

2184 Morris Avenue, Union, NJ 07083
http://www.DigiDMS.com
support@DigiDMS.com
Phone: 908.688.8810, Fax: 877.733.4305

DigiDMS API Access Documentation

Document Type: **User Guide**

Version: 1.0

Preface

Document Owner: DigiDMS, Inc.

Target Audience

This guide is intended for Developer users.

Assumptions

Not Available

Contact Information

DigiDMS, Inc.
2184 Morris Ave,
Union, NJ- 07083
Ph: (908)-688-8810
Fax: (877)-733-4305

Copyright and Disclaimer

Confidentiality

This document and the information contained herein are confidential to and the property of DigiDMS, Inc. and are made available to DigiDMS employees and others acting on behalf of the company for the sole purpose of conducting development or business for DigiDMS, Inc.

The contents of this documents (either in part or whole) must not be reproduced, stored in a retrieval system, transmitted or duplicated in any form or by any means, including photocopying, electronic, mechanical, recording or otherwise, without the prior written permission of the copyright holder: DigiDMS, Inc. DigiDMS may make changes or improvements in the software or specifications described in this document at any time without notice. These changes will be incorporated in new release of this document.

Disclaimer

Every effort has been made to ensure the accuracy and correctness of the information contained herein. However, in the unlikely event of certain errors and/or omission, DigiDMS, Inc. will not be liable to be held responsible for any loss whatsoever due to the usage of this document. DigiDMS, Inc. also reserves the right to modify the contents of this document without any prior notice.

Document History

Version	Date	Change Type	Change Remarks
1.0	09/10/2018	Initial Document	

Related Documents

Contents

Introduction 6
Developer Terms of Use 6
Configuration 6
Guidance for developers 6
Requesting API Access 6
Authorization for patient access 6
Accessing API 7
Obtain secret key 7
Authentication - Generate access token and Refresh token 7
Fetch Patient Data 7
API - Endpoints 7
Application Access 8
User Authentication – Obtaining a access token for registered users only 8
Request URL 8
Access Token 8
Refresh Token 10
Patient Search/Selection 11
Fetch Patient Data 13
Patient's All Data(C-CDA) 13
Patient's Common Clinical Data Set 15
Qualified searches by date 15
Appendix A 16

Introduction

DigiDMS API documentation is meant to provide guidance to patients or third party developers for developing software applications to access Protected Health Information (PHI) from DigiDMS. DigiDMS API is partially based on the Argonaut Profiles based upon the core FHIR DSTU2 API and its documents.
This document provides a comprehensive step by step approach for users to understand authentication process and API methods to communicate and query patient's health data.

Developer Terms of Use

DigiDMS API access is governed by the Terms of Use that developers need to agree.

Configuration

DigiDMS API uses RESTful connection over HTTPS (secure). Any software application or library capable of a secure HTTP connection can communicate with DigiDMS API. Currently data search and read operations are only supported as per the MU 3 requirements.

Guidance for developers

- API attempts to conform to the design principles of Representational State Transfer (REST). API response is in XML data format.
- API is HTTPS based: GET Methods are used to retrieve data from the API.
- API methods will return an error if you do not make your request with the correct parameters and endpoints.
- Meaningful error messages will be provided.
- API access is authenticated via OAUTH2.0 Protocols.

Requesting API Access

There are two types of users who can use the API's outlined in this document to access patient data.

1. External users, vendor applications, patients who are not registered portal users. These users are required to go through a registration process.
2. The other types of users are patients who are registered portal users. These users do not have to go through a further registration process and can use their portal credentials for authentication.

Once a user has been registered, he/she must then go through an authentication process. A successfully authenticated user is given an access token (user token).

Authorization for patient access

Most of the DigiDMS API calls are patient specific. Only patient search/selection is not patient specific. The following authorization rules apply:

If the user is a patient they have access:

- To their own data.
- To patient's data for whom they are authorized.
- No access to the data for any other patients.

If the user is a provider or other practice user they have access:

- To patients data they are authorized to see in their EHR.
- No access to patients they are not authorized to see in their EHR.

Accessing API

1. Obtain Secret Key.
2. Authentication - Generate access token/Refresh token (when token expires)
3. Fetch Patient Data.

Obtain secret key

Already registered users (patients) can navigate to **My account** link on top left of their patient portal account and check for their secret key for accessing the API. New accounts created for a patient will already have a secret key generated and displayed in their My Account screen. Old patient accounts who do not have a secret key generated can click on the **Generate** link on right side of the Secret Key for API bar in **My account** screen.

Non-registered users need to register themselves with DigiDMS, Inc. and obtain credentials to access the API.

Based on the requested scope of access and after authorization from clinic or patient user will be provided a username and secret key that can be subsequently used to generate the access token for accessing patient data.

Authentication - Generate access token and Refresh token

DigiDMS API uses OAuth 2 for authentication and authorization. For Authentication, user needs to pass Grant_type, Username and Secret_key provided by DigiDMS. Authorization server returns Access Token and Refresh Token.

Access Token: This is a short-lived token and is used to fetch patient data.

Refresh Token: If the access token expires, this can be used to get a new access token. This is a fairly long-lived token. If this token has also expired, user needs to get the tokens again using the above parameters.

Fetch Patient Data

Using the access token obtained as explained above user can fetch patient data. Patient data will be returned in XML CDA Version 2.1 format. User can either fetch all data or specify different categories of data to be fetched based on the selected API endpoints.

API - Endpoints

API Name	API Endpoint URL
Access Token	POST https://mypersonalchart.com/FHIRDemo/oauth/token
Patient Search	GET https://mypersonalchart.com/FHIRDemo/api/PatientSearch?name=Jane
Retrieve Patient	GET https://mypersonalchart.com/FHIRDemo/api/PatientSearch?Pno=61
Problem/Condition	GET https://mypersonalchart.com/FHIRDemo/api/Condition?Pno=61
Health Concerns	GET https://mypersonalchart.com/FHIRDemo/api/HealthConcern?Pno=61
Immunization	GET https://mypersonalchart.com/FHIRDemo/api/Immunization?Pno=61
Medication	GET https://mypersonalchart.com/FHIRDemo/api/MedicationStatement?Pno=61

Statement	
Medication Order	GET https://mypersonalchart.com/FHIRDemo/api/MedicationOrder?Pno=61
Allergy Intolerance	GET https://mypersonalchart.com/FHIRDemo/api/AllergyIntolerance?Pno=61
Lab Results	GET https://mypersonalchart.com/FHIRDemo/api/Observation?Pno=61
Diagnostic Report	GET https://mypersonalchart.com/FHIRDemo/api/DiagnosticReport?Pno=61
Vitals	GET https://mypersonalchart.com/FHIRDemo/api/VitalSigns?Pno=61
Procedure	GET https://mypersonalchart.com/FHIRDemo/api/Procedure?Pno=61
Care Plan	GET https://mypersonalchart.com/FHIRDemo/api/CarePlan?Pno=61
Goals	GET https://mypersonalchart.com/FHIRDemo/api/Goal?Pno=61
Device	GET https://mypersonalchart.com/FHIRDemo/api/Device?Pno=61
Smoking Status	GET https://mypersonalchart.com/FHIRDemo/api/SmokingStatus?Pno=61
Get All Data	GET https://mypersonalchart.com/FHIRDemo/api/GetAllData?Pno=61

Note: Pno is mandatory for provider user or other authorized EHR users for all the above endpoints. For patient user Pno should never be passed as parameter with above endpoints. If a patient user does not want to further filter data based on dates then they should always use the above endpoints directly without passing any parameters including pno.

Application Access

User Authentication – Obtaining a access token for registered users only

Prior to making API requests a registered user must obtain an Access Token via the following API

Request URL

- **Base Url:** https://mypersonalchart.com/FHIRDemo/
- **Token Generation:** https://mypersonalchart.com/FHIRDemo/oauth/token
- **Any Endpoints:** https://mypersonalchart.com/FHIRDemo/API/EndpointName/{Parameters}

Access Token

Usage: Validates the user credentials and returns the OAuth 2 access token and refresh token for subsequent requests

URL: https://mypersonalchart.com/FHIRDemo/oauth/token

Method: POST

Header:

Name	Description
Content-Type	application/x-www-form-urlencoded

Parameters:

Parameter	Data Type	Mandatory	Value	Description

grant_type	string	Yes	password	Method for a client application to acquire access token
Username	String	Yes	<Username>	Unique identifier assigned to user on registration.
Secret_key	String	Yes	<Secret Key>	Generated code assigned to user on registration.

Response:

Success (Status 200 OK)

Once you have obtained an OAuth Access Token this will need to be provided as a header in all future requests.

Although subject to change, default values for tokens are:

Access Token - Valid for 30 minutes (1799 Seconds)

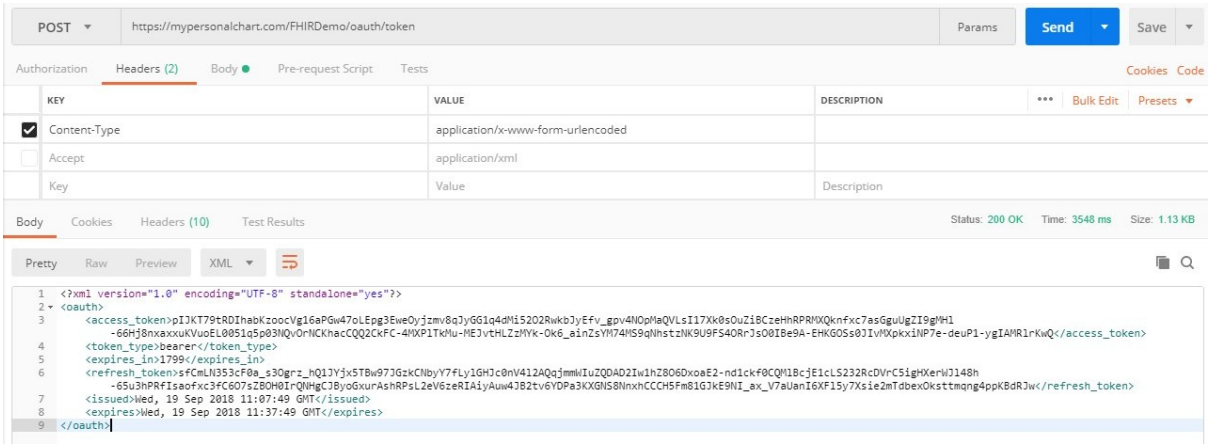
Refresh Token - Valid for 60 minutes (3599 Seconds)

Parameter	Data Type	Value	Description
access_token	string	"[Bearer Token]",	Generated token that user will use to authenticate every request
token_type	String	bearer	Describes the type of token the user is assigned.
expires_in	Numeric	1799	Access tokens lifetime in seconds from when the token was generated
refresh_token	String	"[Refresh Token]"	Token used to obtain a new access token once it expires.
.issued	String	"Wed, 12 Sep 2018 11:33:51 GMT"	Date and time when access token is issued in GMT.
.expires	String	"Wed, 12 Sep 2018 12:03:51 GMT"	Date and time when access token expires in GMT.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<oauth>
<access_token>pIJKT79tRDIhabKzoocVg16aPGw47oLEpg3EweOyjzmv8qJyGG1q4dMi52
O2RwkbJyEfv_gpv4NOpMaQVLsI17Xk0sOuZiBCzeHhRPRMXQknfxc7asGguUgZI9gMHI-
66Hj8nxaxxuKVuoEL0051q5p03NQvOrNCKhacCQQ2CkFC-4MXPlTkMu-MEJvtHLZzMYk-
Ok6_ainZsYM74MS9qNhstzNK9U9FS4ORrJsO0IBe9A-EHKGOSs0JIvMXpkxiNP7e-deuP1-
ygIAMRlrKwQ</access_token>
  <token_type>bearer</token_type>
  <expires_in>1799</expires_in>
<refresh_token>sfCmLN353cF0a_s3Ogrz_hQ1JYjx5TBw97JGzkCNbyY7fLylGHJc0nV4l2AQ
qjmmWIuZQDAD2Iw1hZ8O6DxoaE2-nd1ckf0CQMlBcjE1cLS232RcDVrC5igHXerWJl48h-
```

```
        65u3hPRfIsaofxc3fC6O7sZBOH0IrQNHgCJByoGxurAshRPsL2eV6zeRIAiyAuw4JB2tv6YDPa
        3KXGNS8NnxhCCCH5Fm81GJkE9NI_ax_V7aUanI6XFl5y7Xsie2mTdbexOksttmqng4ppKBd
        RJw</refresh_token>
          <issued>Wed, 19 Sep 2018 11:07:49 GMT</issued>
          <expires>Wed, 19 Sep 2018 11:37:49 GMT</expires>
</oauth>
```

Example



Error

Error code	Description
105	Grant type cannot be blank and should always be password
103	Username/Secret key cannot be blank
104	Unauthorized - The user name or secret key is invalid

Refresh Token

- Refresh token extends the users access over and above the expiry limit of access token.
- User does not have to pass the credentials for receiving the new access token.
- To get a new access token, call the token endpoint of the authorization server and present the refresh token value received in previous response for access token.
- If refresh token has already expired then user will have to send request for receiving new access token presenting the user credentials as explained above.

Pass following parameters to the same endpoint used previously for receiving access token.

Parameter	Data Type	Mandatory	Value	Description
grant_type	string	Yes	refresh_token	Method for a client application to acquire a new access token to extend the access of the API.
refresh_token	String	Yes	<refresh_token>	Present value of refresh_token received in previous response for access token.

Response received will be same as received in call to access token.

Error

Error code	Description
105	Grant type cannot be blank and should always be refresh_token
103	Refresh token cannot be blank
104	Unauthorized - Refresh token value is invalid

Patient Search/Selection

Provider user or other practice users can search for patients they are authorized to see in their EHR and in response will receive patient id's or patient chart number (pno's) that can be subsequently used for fetching those patient's (Common Clinical Data Set) CCDS or CCDA.

Patient user can use the same method to retrieve their demographics information directly without passing any parameters.

URL:
https://mypersonalchart.com/FHIRDemo/api/patientsearch?fname=Elise
https://mypersonalchart.com/FHIRDemo/api/patientsearch?lname=Wo
https://mypersonalchart.com/FHIRDemo/api/patientsearch?gender=M

Usage: Validates user's access token and based on type of user as well as user's validity, returns patient's summarized data in XML format.

Method: GET

Header:

Name	Description
Authorization	**Bearer <access_token>**

Parameters: (Any one of the below parameter is mandatory)

Parameter	Data Type	Mandatory	Value	Description
Pno	String	No	61	Patient Chart Number or Medical Record Number
Fname	String	No	Alice	Patient's given name. Partial given name also can be passed.
Lname	String	No	Smith	Patient's family name. Partial family name also can be passed.
Gender	String	No	M or F	Patient's gender.

BirthDate	String	No	1992-10-17 (YYYY-MM-DD)	Patient's date of birth.
Address	String	No	Morris Ave.	Patient's address.
Zip	String	No	07083	Zip Code of patient location.
Telecom	String	No	4065557896	Patient's contact number.

Response:
Success (Status 200 OK)

```
<PatientsList>
  <patient>
    <Pno>38</Pno>
    <FirstName>Elise</FirstName>
    <LastName>Wong</LastName>
    <MiddleName />
    <BirthDate>1983-06-15T00:00:00-04:00</BirthDate>
    <Gender>F</Gender>
    <Address>9200 Wellington Trail </Address>
    <city>BOZEMAN</city>
    <State>MT</State>
    <Zip>59715</Zip>
    <Telecom>4065557896</Telecom>
  </patient>
</PatientsList>
```

Parameter	Data Type	Value	Description
Pno	String	38	Patient's identification number (Pno) or MRN number
FirstName	String	Elise	Given name of the patient
LastName	String	Wong	Family name of the patient
MiddleName	String		Middle name of the patient
BirthDate	String	1983-06-15	Date of birth of the patient in yyyy-mm-dd format.
Gender	String	F	Birth Sex of the patient.
Address	String	9200 Wellington Trail	Address of patient.
City	String	BOZEMAN	City where patient lives.
State	String	MT	State where patient lives.
Zip	String	59715	Zip Code of patient location.
Telecom	String	4065557896	Patient's contact number.

Error:

See Appendix A.

Fetch Patient Data

Using the access token and patient identification (pno) fetched above a provider or other EHR user can fetch patient's all data (C-CDA) or filter data from the Common Clinical Data Set (CCDS) as per the selected endpoint listed above under title API – Endpoints.

Patient's All Data(C-CDA)

URL: https://mypersonalchart.com/FHIRDemo/api/GetAllData?pno=61
https://mypersonalchart.com/FHIRDemo/api/GetAllData?pno=61&startdate=ge2017-01-01&enddate=le2017-12-31

Usage: Validates user's access token and returns patient's all data(C-CDA) in XML format as per CDA R2 standards.

Method: GET

Header:

Name	Description
Authorization	**Bearer <access_token>**

Parameters:

Parameter	Data Type	Mandatory	Value	Description
Pno	String	Yes (for provider user or other EHR user)	61	Patient Chart Number or Medical Record Number
StartDate	String	No	2010-01-01 (YYYY-MM-DD)	Date from which the data needs to be returned
EndDate	String	No	2018-12-31 (YYYY-MM-DD)	Date to which the data needs to be returned.

Note: Pno is a mandatory parameter for Provider or other EHR users, while patient user should never use Pno as parameter. If they pass pno as parameter it will return error response with code 115 (Check Appendix A for details).

Response XML format: Returns patient data in XML format as per C-CDA R2 standards.
Success (Status 200 OK)

Below is the shortened response for brevity.

```
<?xml version=”1.0” encoding="utf-8"?>
<?xml-stylesheet type="text/xsl" href="CCDA.xsl"?>
<ClinicalDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:sdtc="urn:hl7-org:sdtc" moodCode="EVN"
xmlns="urn:hl7-org:v3">
  <realmCode code="US" />
  <typeId root="2.16.840.1.113883.1.3" extension="POCD_HD000040" />
  <templateId root="2.16.840.1.113883.10.20.22.1.1" extension="2015-08-01" />
  <templateId root="2.16.840.1.113883.10.20.22.1.1" />
  <templateId root="2.16.840.1.113883.10.20.22.1.2" extension="2015-08-01" />
  <templateId root="2.16.840.1.113883.10.20.22.1.2" />
  <id root="f0624e85-14dd-4c21-b56b-ff46e4f0dd59" />
  <code code="34133-9" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"
displayName="Clinical Summary Note" />
  <title>Neighborhood Physicians Clinical Summary</title>
  <effectiveTime value="20180918073033-0400" />
  <confidentialityCode code="N" codeSystem="2.16.840.1.113883.5.25"
codeSystemName="Confidentiality" displayName="Normal" />
  <languageCode code="en-US" />
  <recordTarget typeCode="RCT" contextControlCode="OP">
    <patientRole classCode="PAT">
      <id root="2.16.840.1.113883.1.111.12345" extension="61" />
      <addr use="HP">
        <streetAddressLine>1357, Amber Dr,</streetAddressLine>
        <streetAddressLine nullFlavor="NI" />
        <city>Beaverton</city>
        <state>OR</state>
        <postalCode>97006</postalCode>
        <country>US</country>
      </addr>
      <telecom value="tel:(555)723-1544" use="HP" />
      <telecom value="tel:(555)777-1234" use="MC" />
      <telecom value="mailto:pratiky.hipl+1@gmail.com" use="HP" />
      <patient classCode="PSN" determinerCode="INSTANCE">
        <name use="L">
          <given>Alice</given>
          <given>Jones</given>
          <family>Newman</family>
        </name>
        . . .
        . . .
      </component>
    </structuredBody>
  </component>
</ClinicalDocument>
```

Error:
See Appendix A.

Patient's Common Clinical Data Set

As explained in the above section patient's common clinical data can be fetched by using different endpoints listed above under title API – Endpoints. Base URL, header and body parameters will be same as explained above. Response will be returned in XML format as per CDA R2 standards.

Example URLs for provider user or other EHR user:

Problem/Condition GET https://mypersonalchart.com/FHIRDemo/api/Condition?Pno=61
Immunization GET https://mypersonalchart.com/FHIRDemo/api/Immunization?Pno=61
Medication Statement GET https://mypersonalchart.com/FHIRDemo/api/MedicationStatement?Pno=61
Medication Order GET https://mypersonalchart.com/FHIRDemo/api/MedicationOrder?Pno=61
Allergy Intolerance GET https://mypersonalchart.com/FHIRDemo/api/AllergyIntolerance?Pno=61

Example URLs for patient user:

Problem/Condition GET https://mypersonalchart.com/FHIRDemo/api/Condition
Immunization GET https://mypersonalchart.com/FHIRDemo/api/Immunization
Medication Statement GET https://mypersonalchart.com/FHIRDemo/api/MedicationStatement
Medication Order GET https://mypersonalchart.com/FHIRDemo/api/MedicationOrder
Allergy Intolerance GET https://mypersonalchart.com/FHIRDemo/api/AllergyIntolerance

Qualified searches by date

Start and End dates are passed as query parameters with the URL. Since the URL parameters cannot handle operators (for example, >, <=) these are passed by appending to date values as shown below.

startdate=ge2017-01-01&enddate=le2017-12-31

Following operators are supported:

Operator	Description
Eq	Equal to
Gt	Greater than
Ge	Greater than or equal to
Lt	Less than
Le	Less than or equal to

Note: All API-Endpoints will support qualified searches by date using the above mentioned operators. Only GetAllData API-Endpoint does not support operators explained above. For GetAllData if data for specific date is to be returned then pass the start and end date parameter as explained below:

Specific Date search: startdate=2017-01-01&enddate=2017-01-01
Date range search: startdate=2017-01-01&enddate=2017-12-31

Appendix A

Common codes for error responses:

Error code	Description
100	The resource request contains invalid parameter(s). Invalid parameters: <parameter name>
101	No such record found.
102	The read resource request contained an invalid ID.
107	The read resource request has been merged.
110	No parameters are provided in the search request.
111	Required search parameter missing.
112	The resource request contained an invalid combination of parameters.
115	Required search parameter has an invalid value.