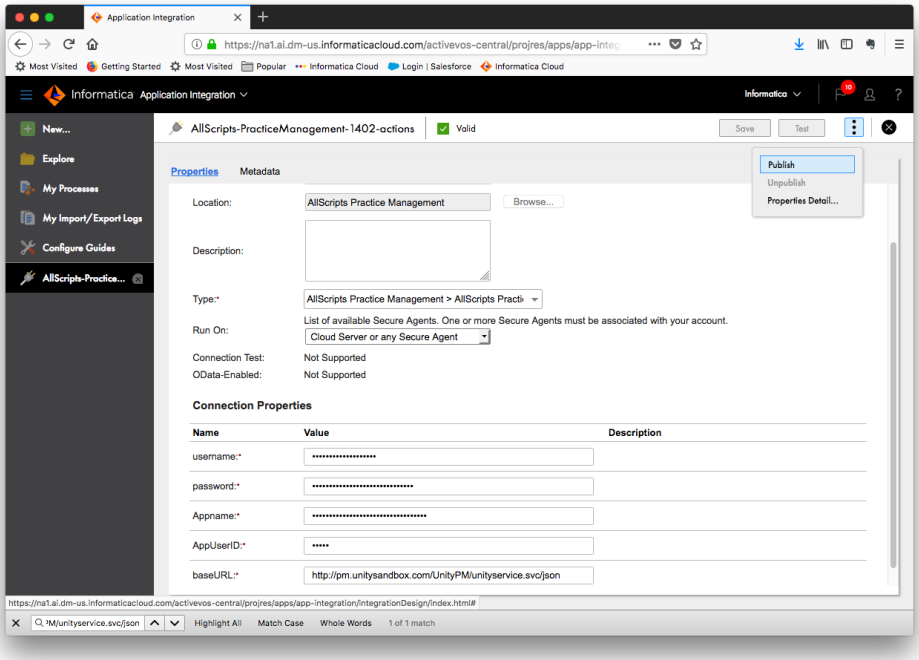
   a. Under Properties, enter the required information. Note that once you enter the data, the values in the first four fields will be displayed as bullets - •••••. That is because those fields have been defined in the Service Connector as being encrypted.
      i. Username
      ii. Password
      iii. Appname
      iv. AppUserID

v. BaseURL - The baseURL is the AllScripts Practice Management URL. In the case of accessing a Unity PM sandbox, it would be like “http://<your hostname>/UnityPM/unityservice.svc/json

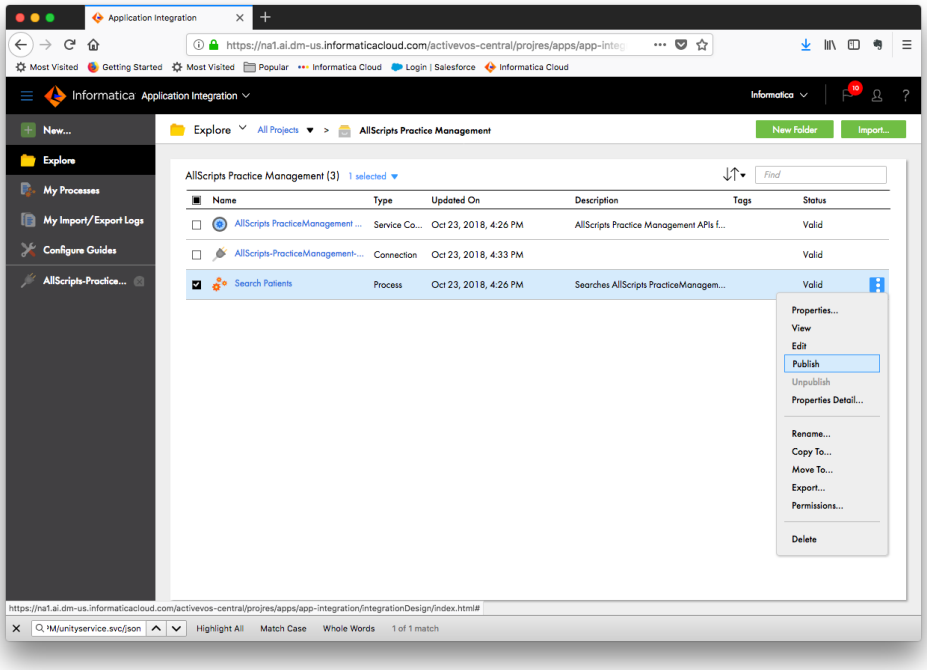
b. Save the Connection



c. Publish the Connection

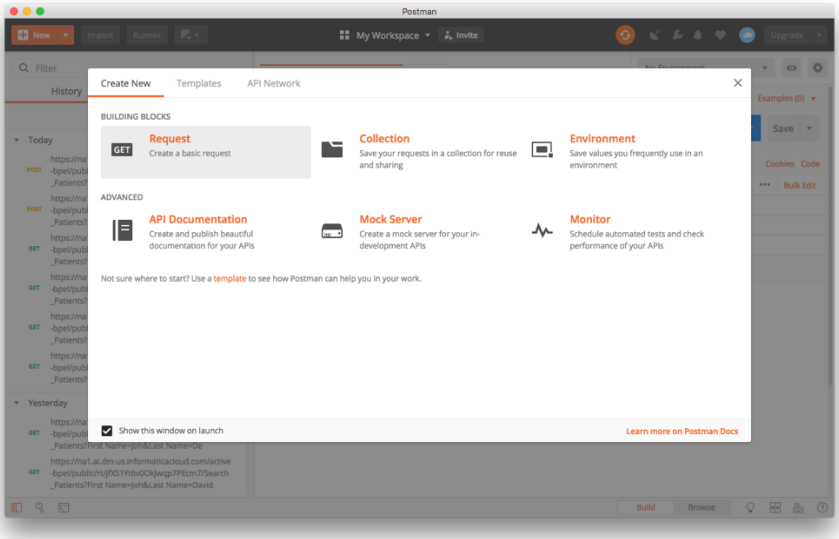


7. Publish the “Search Patients” Process



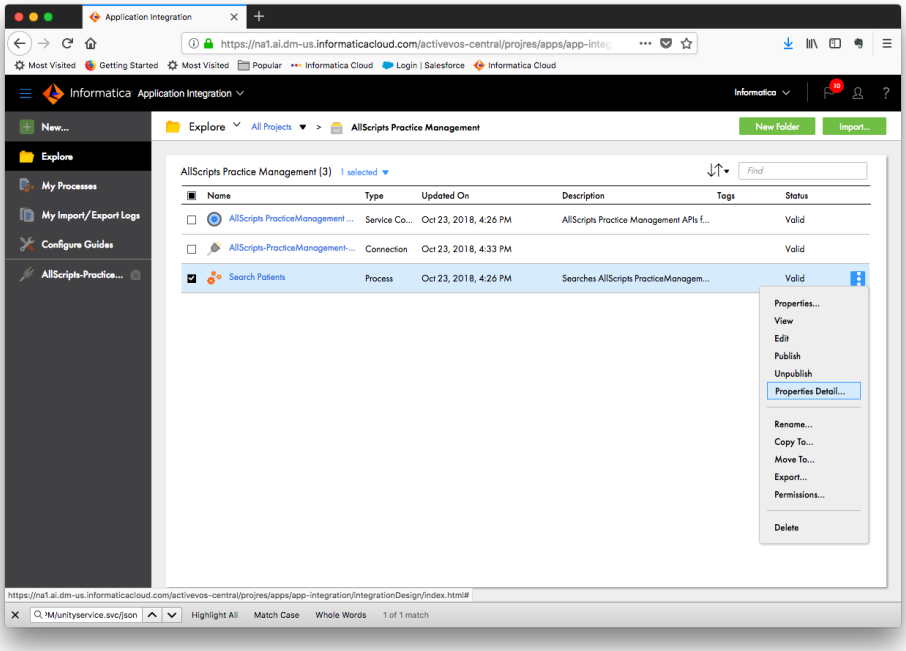
8. Test the “Search Patients” process in your REST tool of choice (Postman, RESTed, etc.) We’ll use Postman here.

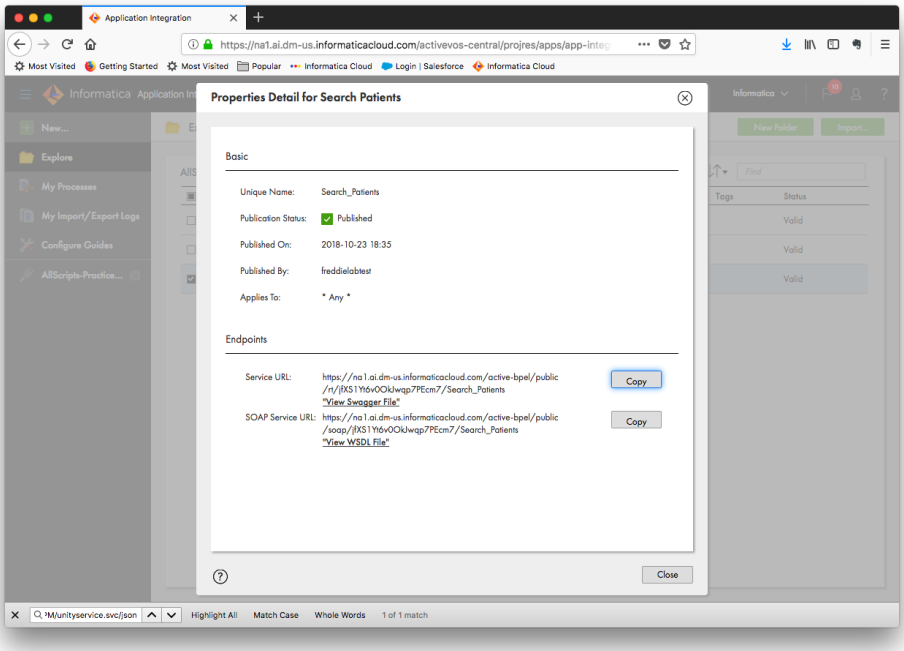
    a. Create a new request



    b. Define it as a POST operation
    c. The Request URL is the Service Address from the just-published “Search Patients” process

i. Actions-> Properties Detail…-> Service URL-> Copy

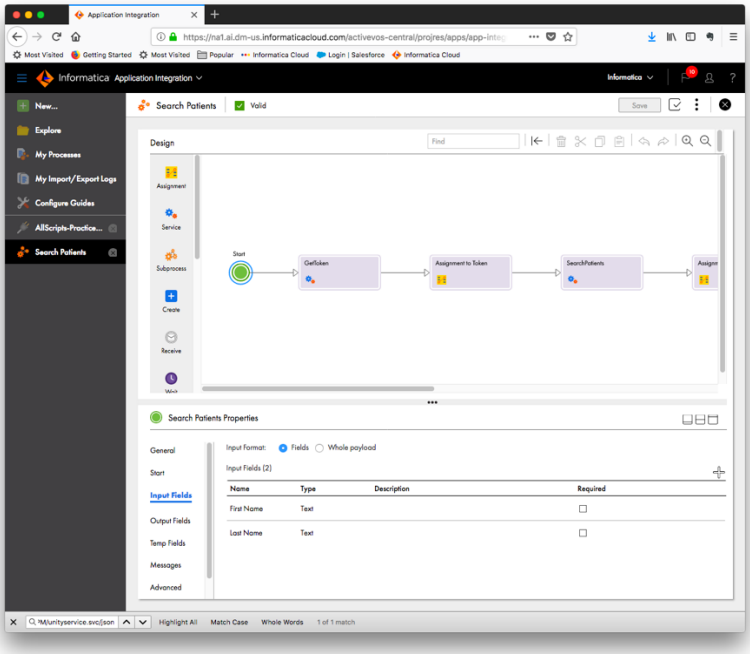


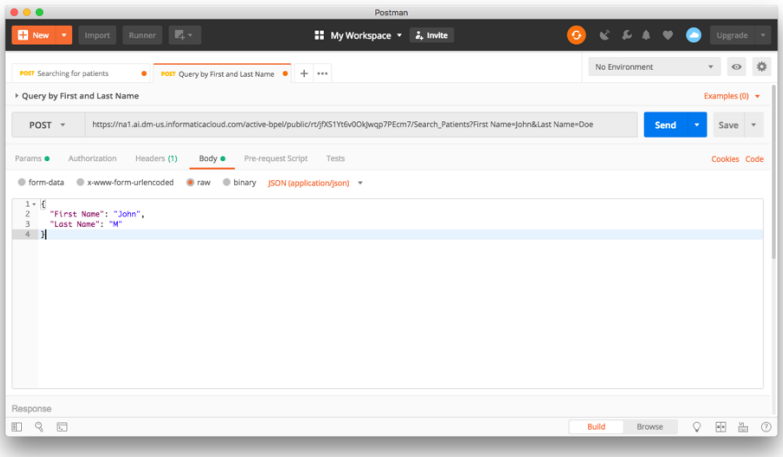


d. The request body is of type “application/json”
e. Specify your input search variables:

```
{
   “First Name”: “<First name or first letters of first name>”,
   “Last Name”: “<Last name or first letters of last name>”,
}
```

Note that you can determine what inputs are required for any process by selecting the **Start** button in the “Search Patients” process.





f. Invoke the operation (“Send”, in the case of Postman)
g. Assuming there are records in your system that match the search criteria, you should receive a response that looks like this. Note that your search may yield more than one patient, depending upon your search criteria and the data in your system.

h.


```
{
    "List of Patients": [
        {
            "dateofbirth": "01/01/1980",
            "firstname": "john",
            "WorkPhone": null,
            "gender": "M",
            "ZipCode": "06030",
            "MRN": "311",
            "middlename": null,
            "City": "nashua",
            "lastname": "mathew",
            "ssn": "543189876",
            "AddressLine1": "suite 600",
            "AddressLine2": null,
            "HomePhone": null,
            "State": "NH",
            "PhoneNumber": null,
            "age": "38y",
            "OtherNumber": null,
            "ID": "326",
            "CellPhone": null
        }
    ],
    "Count": 1
}
```