

Open Dental Software

FHIR Interface Specification

For Open Dental 19.1



July 10, 2019

Table of Contents

Table of Contents	2
About Open Dental FHIR	3
Testing Environment	3
Resources	3
AllergyIntolerance.....	4
Appointment.....	4
Condition.....	5
Location.....	6
Medication.....	6
MedicationStatement.....	6
Organization.....	7
Patient.....	7
Practitioner.....	8
Procedure.....	8
ProcedureRequest.....	9
Schedule.....	10
Slot.....	10
Subscription.....	11
Use Cases	11
Use Case 1 – Appointments for Date.....	11
Use Case 2 – List of Patients.....	11
Use Case 3 – Find an opening and create an appointment.....	13
Use Case 4 – Updating an appointment status.....	15
Use Case 5 – Creating and updating a procedure.....	16
Setting Up FHIR	19
Steps to Enable FHIR.....	19
API Keys.....	20

Appendix – CapabilityStatement..... 21

About Open Dental FHIR

Open Dental has a RESTful API service that conforms to the FHIR standard defined by <https://www.hl7.org/fhir>. This FHIR service can be used to look up and create patient and appointments. For a detailed, technical description of Open Dental FHIR's capabilities, see the Capability Statement in Appendix A.

This document describes the FHIR implementation in Open Dental v. 19.1. The version of FHIR used in this implementation is 3.0.1.

These are the resources currently accessible through the FHIR API:

- AllergyIntolerance
- Appointment
- Condition
- Location
- Medication
- MedicationStatement
- Organization
- Patient
- Practitioner
- Procedure
- ProcedureRequest
- Schedule
- Slot
- Subscription

These resources allow the GET method (meaning that the data can be retrieved through the API). Four resources additionally implement the POST and PUT method (resources can be created and updated through the API): Appointment, Patient, Subscription, and Procedure.

Testing Environment

Open Dental hosts a test database for developers to play with FHIR. The base URL is the same for all customers and is <https://api.opendental.com/fhir>. The Capability Statement on this server gives a detailed, technical description of Open Dental's FHIR capabilities. These three headers must be included in requests sent to this server:

Content-Type: application/json

Authorization: FHIRAPIKey=uTCUIGI4sTWfvKWO

Accept: application/json

A browser extension or other software is necessary to send request headers.

Resources

Important notes concerning the functionality of resources are given here. To see a complete list of resources along with the methods and search parameters supported, see the Capability Statement.

AllergyIntolerance

Risk of harmful or undesirable, physiological response which is unique to an individual and associated with exposure to a substance.

Operations Supported: Read

Fields supported:

Field	Comments
identifier	The primary key of the database row (allergy.AllergyNum)
clinicalStatus	active, inactive
verificationStatus	Currently always confirmed
type	allergy, intolerance, none
category	food, medication, environment, none
code	Text description of the allergy
patient	The person who has the intolerance
assertedDate	Date record was believed accurate
lastOccurance	Date of last known occurrence of a reaction
note	Additional notes entered concerning the allergy
reaction	Adverse reaction events linked to exposure to substance

Additional details: <https://www.hl7.org/fhir/allergyintolerance.html>

Appointment

An appointment for a patient

Operations Supported: Read, Create, Update

Fields supported:

Field	Comments
identifier	Corresponds to appointment.AptNum
status	proposed, pending, booked, etc. See below for all statuses
priority	Scale of 1 to 9, 1 as highest, 9 as lowest
start	The beginning of the appointment
end	The end of the appointment
minutesDuration	The number of minutes the appointment lasts
comment	Corresponds to appointment.Note
participant	Used to specify the patient, provider, and operatory
participant.status	On the patient participant, used to update the confirmation status on the appointment

Additional details: <https://www.hl7.org/fhir/appointment.html>

See Use Case 3 to see an example of creating an appointment.
 See Use Case 4 to see an example of updating an appointment.

The statuses on a FHIR Appointment resource correspond to the following appointment statuses in Open Dental:

- proposed – An appointment that is on the Unscheduled List
- pending – An appointment with a status of Scheduled or ASAP
- booked – An appointment with a status of Scheduled or ASAP
- arrived – An appointment that has a value in the Time Arrived field but not in the Time Dismissed field
- fulfilled – An appointment with a status of Complete
- cancelled – An appointment that has been deleted
- noshow – An appointment with a status of Broken

The statuses on the patient participant correspond to the following appointment confirmation statuses in Open Dental:

- needsaction – The offices default unconfirmed status
- accepted – The offices default confirmed status

When creating an appointment, the following fields are required: Patient, location (operator), and status. If a practitioner is not specified, the provider scheduled in that operator for that time slot is used. If there is none, the patient’s primary provider is used. If a secondary provider is not specified, the hygienist for the operator is used if the preference to use the secondary provider from the operator is set. Otherwise, the patient’s secondary provider is used.

When updating an appointment, all fields that are supported by Open Dental's FHIR implementation will be updated. See Use Case 4 for how to update a single field.

Condition

Detailed information about conditions, problems or diagnoses. These are referred to as Problems in Open Dental.

Operations Supported: Read
 Fields supported:

Field	Comments
identifier	The primary key of the database row (disease.DiseaseNum)
clinicalStatus	active, inactive, resolved
verificationStatus	Currently always confirmed
code	May contain the SNOMED, ICD9, or ICD10 code or a text description
subject	The patient who has this condition

onsetDateTime	The date the condition began
abatementDateTime	If/when in resolution/remission
note	Additional information about the Condition

Additional details: <https://www.hl7.org/fhir/condition.html>

Location

A location corresponds to an operatory within Open Dental.

Operations Supported: Read

Fields supported:

Field	Comments
identifier	Corresponds to operatory.OpNum
status	active, inactive
name	Correspondes to operatory.OpName
description	Correspondes to operatory.OpName
mode	Always instance
telecom	Contact information for the clinic the operatory belongs to
address	Always room
managingOrganization	The clinic the operatory belongs to

Additional details: <https://www.hl7.org/fhir/location.html>

Medication

Definition of a Medication.

Operations Supported: Read

Fields supported:

Field	Comments
code	Either an RxNorm or a text description
status	Currently always active
isBrand	True if the medication is a brand, false if it is a generic

Additional details: <https://www.hl7.org/fhir/medication.html>

MedicationStatement

Record of medication being taken by a patient.

Operations Supported: Read

Fields supported:

Field	Comments
identifier	The primary key of the database row (medicationpat.MedicationPatNum)
status	active, completed
medicationCodeableConcept	A text description of the medication
medicationReference	Reference to the Medication resource
effectivePeriod	The interval when the medication was taken
effectiveDateTime	The date when the medication was taken
dateAsserted	When the statement was asserted
subject	The patient who is taking the medication
note	Further information about the medication

Additional details: <https://www.hl7.org/fhir/medicationstatement.html>

Each MedicationStatement will have an effectivePeriod present when the start and the stop dates are entered for the medication and it will have an effectiveDateTime when only the start date is entered.

Organization

One Organization represents the practice as entered within Open Dental under Setup -> Practice. Every other Organization resource is a clinic within Open Dental.

Operations Supported: Read

Fields supported:

Field	Comments
identifier	0 if this Organization is the practice, otherwise clinic.ClinicNum
name	Correspondes to clinic.Abbbr.
telecom	The practice or clinic phone number
address	The practice or clinic physical address
partOf	If this is a clinic, will point to the practice

Additional details: <https://www.hl7.org/fhir/organization.html>

Patient

An individual for whom care is provided

Operations Supported: Read, Create

Fields supported:

Field	Comments
identifier	An identifier for this patient. Corresponds to patient.PatNum
active	Whether this patient's record is in active use.

name	A name associated with the patient.
telecom	A contact detail for the individual.
gender	AdministrativeGender
birthDate	The date of birth for the individual.
deceasedDateTime	The date time deceased for the individual.
address	Address for the individual
maritalStatus	Marital (civil) status of a patient.
photo	Image of the patient. Must include the parameter includePhoto=true
communication	The patient's preferred language
careProvider	Patient's nominated primary care provider.
managingOrganization	Patient's assigned clinic.

Additional details: <https://www.hl7.org/fhir/patient.html>

See Use Case 3 to see an example of creating a patient.

Practitioner

A Practitioner corresponds to a provider in Open Dental, usually a dentist or a hygienist.

Operations Supported: Read

Fields supported:

Field	Comments
identifier	Corresponds to provider.ProvNum
active	True or false
name	The first and last name of the provider
gender	Currently this field will be always unknown
practitionerRole	role will be either 'provider' or 'hygienist'. Specialty is drawn from the provider specialties within Open Dental. These specialties are user-editable and do not conform to any code system.

Additional details: <https://www.hl7.org/fhir/practitioner.html>

Procedure

A Procedure corresponds to a completed procedure in Open Dental..

Operations Supported: Read, Create, Update

Fields supported:

Field	Comments
identifier	Corresponds to procedurelog.ProcNum.
partOf	Corresponds to attached procedure (procgroupitem

	table) if the current procedure is a Group Note.
status	Corresponds to procedurelog.ProcStatus. Always a status of "completed".
category	Corresponds to procedurecode.ProcCat and the matching name of the definition. Only included with read requests. Cannot update.
code	Corresponds to procedurecode.ProcCode.
subject	Reference to the patient attached to the procedure. Corresponds to procedurelog.PatNum.
performedDateTime	The date the procedure was performed. Corresponds to procedurelog.ProcDate.
performer	The performer of the procedure. In this case, the actor corresponds to the practitioner (procedurelog.ProvNum). The onBehalfOf field represents the Organization attached to this procedure (procedurelog.ClinicNum).
bodySite	The location on the body where the procedure occurred. Can correspond to procedurelog.Surf or procedurelog.ToothNum. See section below on Reading/Writing the bodySite.
location	Corresponds to the OpNum of the appointment this procedure is attached to. Only included with read requests. Cannot update.
note	Corresponds to the procnote table. The most recent procnote will be included in the read request.

Additional details: <http://hl7.org/fhir/procedure.html>

See Use Case 5 for an example of inserting and updating a procedure.

Reading/Writing the bodySite:

The bodySite field can store information in the procedurelog.Surf and/or procedurelog.ToothNum. FHIR uses two code systems to implement surface and tooth number. The system for surface will be stored in procedurelog.Surf. If the exact surface is not in the system, such as MOL, multiple surfaces can be included and will be concatenated together (include M, O, and L). The tooth number system will be used to set procedurelog.ToothNum when specifying a specific tooth. The quadrants can also be specified and will be set in procedurelog.Surf. Only set the quadrant if the procedure applies to the entire section of the mouth. See Use Case 5 for an example of inserting and updating the bodySite.

When updating a procedure, all fields that are supported by Open Dental's FHIR implementation will be updated sans location and category. This means if a field is omitted in the update request, it will be set back to its default value.

ProcedureRequest

A ProcedureRequest corresponds to a treatment planned procedure in Open Dental.

Operations Supported: Read, Create

Fields supported:

Field	Comments
identifier	Corresponds to procedurelog.ProcNum.
status	Corresponds to procedurelog.ProcStatus. Always a status of "active".
intent	The intent of the procedure request. Always set to intent of "proposal".
category	Currently this field will be always unknown
code	Corresponds to procedurecode.ProcCode.
subject	Reference to the patient attached to the procedure. Corresponds to procedurelog.PatNum.
occurrenceDateTime	The date the procedure will occur. Corresponds to procedurelog.ProcDate.
authoredOn	The date the procedure was initially created. Only included with read requests. Cannot update.
performer	The performer of the procedure. In this case, the actor corresponds to the practitioner (procedurelog.ProvNum).
bodySite	The location on the body where the procedure occurred. Can correspond to procedurelog.Surf or procedurelog.ToothNum. See section in Procedure: Reading/Writing the bodySite.
note	Corresponds to the procnote table. The most recent procnote will be included in the read request.

Additional details: <http://hl7.org/fhir/stu3/procedurerequest.html>

Schedule

Operations Supported: Read

Fields supported:

Field	Comments
identifier	An alphanumeric string that identifies the date and provider/location
actor	Either a Location or a Practitioner
planningHorizon	One full day

Additional details: <https://www.hl7.org/fhir/schedule.html>

A Schedule resource will exist for a Practitioner if the provider has a schedule set for that day within Open Dental. Every operatory in Open Dental will have a Schedule resource for every day. If no date range is specified for a GET call, then the schedules for the next 28 days will be returned.

Slot

Operations Supported: Read

Fields supported:

Field	Comments
identifier	An alphanumeric string that identifies the date, provider/location, and the start and end.
schedule	Reference to the Schedule resource for this Slot
freeBusyType	free, busy
start	The start date and time of the slot
end	The end date and time of the slot
overbooked	If the provider is scheduled for multiple appointments at this time, will be true

Additional details: <https://www.hl7.org/fhir/slot.html>

Slots are divided into five, ten, or fifteen minute intervals (depending on the appointment time increment preference). A Slot that is linked to a Schedule that is linked to a Practitioner will be considered free if there is a schedule within Open Dental for that provider during that time and the provider is not scheduled for an appointment during that time. A Slot that is linked to a Location will be considered free if that operatory has a provider scheduled for that time and the operatory is an operatory considered for Web Sched and there is no appointment in that Slot.

Subscription

Operations Supported: Read, Create, Update, Delete

Fields supported:

Field	Comments
criteria	The rules for this subscription
contact	Contact details for the subscription
reason	Description of why this subscription was created
status	requested, active, error, off
error	Latest error note
channel	The channel on which to report matches to the criteria
end	When to automatically delete the subscription

Additional details: <https://www.hl7.org/fhir/subscription.html>

Subscriptions can be used to find out about changes that occur to Patients and Appointments. The channel type that is supported is rest-hook, so when a change occurs, an empty POST request is sent to the channel endpoint. A notification will be sent anytime there is a change in the Patient or Appointment database tables, so it is possible that a notification will be sent even though the resource as returned by FHIR has the exact same fields.

To use Subscriptions, the Open Dental eConnector service must be running. The interval at which the

service sends out notifications can be set in the Open Dental program in Setup -> Advanced Setup -> FHIR.

Use Cases

All the following use cases can be performed on the demo server using the specified URLs.

Use Case 1 – Appointments for Date

Find all appointments scheduled for clinic Hogwarts Hospital Wing for January 3rd, 2018.

- The client will find out the id for the Hogwarts Hospital Wing clinic by querying the Organization resource:
`https://api.opendental.com/fhir/organization?name=Hogwarts%20Hospital%20Wing`
- Then the client will need all the Locations that have Hogwarts Hospital Wing for their organization (we'll say the id for Hogwarts Hospital Wing is 1):
`https://api.opendental.com/fhir/location?organization=Organization/1`
- The client can now query the Appointment resources (suppose that the location returned above has the id of 2 and 3):
`https://api.opendental.com/fhir/appointment?location=2,3&date=2018-01-03&status=booked`

Use Case 2 – List of Patients

Keep an updated list of patients who have Madame Pomfrey as a provider

- The client will find out the id for Madame Pomfrey by querying the Practitioner resource:
`https://api.opendental.com/fhir/practitioner?family=pomfrey&given=madame`
- This will return a Practitioner resource. In this case the id for that resource will be 1.
- The client will then issue a query against the patient resources like the following:
`https://api.opendental.com/fhir/patient?careprovider=Practitioner/1`
- To be informed of new patients that are assigned Madame Pomfrey as their provider, the client then creates a Subscription resource like this one:

```
{
  "criteria": "patient?careProvider=Practitioner/1",
  "contact": [
    {
      "system": "email",
      "value": "andrew@friendsofopendental.com",
      "use": "work",
      "rank": 1,
      "period": {
        "start": "2016-08-01T08:00:00"
      }
    }
  ],
  "reason": "To provider surveys to Madame Pomfrey's patients to determine her quality of care",
}
```

```

"status": "requested",
"channel": {
  "type": "rest-hook",
  "endpoint": "http://www.friendsofopendental.com/fhirpatients"
},
"end": "2018-01-01T00:00:00"
}

```

Then the client will perform a POST request it to this URL:
<https://api.opendental.com/fhir/subscription>

- Every few minutes Open Dental's EConnector will check active subscriptions and if any match the criteria, it will send an empty POST request to the endpoint URL (<http://www.friendsofopendental.com/fhirpatients>) that was included when the Subscription was created.
- Subscriptions will need to be enabled within Open Dental. This is done in Setup -> Advanced Setup -> FHIR -> "Process subscription interval in minutes."
- When the client receives that POST request, it can issue the same query with a parameter for the time that it last checked:
https://api.opendental.com/fhir/https://api.opendental.com/fhir/patient?careprovider=Practitioner/1&_lastupdated=ge2016-09-21T18:37:10

Use Case 3 – Find an opening and create an appointment

Find a time where an appointment is not scheduled in the operatory named 'Madame Pomfrey's Operatory' for March 17th, 2017, and create an appointment for a patient named Penelope Clearwater with the provider Madame Pomfrey. Create the patient if she does not exist.

- The client will first find the operatory id for the operatory using this query:
<https://api.opendental.com/fhir/location?name=madame+pomfrey%27s+operatory>
- Using the returned id of 1, issue a query for the schedule for that location for that day:
<https://api.opendental.com/fhir/schedule?actor=Location/1&date=2017-03-17>
- The id from that resource will be 20170317L1. That id will be used for the 'schedule' parameter for the Slot resource: <https://api.opendental.com/fhir/slot?schedule=20170317L1&status=free>
- Using the list of available slots, we can now pick a time for the appointment. We'll pick 8:00 AM for this example.
- Then to find the patient for this appointment, we will issue this query:
<https://api.opendental.com/fhir/patient?family=clearwater&given=penelope>
- If this returns 0 results, we will need to create the patient by submitting a POST request to <https://api.opendental.com/fhir/patient>

```

{
  "name": [
    {
      "use": "usual",
      "family": "Clearwater",
      "given": "Penelope"
    }
  ]
}

```

```

    }
  ],
  "telecom": [
    {
      "system": "phone",
      "value": "(123) 456-7890",
      "use": "home"
    }
  ],
  "gender": "female",
  "birthDate": "1996-09-19"
}

```

- The value from the Location header of the response will be the id for the patient on the appointment (in this case, 157).
- The next step is to find the id for the provider Madam Pomfrey. This is the query that will be used: <https://api.opendental.com/fhir/practitioner?family=pomfrey&given=madame>
- Using the id of 1 returned from the last query, we can construct the appointment resource.

```

{
  "status": "booked",
  "priority": 5,
  "start": "2017-03-17T08:00:00",
  "end": "2017-03-17T08:40:00",
  "minutesDuration": 40,
  "participant": [
    {
      "type": [
        {
          "code": [
            {
              "system": "http://hl7.org/fhir/participant-type",
              "code": "PART"
            }
          ]
        }
      ],
      "actor": {
        "reference": "Patient/157"
      },
      "status": "needsaction"
    },
    {
      "type": [
        {
          "code": [
            {
              "system": "http://hl7.org/fhir/participant-type",
              "code": "PPRF"
            }
          ]
        }
      ]
    }
  ]
}

```

```

    }
  ]
}
],
"actor": {
  "reference": "Practitioner/1"
}
},
{
  "type": [
    {
      "code": [
        {
          "system": "http://hl7.org/fhir/participant-type",
          "code": "PART"
        }
      ]
    }
  ],
  "actor": {
    "reference": "Location/1"
  }
}
]
}

```

- Now, posting to <https://api.opendental.com/fhir/appointment> should return an HTTP status code of 201.

Use Case 4 – Updating an appointment status

Find a specific appointment through a GET request, modify the appointment status field, and PUT the update to the server while including all unmodified fields.

- First, we need to get the appointment we want to modify. In this case, the appointment we want is at ID 4. Send a GET request to the following URL:
<https://api.opendental.com/fhir/appointment/4>
- Now, we need to modify the appointment status field in the returned payload. We want to set the status to “Complete” in Open Dental which is represented by fulfilled.

```

{
  "status": "fulfilled",
  "priority": 5,
  "start": "2017-03-17T08:00:00",
  "end": "2017-03-17T08:40:00",
  "minutesDuration": 40,
  "participant": [
    {
      "type": [

```

```

        "code": [
          {
            "system": "http://hl7.org/fhir/participant-type",
            "code": "PART"
          }
        ]
      },
      "actor": {
        "reference": "Patient/1"
      },
      "status": "needsaction"
    },
    {
      "type": [
        {
          "code": [
            {
              "system": "http://hl7.org/fhir/participant-type",
              "code": "PPRF"
            }
          ]
        }
      ],
      "actor": {
        "reference": "Practitioner/6"
      }
    },
    {
      "type": [
        {
          "code": [
            {
              "system": "http://hl7.org/fhir/participant-type",
              "code": "PART"
            }
          ]
        }
      ],
      "actor": {
        "reference": "Location/1"
      }
    }
  ]
}

```

- Now, PUTing to <https://api.opendental.com/fhir/appointment/4> should return an HTTP status code of 200.

Use Case 5 – Creating and updating a procedure

Use a POST request to a post a procedure. Then, use a PUT request to modify the procedure by changing its tooth, surface, and attached provider.

- First, we need to create the procedure for the given patient. We will send the post request to the following URL:
<https://api.opendental.com/fhir/procedure>

```
{
  "status": "completed",
  "code": {
    "coding": [
      {
        "system": "http://hl7.org/fhir/us/sid/cdt",
        "code": "D2331",
        "display": "resin-based composite - two surfaces,
anterior"
      }
    ]
  },
  "subject": {
    "reference": "Patient/5",
    "display": "Harry Potter"
  },
  "performedDateTime": "2016-07-27T00:00:00",
  "performer": [
    {
      "actor": {
        "reference": "Practitioner/1",
        "display": "Madame S. Pomfrey, DMD"
      },
      "onBehalfOf": {
        "reference": "Organization/1",
        "display": "Hogwarts Hospital Wing"
      }
    }
  ],
  "bodySite": [
    {
      "coding": [
        {
          "system": "http://hl7.org/fhir/ex-tooth",
          "code": "36",
          "display": "36",
          "userSelected": false
        },
        {
          "system": "http://hl7.org/fhir/FDI-surface",
          "code": "D",

```

```

        "display": "Distal",
        "userSelected": false
      },
      {
        "system": "http://hl7.org/fhir/FDI-surface",
        "code": "L",
        "display": "Lingual",
        "userSelected": false
      }
    ]
  },
  ],
  "note": [
    {
      "text": "This is the note."
    }
  ]
}

```

- Now, this procedure has been inserted into the database. Take the id received in the response and use it to post the update. We want to update the tooth to 46, the surface to MOD, and the provider to Practitioner/2.
NOTE: We can simply use the code MOD as it is one of the predefined surfaces for this system. If the surface is not predefined, update the surface similar to how we created it above.
- Send the following JSON as a PUT Request to:
https://api.opendental.com/fhir/procedure/ID_Goes_here
- A 200 status should be received, and the result should include the updated fields.

```

{
  "status": "completed",
  "code": {
    "coding": [
      {
        "system": "http://hl7.org/fhir/us/sid/cdt",
        "code": "D2331",
        "display": "resin-based composite - two surfaces,
anterior"
      }
    ]
  },
  "subject": {
    "reference": "Patient/5",
    "display": "Harry Potter"
  },
  "performedDateTime": "2016-07-27T00:00:00",
  "performer": [

```

```

    {
      "actor":{
        "reference":"Practitioner/2"
      },
      "onBehalfOf":{
        "reference":"Organization/1",
        "display":"Hogwarts Hospital Wing"
      }
    }
  ],
  "bodySite":[
    {
      "coding":[
        {
          "system":"http://hl7.org/fhir/ex-tooth",
          "code":"46",
          "display":"46"
        },
        {
          "system":"http://hl7.org/fhir/FDI-surface",
          "code":"MOD"
        }
      ]
    }
  ],
  "note":[
    {
      "text":"This is the note."
    }
  ]
}

```

Setting Up FHIR

As of 18.4, the FHIR web service is hosted at Open Dental headquarters at the following URL:
<https://api.opendental.com/fhir>.

All requests will be routed through this address to the appropriate office. This moves away from the old method of each office hosting their own FHIR service. The API Key specified in the Authorization header is linked to a specific office.

Steps to Enable FHIR

In order to use FHIR, the office must have an eConnector running.

1. Launch the Open Dental program. Enable FHIR by going to Setup -> Advanced Setup -> FHIR and checking the Enabled checkbox.

API Keys

When requesting data from the FHIR server, an API key must be present in the request header. API keys can be acquired from within the Open Dental program and distributed to developers. A single office can generate as many keys as it wants. These keys are free of charge if they only require the ability to read data from the FHIR server. To write data through FHIR, a small fee will be charged for each key. The Subscription resource, however, can use a read-only key to create and update Subscriptions.

To create a new API key from within Open Dental, go to Setup -> Advanced Setup -> FHIR. Click the Generate Key in the lower left. Once the key is created, you can assign permissions for the resources to which you want this API key to have access.

Enabled

Process subscription interval in minutes. Leave blank to disable subscriptions.

API Keys		
Developer	API Key	Notes
Smiles Software	2oFo3Ev7DkjuGR3x	
AllTeeth.com	nrraqBdnZNN1X8i4	This key is authorized for read operations only.
More Patients 4U	zfgkV6PqSs1Hu0RT	This key is pending approval for write operations. Read operations are authorized.

Generate Key

Permissions for API key:

- AppointmentCreate
- AppointmentRead
- AppointmentUpdate
- PatientCreate
- PatientRead
- Subscriptions
- LocationRead
- OrganizationRead
- PractitionerRead
- ScheduleRead
- AllergyIntoleranceRead
- MedicationRead
- ConditionRead
- ProcedureRequestRead
- ProcedureRequestCreate
- ProcedureRead
- ProcedureCreate
- ProcedureUpdate

Set All Unset All

Close

Appendix – CapabilityStatement

The Capability Statement describes in a machine/human-readable format the implementation of Open Dental FHIR. It contains additional documentation about each resource and the search parameters that can be used with each resource. An up-to-date Conformance Statement can always be retrieved from <https://api.opendental.com/fhir/capabilitystatement>.

```
{
  "fullUrl":"capabilitystatement/open-dental-capability-statement",
  "resource":{
    "url":"https://api.opendental.com/fhir/CapabilityStatement/open-
dental-capability-statement",
    "version":"1.0.1",
    "status":"active",
    "experimental":false,
    "publisher":"Open Dental Software, Inc.",
    "contact":[
      {
        "name":"Brian Miller",
        "telecom":[
          {
            "system":"phone",
            "value":"5033635432",
            "use":"work",
            "rank":1
          },
          {
            "system":"email",
            "value":"brian@opendental.com",
            "use":"work",
            "rank":2
          }
        ]
      },
      {
        "name":"Chris McGehee",
        "telecom":[
          {
            "system":"phone",
            "value":"5033635432",
            "use":"work",
            "rank":2
          },
          {
            "system":"email",
            "value":"chris@opendental.com",
            "use":"work",
```

```

"rank":1
}
]
}
],
"date":"2018-11-28T00:00:00",
"description":"This capability statement describes the RESTful API
that accesses the Open Dental database.",
"requirements":"Published to give operational support to third-party
vendors wishing to interface with the Open Dental database.",
"kind":"instance",
"software":{
"name":"Open Dental",
"version":"19.1.1.0",
"releaseDate":"2018-11-28T00:00:00"
},
"implementation":{
"description":"FHIR server for a dental office running Open Dental",
"url":"https://api.opendental.com/fhir"
},
"fhirVersion":"3.0.1",
"acceptUnknown":"both",
"format":[
"application/json"
],
"rest":[
{
"mode":"server",
"documentation":"This FHIR server can be used to look up patient
demographics and appointments and to find openings where appointments
can be scheduled. A client system can keep itself in sync with this
server through Subscriptions.",
"security":{
"cors":false,
"description":"This FHIR server uses API keys to enforce security.
Each API key has a set of permissions that allow access to certain
resources. These keys can be generated and managed within the Open
Dental program (Setup -> Advanced Setup -> FHIR). These API keys are
only valid for the dental office that generated the API key. Once a
client has an API key, it can include it in the header with the
format 'Authorization: FHIRAPIKey apikey=mysupersecretapikey'."
},
"resource":[
{
"type":"AllergyIntolerance",
"interaction":[
{
"code":"read"
}
],

```

```

{
  "code": "search-type"
}
],
"versioning": "no-version",
"readHistory": false,
"updateCreate": false,
"conditionalCreate": false,
"conditionalUpdate": false,
"conditionalDelete": "not-supported",
"searchParam": [
  {
    "name": "category",
    "type": "token",
    "documentation": "The category of this allergy. Either food, medication, or environment. Determined based on the allergy def allergy type (allergydef.SnomedType)."
  },
  {
    "name": "clinical-status",
    "type": "token",
    "documentation": "Whether the allergy is active or not. Possible values are active or inactive."
  },
  {
    "name": "code",
    "type": "token",
    "documentation": "Code that identifies the allergy or intolerance. Corresponds to the name of the allergy or the name of the medication or the RxCui of the medication (allergydef.Description, medication.MedName, or medication.RxCui)."
  },
  {
    "name": "date",
    "type": "date",
    "documentation": "Date record was believed accurate. Corresponds to the last time this row was updated (allergy.DateTStamp)."
  },
  {
    "name": "identifier",
    "type": "string",
    "documentation": "The logical Id of the AllergyIntolerance. This is the same value as allergy.AllergyNum."
  },
  {
    "name": "last-date",
    "type": "date",
    "documentation": "The date a reaction was recorded. Corresponds to allergy.DateAdverseReaction."
  }
]

```

```

},
{
  "name": "patient",
  "type": "reference",
  "documentation": "Who the sensitivity is for (allergy.PatNum).",
  "target": [
    "Patient"
  ]
},
{
  "name": "type",
  "type": "token",
  "documentation": "The underlying mechanism (allergy or intolerance). Determined based on the allergy def allergy type (allergydef.SnomedType).",
  "interaction": [
    {
      "code": "read"
    },
    {
      "code": "search-type"
    },
    {
      "code": "create"
    },
    {
      "code": "update"
    }
  ],
  "versioning": "no-version",
  "readHistory": false,
  "updateCreate": false,
  "conditionalCreate": false,
  "conditionalUpdate": false,
  "conditionalDelete": "not-supported",
  "searchParam": [
    {
      "name": "patient",
      "type": "reference",
      "documentation": "The patient on the appointment. Corresponds to appointment.PatNum.",
      "target": [
        "Patient"
      ]
    }
  ]
}

```

```

},
{
  "name": "location",
  "type": "reference",
  "documentation": "The operatory in which the appointment is scheduled. Corresponds to appointment.Op.",
  "target": [
    "Location"
  ]
},
{
  "name": "identifier",
  "type": "string",
  "documentation": "The logical Id of the appointment. This is the same value as appointment.AptNum."
},
{
  "name": "practitioner",
  "type": "reference",
  "documentation": "The provider or hygienist on the appointment. Corresponds to appointment.ProvNum or appointment.ProvHyg.",
  "target": [
    "Practitioner"
  ]
},
{
  "name": "status",
  "type": "token",
  "documentation": "The status of the appointment. A status of 'booked' corresponds to an appointment.AptStatus of Scheduled or ASAP. A status of 'cancelled' corresponds to an appointment that has been deleted. A status of 'noshow' corresponds to an appointment.AptStatus of Broken'. A status of 'proposed' corresponds to an appointment.AptStatus of Planned. A status of 'pending' corresponds to a an appointment.AptStatus of UnschedList. A status of 'fulfilled' corresponds to an appointment.AptStatus of 'complete'."
},
{
  "name": "date",
  "type": "date",
  "documentation": "The start time of the appointment. This corresponds to appointment.AptDateTime."
},
{
  "name": "_lastUpdated",
  "type": "date",
  "documentation": "The last time any of the fields in the appointment were modified (this includes fields that are stored in the database and not transferred through FHIR). This corresponds to

```

```

appointment.DateTStamp."
}
]
},
{
  "type": "Condition",
  "interaction": [
    {
      "code": "read"
    },
    {
      "code": "search-type"
    }
  ],
  "versioning": "no-version",
  "readHistory": false,
  "updateCreate": false,
  "conditionalCreate": false,
  "conditionalUpdate": false,
  "conditionalDelete": "not-supported",
  "searchParam": [
    {
      "name": "abatement-date",
      "type": "date",
      "documentation": "Date-related abatements as a dateTime. Equivalent to disease.DateStop."
    },
    {
      "name": "clinical-status",
      "type": "token",
      "documentation": "The clinical status of the condition. Based on disease.ProbStatus."
    },
    {
      "name": "code",
      "type": "token",
      "documentation": "Code for the condition. Uses either diseasedef.SnomedCode, diseasedef.Icd10Code, diseasedef.ICD9Code, or diseasedef.DiseaseName."
    },
    {
      "name": "identifier",
      "type": "string",
      "documentation": "The logical Id of the Condition. Equivalent to disease.DiseaseNum."
    },
    {
      "name": "onset-date",
      "type": "date",

```

```

"documentation":"Date related onsets as a dateTime. Equivalent to
disease.DateStart."
},
{
"name":"patient",
"type":"reference",
"documentation":"Who has the condition? Equivalent to
disease.PatNum."
},
{
"name":"subject",
"type":"reference",
"documentation":"Who has the condition? Equivalent to
disease.PatNum."
}
]
},
{
"type":"Location",
"interaction":[
{
"code":"read"
},
{
"code":"search-type"
}
],
"versioning":"no-version",
"readHistory":false,
"updateCreate":false,
"conditionalCreate":false,
"conditionalUpdate":false,
"conditionalDelete":"not-supported",
"searchParam":[
{
"name":"address",
"type":"string",
"documentation":"The physical address of the clinic to which the
operator is assigned. Corresponds to any value in clinic.Address,
clinic.Address2, clinic.City, clinic.State, clinic.Zip."
},
{
"name":"address-city",
"type":"string",
"documentation":"The city of the clinic to which the operator is
assigned. Corresponds to clinic.City."
},
{
"name":"address-postalcode",

```

```

"type":"string",
"documentation":"The zip code of the clinic to which the operator is
assigned. Corresponds to clinic.Zip."
},
{
"name":"address-state",
"type":"string",
"documentation":"The state of the clinic to which the operator is
assigned. Corresponds to clinic.State."
},
{
"name":"identifier",
"type":"string",
"documentation":"The logical Id of the location. This is the same
value as operator.OperatorNum."
},
{
"name":"name",
"type":"string",
"documentation":"The name or abbreviation of the operator.
Corresponds to operator.OpName or operator.Abbrev."
},
{
"name":"organization",
"type":"reference",
"documentation":"The clinic to which the operator is assigned.
Corresponds to operator.ClinicNum.",
"target":[
"Organization"
]
},
{
"name":"status",
"type":"token",
"documentation":"The status 'active' corresponds to an operator that
is not hidden. The status 'inactive' corresponds to an operator that
is hidden."
}
]
},
{
"type":"Medication",
"interaction":[
{
"code":"read"
},
{
"code":"search-type"
}
]
}

```

```

],
"versioning":"no-version",
"readHistory":false,
"updateCreate":false,
"conditionalCreate":false,
"conditionalUpdate":false,
"conditionalDelete":"not-supported",
"searchParam":[
{
"name":"code",
"type":"token",
"documentation":"Codes that identify this medication. Corresponds to
the name of the medication or the RxCui of the medication
(medication.MedName or medication.RxCui).",
},
{
"name":"identifier",
"type":"string",
"documentation":"The logical Id of the Medication. This is the same
value as medication.MedicationNum."
}
],
},
{
"type":"MedicationStatement",
"interaction":[
{
"code":"read"
},
{
"code":"search-type"
}
],
"versioning":"no-version",
"readHistory":false,
"updateCreate":false,
"conditionalCreate":false,
"conditionalUpdate":false,
"conditionalDelete":"not-supported",
"searchParam":[
{
"name":"code",
"type":"token",
"documentation":"Return statements of this medication code.
Equivalent to medicationpat.MedName."
},
{
"name":"effective",
"type":"date",

```

```

"documentation":"Date when patient was taking (or not taking) the
medication. Based on medication.DateStart and medication.DateStop."
},
{
"name":"identifier",
"type":"string",
"documentation":"The logical Id of the Medication. This is the same
value as medicationpat.MedicationPatNum."
},
{
"name":"medication",
"type":"reference",
"documentation":"Return statements of this medication reference.
Equivalent to medicationpat.MedicationNum."
},
{
"name":"patient",
"type":"reference",
"documentation":"Returns statements for a specific patient.
Equivalent to medicationpat.PatNum."
},
{
"name":"status",
"type":"token",
"documentation":"Return statements that match the given status. Only
'active' and 'completed' are supported."
},
{
"name":"subject",
"type":"reference",
"documentation":"Returns statements for a specific patient.
Equivalent to medicationpat.PatNum."
}
]
},
{
"type":"Organization",
"interaction":[
{
"code":"read"
},
{
"code":"search-type"
}
],
"versioning":"no-version",
"readHistory":false,
"updateCreate":false,
"conditionalCreate":false,

```

```

"conditionalUpdate":false,
"conditionalDelete":"not-supported",
"searchParam":[
{
"name":"address",
"type":"string",
"documentation":"The physical or billing address of the clinic or
practice. Corresponds to any value in clinic.Address,
clinic.Address2, clinic.City, clinic.State, clinic.Zip,
clinic.BillingAddress, clinic.BillingAddress2, clinic.BillingCity,
clinic.BillingState, clinic.BillingZip."
},
{
"name":"address-city",
"type":"string",
"documentation":"The city of the clinic or practice. Corresponds to
clinic.City or clinic.BillingCity."
},
{
"name":"address-postalcode",
"type":"string",
"documentation":"The zip code of the clinic or practice. Corresponds
to clinic.Zip or clinic.BillingZip."
},
{
"name":"address-use",
"type":"string",
"documentation":"All addresses are considered 'work' so that is the
only value that will return results."
},
{
"name":"identifier",
"type":"string",
"documentation":"The logical Id of the clinic or practice. This is
the same value as clinic.ClinicNum, however the dental practice can
be represented as a Clinic resource with a logical Id of 0."
},
{
"name":"name",
"type":"string",
"documentation":"The name of the clinic or practice. Corresponds to
clinic.Abbr."
},
{
"name":"partof",
"type":"reference",
"documentation":"In this implementation, all clinics are a partof the
practice.",
"target":[

```

```

"Organization"
],
"chain":[
"name"
]
}
],
},
{
"type":"Patient",
"interaction":[
{
"code":"read"
},
{
"code":"search-type"
},
{
"code":"create"
}
],
"versioning":"no-version",
"readHistory":false,
"updateCreate":false,
"conditionalCreate":false,
"conditionalUpdate":false,
"conditionalDelete":"not-supported",
"searchParam":[
{
"name":"active",
"type":"token",
"documentation":"Whether a patient is active or not. A value of
'true' corresponds to a patient.PatStatus of Patient. A value of
'false' corresponds to a patient.PatStatus of anything other than
Patient."
},
{
"name":"address",
"type":"string",
"documentation":"The address of the patient. Corresponds to any value
in patient.Address, patient.Address2, patient.City, patient.State,
patient.Zip, or patient.Country."
},
{
"name":"address-city",
"type":"string",
"documentation":"The patient's city. Corresponds to patient.City."
},
{

```

```

"name":"address-postalcode",
"type":"string",
"documentation":"The patient's zip code. Corresponds to patient.Zip."
},
{
"name":"address-state",
"type":"string",
"documentation":"The patient's state. Corresponds to patient.State."
},
{
"name":"address-country",
"type":"string",
"documentation":"The patient's country. Corresponds to
patient.Country."
},
{
"name":"birthdatedate",
"type":"date",
"documentation":"The patient's birthdate. This corresponds to
patient.Birthdate."
},
{
"name":"careprovider",
"type":"reference",
"documentation":"The patient's primary provider. Corresponds to
patient.PriProv.",
"target":[
"Practitioner"
]
},
{
"name":"deathdate",
"type":"date",
"documentation":"The date time the patient passed away. This
corresponds to patient.DateTimeDeceased."
},
{
"name":"deceased",
"type":"token",
"documentation":"Whether a patient deceased or not. A value of 'true'
corresponds to a patient.PatStatus of Deceased or a value is entered
in patient.DateTimeDeceased. A value of 'false' corresponds to a
patient.PatStatus of anything other than Deceased and no value is
entered in papatient.DateTimeDeceased."
},
{
"name":"email",
"type":"string",
"documentation":"The patient's email. Corresponds to patient.Email."

```

```

},
{
  "name": "family",
  "type": "string",
  "documentation": "The patient's family (last) name. Corresponds to patient.LName."
},
{
  "name": "gender",
  "type": "token",
  "documentation": "The patient's gender. Corresponds to patient.Gender. The genders 'other' and 'unknown' corresponds to Unknown in Open Dental."
},
{
  "name": "given",
  "type": "string",
  "documentation": "The patient's given (first or middle) name. Corresponds to patient.FName or patient.MiddleI."
},
{
  "name": "identifier",
  "type": "token",
  "documentation": "This can match either the business id of the patient or the patient's Social Security Number. This corresponds to patient.PatNum or patient.SSN. If intending to search for the SSN, the token's system should be 'http://hl7.org/fhir/sid/us-ssn'."
},
{
  "name": "language",
  "type": "string",
  "documentation": "The patient's primary language. A two- or three-letter ISO language tag is accepted. Because Open Dental only stores a patient's language and not their language region, searches for en-US, en-UK, or en-CA will match all patients whose language is English."
},
{
  "name": "name",
  "type": "string",
  "documentation": "The patient's name. Corresponds to a portion of patient.FName, patient.MiddleI, or patient.LName."
},
{
  "name": "organization",
  "type": "reference",
  "documentation": "The clinic to which the patient is assigned. Corresponds to patient.ClinicNum.",
  "target": [

```

```

"Organization"
]
},
{
"name":"phone",
"type":"string",
"documentation":"A patient's phone number. Corresponds to
patient.HmPhone, patient.WkPhone, or patient.WirelessPhone."
},
{
"name":"telecom",
"type":"string",
"documentation":"A patient's phone number or email. Corresponds to
patient.HmPhone, patient.WkPhone, patient.WirelessPhone, or
patient.Email."
},
{
"name":"_lastUpdated",
"type":"date",
"documentation":"The last time any of the fields in for the patient
were modified (this includes fields that are stored in the database
and not transferred through FHIR). This corresponds to
patient.DateTStamp."
},
{
"name":"includePhoto",
"type":"token",
"documentation":"Custom search parameter. To reduce bandwidth, the
Patient.photo element is excluded by default. To have the photo
included, this parameter must be included with a value of 'true'."
},
{
"name":"phoneNumberMatch",
"type":"string",
"documentation":"Custom search parameter. This parameter can be used
to search for patients whose phone number is a partial match. Any
non-digit characters are ignored both in the parameter value and in
the database value."
}
]
},
{
"type":"Practitioner",
"interaction":[
{
"code":"read"
},
{
"code":"search-type"
}
]
}
}

```

```

}
],
"versioning":"no-version",
"readHistory":false,
"updateCreate":false,
"conditionalCreate":false,
"conditionalUpdate":false,
"conditionalDelete":"not-supported",
"searchParam":[
{
"name":"family",
"type":"string",
"documentation":"The practitioner's family (last) name. Corresponds
to provider.LName."
},
{
"name":"given",
"type":"string",
"documentation":"The practitioner's given (first or middle) name.
Corresponds to provider.FName or provider.MI."
},
{
"name":"identifier",
"type":"string",
"documentation":"The logical id of the practitioner. This is the same
value as provider.ProvNum."
},
{
"name":"name",
"type":"string",
"documentation":"The practitioner's name. Corresponds to a portion of
provider.FName, provider.MI, or patient.provider."
},
{
"name":"role",
"type":"token",
"documentation":"The provider's role in the practice. The parameter
values supported are 'provider' and 'hygienist'. The former
corresponds to provider.IsSecondary=0 and the latter corresponds to
provider.IsSecondary=1."
},
{
"name":"specialty",
"type":"string",
"documentation":"The provider's specialty. These parameter values are
the values set up within Open Dental as provider specialties."
}
]
},

```

```

{
  "type": "Procedure",
  "interaction": [
    {
      "code": "read"
    },
    {
      "code": "search-type"
    },
    {
      "code": "create"
    },
    {
      "code": "update"
    }
  ],
  "versioning": "no-version",
  "readHistory": false,
  "updateCreate": false,
  "conditionalCreate": false,
  "conditionalUpdate": false,
  "conditionalDelete": "not-supported",
  "searchParam": [
    {
      "name": "code",
      "type": "token",
      "documentation": "The code of the procedure. Corresponds to procedurecode.ProcCode."
    },
    {
      "name": "patient",
      "type": "reference",
      "documentation": "The patient's reference. Corresponds to procedurelog.PatNum."
    },
    {
      "name": "date",
      "type": "date",
      "documentation": "The date this procedure occurred. Corresponds to procedurelog.ProcDate."
    },
    {
      "name": "identifier",
      "type": "number",
      "documentation": "The id of the procedure. Corresponds to procedurelog.ProcNum."
    },
    {
      "name": "performer",

```

```

"type":"reference",
"documentation":"The performer for the procedure. Must be a
practitioner. Corresponds to procedurelog.ProvNum."
},
{
"name":"location",
"type":"reference",
"documentation":"The location for the procedure. Corresponds to
procedurelog.ClinicNum."
}
],
},
{
"type":"ProcedureRequest",
"interaction":[
{
"code":"read"
},
{
"code":"search-type"
},
{
"code":"create"
}
],
"versioning":"no-version",
"readHistory":false,
"updateCreate":false,
"conditionalCreate":false,
"conditionalUpdate":false,
"conditionalDelete":"not-supported",
"searchParam":[
{
"name":"code",
"type":"token",
"documentation":"The code of the procedure request. Corresponds to
procedurecode.ProcCode."
},
{
"name":"patient",
"type":"reference",
"documentation":"The patient's reference. Corresponds to
procedurelog.PatNum."
},
{
"name":"occurrence",
"type":"date",
"documentation":"The date this procedure request will occur.
Corresponds to procedurelog.ProcDate."
}
]
}
}

```

```

},
{
  "name": "identifier",
  "type": "number",
  "documentation": "The id of the procedure request. Corresponds to
  procedurelog.ProcNum."
},
{
  "name": "performer",
  "type": "reference",
  "documentation": "The performer for the procedure request. Must be a
  practitioner. Corresponds to procedurelog.ProvNum."
}
]
},
{
  "type": "Schedule",
  "interaction": [
    {
      "code": "read"
    },
    {
      "code": "search-type"
    }
  ],
  "versioning": "no-version",
  "readHistory": false,
  "updateCreate": false,
  "conditionalCreate": false,
  "conditionalUpdate": false,
  "conditionalDelete": "not-supported",
  "searchParam": [
    {
      "name": "actor",
      "type": "reference",
      "documentation": "The actor on the Schedule.",
      "target": [
        "Location",
        "Practitioner"
      ]
    },
    {
      "name": "date",
      "type": "date",
      "documentation": "The date of the schedule. Unless an identifier
      parameter is included, the dates being searched must be a finite date
      range."
    }
  ],
  {

```

```

"name":"identifier",
"type":"string",
"documentation":"The logical id of the practitioner."
}
],
{
"type":"Slot",
"interaction":[
{
"code":"read"
},
{
"code":"search-type"
}
],
"versioning":"no-version",
"readHistory":false,
"updateCreate":false,
"conditionalCreate":false,
"conditionalUpdate":false,
"conditionalDelete":"not-supported",
"searchParam":[
{
"name":"status",
"type":"token",
"documentation":"The free/busy status of the appointment. The code
system must be http://hl7.org/fhir/slotstatus."
},
{
"name":"identifier",
"type":"string",
"documentation":"A Slot Identifier. The logical id of the Slot. A
parameter of either 'identifier' or 'schedule' must be included."
},
{
"name":"schedule",
"type":"reference",
"documentation":"The Schedule Resource that we are seeking a slot
within. A parameter of either 'identifier' or 'schedule' must be
included.",
"target":[
"Schedule"
]
},
{
"name":"start",
"type":"date",
"documentation":"The start time of the slot."
}
]
}

```

```

}
]
},
{
  "type": "Subscription",
  "interaction": [
    {
      "code": "read"
    },
    {
      "code": "create"
    },
    {
      "code": "update"
    },
    {
      "code": "delete"
    },
    {
      "code": "search-type"
    }
  ],
  "versioning": "no-version",
  "readHistory": false,
  "updateCreate": false,
  "conditionalCreate": false,
  "conditionalUpdate": false,
  "conditionalDelete": "not-supported",
  "searchParam": [
    {
      "name": "contact",
      "type": "string",
      "documentation": "The value for a contact field for the Subscription."
    },
    {
      "name": "criteria",
      "type": "string",
      "documentation": "Rule for server push criteria."
    },
    {
      "name": "payload",
      "type": "string",
      "documentation": "\tMimetype to send, or blank for no payload."
    },
    {
      "name": "status",
      "type": "token",
      "documentation": "The status of the Subscription. System must be
http://hl7.org/fhir/ValueSet/subscription-status"
    }
  ]
}

```

```

},
{
  "name": "type",
  "type": "token",
  "documentation": "The type of the Subscription channel. System must be
http://hl7.org/fhir/ValueSet/subscription-channel-type"
},
{
  "name": "url",
  "type": "uri",
  "documentation": "The endpoint of the Subscription channel."
},
{
  "name": "identifier",
  "type": "string",
  "documentation": "The logical id of the practitioner."
}
]
}
],
"interaction": [
{
  "code": "search-system"
}
],
"searchParam": [
{
  "name": "_id",
  "type": "string",
  "documentation": "The logical id of the resource."
}
]
}
],
"id": "open-dental-capability-statement"
}
}

```