



مرکز مدیریت آمار و فناوری اطلاعات
وزارت بهداشت، درمان و آموزش پزشکی

درگاه یکپارچه تبادل اطلاعات سلامت (دیتاس)

سند راهنمای فنی پیاده سازی سرویس مرجع کدینگ سلامت ایران (مکسا)

نگارش ۱.۵

خرداد ۱۴۰۵

شناسنامه سند

درگاه یکپارچه تبادل اطلاعات سلامت (دیتاس)	
نام سند :	سند راهنمای فنی پیاده سازی سرویس مرجع کدینگ سلامت ایران (مکسا)
ارائه دهنده سرویس:	مرکز مدیریت آمار و فناوری اطلاعات وزارت بهداشت، درمان و آموزش پزشکی
نام فایل :	DITAS_MAXAService_V1.5
تاریخ انتشار نگارش اولیه سند :	۱۴۰۰/۰۴/۰۹
تاریخ انتشار نگارش فعلی سند :	۱۴۰۵/۰۳/۱۶
شرح سند :	این سند به منظور بهره‌برداری از سرویس مرجع کدینگ سلامت ایران تهیه شده است.
نویسندگان :	معاونت فنی مرکز مدیریت آمار و فناوری اطلاعات وزارت بهداشت، درمان و آموزش پزشکی

- کلیه حقوق این سند متعلق به مرکز مدیریت آمار و فناوری اطلاعات وزارت بهداشت، درمان و آموزش پزشکی می‌باشد. هرگونه کپی برداری و استفاده غیرمجاز از آن پیگرد قانونی دارد.
- ارائه دهنده سرویس موظف است هرگونه تغییر در ساختار سرویس را به مسئول دیتاس اطلاع دهد و هرگونه ایجاد تغییر در سند بدون هماهنگی با ایشان غیرقانونی است.

تاریخچه ویرایش سند

نویسنده/ویراستار	تاریخ	نگارش	اقدامات
محسن میرزایی	۱۴۰۰/۰۴/۰۹	۱.۰	تدوین اولیه سند
حامد رضایی	۱۴۰۰/۰۴/۲۳	۱.۰.۱	اصلاح و بازبینی محتوایی
پوریا نسیمی	۱۴۰۰/۰۶/۲۶	۱.۰.۲	افزودن مقدمه و تعاریف
حامد رضایی	۱۴۰۰/۰۶/۲۷	۱.۱	اصلاح و بازبینی محتوایی، تایید نسخه ۱.۱
پوریا نسیمی، شیرین شاطریان	۱۴۰۱/۰۶/۱۷	۱.۱.۱	اصلاح و بازبینی محتوایی
حامد رضایی	۱۴۰۱/۰۶/۱۸	۱.۱.۲	اصلاح ساختار و بازبینی محتوایی
ابراهیم کشاورز صفری	۱۴۰۱/۰۶/۱۹	۱.۱.۳	اصلاح قالب، بازبینی محتوایی و نگارشی
افشین شرفی	۱۴۰۱/۰۶/۲۰	۱.۱.۴	بازبینی و ویرایش نمونه‌های ورودی و خروجی توابع
حامد رضایی	۱۴۰۱/۰۶/۲۱	۱.۲	اصلاح ساختار
ابراهیم کشاورز صفری	۱۴۰۱/۰۶/۲۱	۱.۲	تایید نسخه ۱.۲
ابراهیم کشاورز صفری	۱۴۰۱/۱۲/۰۱	۱.۳	بروزرسانی کدهای نمونه ورودی توابع سرویس
شیرین شاطریان	۱۴۰۱/۱۲/۰۱	۱.۳	کنترل کیفی
پوریا نسیمی	۱۴۰۵/۰۲/۲۱	۱.۴	- اختیاری شدن تاریخ در ورودی تابع CtsTerminologies - اختیاری شدن تاریخ، افزودن دسته بندی، شماره صفحه و اندازه صفحه در ورودی تابع SearchTerminology - اضافه شدن توضیحات، دسته بندی ها و والد (از نوع DoCodedTextDto) به خروجی کدینگ های تابع SearchTerminology - افزودن شماره صفحه و اندازه صفحه در ورودی تابع TerminologiesTermByVersion - اضافه شدن توضیحات، دسته بندی ها و والد (از نوع DoCodedTextDto) به خروجی کدینگ های تابع TerminologiesTermByVersion - تابع TerminologyCategories جهت دریافت دسته بندی های یک ترمینولوژی
معصومه خانی لیلا کرمی	۱۴۰۵/۰۳/۱۶	۱.۵	کنترل کیفی

تاییدکننده سند

نویسنده/ویراستار	تاریخ	نگارش	وضعیت
سید رضا مظهری	۱۴۰۵/۰۳/۱۶	۱.۵	تایید نهایی

فهرست مطالب

فهرست مطالب	۴
فهرست جداول	۵
مقدمه	۶
تعاریف	۶
نحوه احراز هویت کاربر دیتاس	۷
دسترسی به خدمات	۸
تابع دریافت توکن دیتاس	۸
تابع بروزرسانی توکن دیتاس	۹
نحوه فراخوانی توابع سرویس مرکز کدینگ سلامت ایران (مکسا)	۱۱
پارامترهای ورودی Header توابع سرویس مکسا	۱۱
تابع Terminology	۱۱
تابع SearchTerminology	۱۲
تابع ValidateCts	۱۶
تابع CtsTerminologies	۱۸
تابع TerminologyValueByVersion	۲۰
تابع TerminologiesTermByVersion	۲۱
تابع TerminologyVersions	۲۳
تابع TerminologyCategories	۲۵
پیوست‌ها	۲۷
پیوست ۱- ساختار خروجی فراخوانی توابع دیتاس	۲۷
پیوست ۲- کدهای وضعیت پاسخ دیتاس نسبت به فراخوانی توابع	۲۸

فهرست جداول

جدول ۱- پارامترهای ورودی Header تابع دریافت توکن	۸
جدول ۲- پارامترهای ورودی Body تابع دریافت توکن	۸
جدول ۳- پارامترهای خروجی Body تابع دریافت توکن دیتاس	۹
جدول ۴- پارامترهای خروجی Header تابع دریافت توکن	۹
جدول ۵- پارامترهای ورودی Header تابع بروزرسانی توکن دیتاس	۱۰
جدول ۶- پارامترهای خروجی تابع بروزرسانی توکن	۱۰
جدول ۷- پارامترهای ورودی Header توابع سرویس مکسا	۱۱
جدول ۸- پارامترهای ورودی Body تابع Terminology	۱۲
جدول ۹- پارامترهای ورودی Body تابع SearchTerminology	۱۳
جدول ۱۰- پارامترهای ورودی Body تابع ValidateCts	۱۷
جدول ۱۱- پارامترهای ورودی Body تابع CtsTerminologies	۱۹
جدول ۱۲- پارامترهای ورودی Body تابع TerminologyValueByVersion	۲۱
جدول ۱۳- پارامترهای ورودی Body تابع TerminologiesTermByVersion	۲۲
جدول ۱۴- پارامترهای خروجی Body تابع TerminologiesTermByVersion	۲۳
جدول ۱۵- پارامترهای ورودی Body تابع TerminologyVersions	۲۴
جدول ۱۶- پارامترهای خروجی Body تابع TerminologyVersions	۲۵
جدول ۱۷- پارامترهای ورودی Body تابع TerminologyCategories	۲۵



مقدمه

به استناد ماده ۳۵ قانون برنامه پنجم توسعه جمهوری اسلامی ایران مصوب ۱۳۸۹/۱۰/۱۵ و ماده ۷۴ قانون برنامه ششم توسعه کشور، به منظور حفظ یکپارچگی در مدیریت دانش و اطلاعات حوزه سلامت و تعامل موثر سیستم های یکپارچه مبتنی بر فناوری اطلاعات بیمه ای در وزارت رفاه و تامین اجتماعی و سازمان ها و مراکز بیمه ای با سامانه پرونده الکترونیکی سلامت ایرانیان کمیته مرجع کدینگ سلامت ایران تاسیس شد. به منظور ساماندهی به کدینگ مصوب کمیته مذکور و ارائه مکانیزم دسترسی آزاد به این اطلاعات، سامانه مکسا توسط مرکز آمار و فناوری اطلاعات وزارت بهداشت راه اندازی گردید.

تعاریف

دیتاس (DITAS)

به درگاه یکپارچه تبادل اطلاعات سلامت اطلاق می شود.

ارائه دهنده سرویس

معاونت/ سازمان/ مرکزی که سرویس الکترونیکی آن بر بستر دیتاس پیاده سازی شده باشد.

احراز هویت کاربر (Authorization)

از این پارامتر جهت احراز هویت کاربران دیتاس استفاده می شود که به صورت OAuth2 می باشد.

شناسه سرویس ۱ (PID)

شناسه یکتای ارائه شده به کاربر برای فراخوانی سرویس.

ترمینولوژی

علم مطالعه واژه ها و اصطلاحات تخصصی است که در این سند منظور ترمینولوژی های حوزه سلامت مانند SNOMEDCT, RVU3, Loinc و غیره می باشد.

کدینگ

به مفهوم کدگذاری مفاهیم گفته می شود که در نهایت می تواند منجر به ایجاد یک سیستم و ترمینولوژی گردد.

¹ Package ID



مکسا

مرجع کدینگ سلامت ایران

نگاشت

تناظر کدهای مفاهیم از ترمینولوژی های مختلف را نگاشت یا Mapping می گویند.

سامانه اطلاعاتی مراکز ارائه دهنده خدمات سلامت (POCS)

سامانه های اطلاعاتی مراکز ارائه دهنده خدمات سلامت شامل تمامی نرم افزارها و سیستم های اطلاعاتی هستند که در مراکز ارائه دهنده خدمات سلامت مانند بیمارستانها، درمانگاه ها، آزمایشگاه های بالینی، مراکز تصویر برداری، خانه بهداشت و غیره نصب بوده و قسمتی از اطلاعات درمان و بهداشت شهروندان را در خود ذخیره و مدیریت می کنند. به اختصار سامانه اطلاعاتی مراکز ارائه دهنده خدمات سلامت را POCS (Point of Care System) می نامیم.

سامانه پرونده الکترونیکی سلامت (سپاس)

سپاس مخفف سامانه پرونده الکترونیکی سلامت می باشد که موضوع ماده ۳۵ قانون برنامه پنجم توسعه کشور است. پرونده الکترونیکی سلامت مجموعه ای از کلیه اطلاعات مرتبط با سلامت شهروندان، از پیش از تولد (شامل: اطلاعات دوران جنینی و ماقبل آن، مانند اطلاعات مربوط به لقاح آزمایشگاهی) تا پس از مرگ (مانند: اطلاعات به دست آمده از اتوپسی، محل دفن و...) است که به صورت مداوم و با گذشت زمان به شکل الکترونیکی ذخیره می شود و در صورت نیاز، بدون ارتباط با مکان یا زمان خاص، تمام یا بخشی از آن، به سرعت در دسترس افراد مجاز قرار می گیرد. سپاس مخفف «سامانه پرونده الکترونیکی سلامت ایران» می باشد و طرح ایجاد و توسعه این سامانه، تحت عنوان طرح سپاس شناخته می شود. در واقع این طرح، شامل مجموعه فعالیت هایی است که در محورهای مختلف صورت گرفته و در نهایت منجر به ایجاد یک نظام اطلاعاتی یکپارچه برای ثبت، بازیابی و تبادل اطلاعات سلامت شهروندان خواهد شد و برای ارائه خدمات نوین الکترونیکی در حوزه سلامت، بستر مناسبی فراهم خواهد کرد.

نحوه احراز هویت کاربر دیتاس

باتوجه به اینکه تمامی خدمات ارائه شده نیازمند دسترسی به توکن خاص هر مرکز می باشد این تابع به منظور ارائه توکن مورد استفاده قرار می گیرد. توکن دریافت شده توسط این تابع در بخش Header درخواست ها ثبت می گردد. خروجی این تابع توکن



دسترسی‌ای با تاریخ انقضاء مشخص می باشد.

دسترسی به خدمات

دسترسی به خدمات ارائه شده از طریق آدرس زیر امکان پذیر است:

Base Url: <https://apigateway.behdasht.gov.ir>

تمامی خدمات نیازمند توکن دسترسی هستند که در بخش هدر درخواست‌ها قرار می گیرند.

تابع دریافت توکن دیتاس^۱

این تابع با استفاده از ورودی‌های مخصوص به هر کاربر، توکن دسترسی‌ای با تاریخ انقضاء مشخص را برمی گرداند.

- قالب ورودی

```
POST /oauth/token HTTP/1.1
Host: apigateway.behdasht.gov.ir
Authorization: Basic {authorizationCode}
grant_type=password&username=yourusername&password=yourpassword
```

جدول ۱- پارامترهای ورودی Header تابع دریافت توکن

ردیف	نام پارامتر	نوع داده	الزامی	توضیحات
۱	authorizationCode	basic auth	بله	شامل Client_Id:Client_Secret مخصوص به هر کاربر است که به صورت Base64 ارسال می شود.

جدول ۲- پارامترهای ورودی Body تابع دریافت توکن

ردیف	نام پارامتر	مقدار ورودی	الزامی	توضیحات
۱	grant_type	password	بله	مدل دسترسی
۲	username	Your username	بله	نام کاربری شما
۳	password	Your password	بله	رمز عبور شما

- نمونه ورودی

```
POST /oauth/token HTTP/1.1
```

^۱ Get Token





Host: apigateway.behdasht.gov.ir
Authorization: *****=
grant_type=password&username=yourusername&password=yourpassw

- نمونه خروجی موفق

```
{
  "access_token": "c8abceda-aa31-4a7e-95c0-213e5709e6b6",
  "token_type": "bearer",
  "refresh_token": "84dc3bf1-7342-4c5e-adc3-c2304583ae02",
  "expires_in": 763,
  "scope": "trust read write"
}
```

- نمونه خروجی ناموفق

```
{
  "error": "invalid_grant",
  "error_description": "Bad credentials"
}
```

جدول ۳- پارامترهای خروجی Body تابع دریافت توکن دیتاس

ردیف	نام پارامتر	نوع داده	توضیحات
۱	access_token	string	توکن دسترسی
۲	refresh_token	string	بروزرسانی توکن
۳	scope	string	محدوده دسترسی توکن
۴	token_type	string	نوع توکن
۵	expires_in	string	مقدار زمان باقیمانده تا منقضی شدن توکن (بر اساس ثانیه)

جدول ۴- پارامترهای خروجی Header تابع دریافت توکن

ردیف	نام پارامتر	نوع داده	توضیحات
۱	requestId	string	شناسه پیگیری برای رهگیری درخواست

پارامترهای خروجی Header همه توابع دیتاس مطابق جدول ۴ می باشد.

تابع بروزرسانی توکن دیتاس^۱

این تابع، توکن دریافتی از تابع دریافت توکن دیتاس را بروزرسانی می کند. پارامترهای خروجی Header این تابع مطابق جدول ۶ می باشد.

^۱ Refresh Token



- قالب ورودی

POST /oauth/token HTTP/1.1
Host: apigateway.behdasht.gov.ir
Authorization: Basic {authorizationCode}
grant_type=refresh_token&refresh_token={refreshToken}

جدول ۵- پارامترهای ورودی Header تابع بروزرسانی توکن دیتاس

ردیف	نام پارامتر	نوع داده	الزامی	توضیحات
۱	Authorization	basic auth	بله	شامل Client_Id:Client_Secret مخصوص به هر کاربر است که به صورت Base64 ارسال می شود.

- نمونه ورودی

POST /oauth/token HTTP/1.1
Host: apigateway.behdasht.gov.ir
Authorization: *****=
grant_type=refresh_token&refresh_token=f998a112-b166-4177-8e2e-c2d89fedb352

- نمونه خروجی موفق

```
{
  "access_token": "529d80a1-e8af-44b2-9dac-565cff4258f1",
  "token_type": "bearer",
  "refresh_token": "f998a112-b166-4177-8e2e-c2d89fedb352",
  "expires_in": 899,
  "scope": "trust read write"
}
```

- نمونه خروجی ناموفق

```
{
  "error": "invalid_grant",
  "error_description": "Bad credentials"
}
```

جدول ۶- پارامترهای خروجی تابع بروزرسانی توکن

ردیف	نام پارامتر	نوع داده	توضیحات
۱	accessToken	string	توکن دسترسی
۲	refreshToken	string	بروز رسانی توکن
۳	scope	string	محدوده دسترسی توکن



ردیف	نام پارامتر	نوع داده	توضیحات
۴	tokenType	string	نوع توکن
۵	expiresIn	string	مقدار زمان باقیمانده تا انقضا شدن توکن (بر اساس ثانیه)

نحوه فراخوانی توابع سرویس مرکز کدینگ سلامت ایران (مکسا)

پارامترهای ورودی Header توابع سرویس مکسا

در تمامی توابع این سرویس، از پارامترهای جدول ۷ به عنوان ورودی Header استفاده می شود.

جدول ۷- پارامترهای ورودی Header توابع سرویس مکسا

ردیف	نام پارامتر	نوع داده	الزامی	توضیحات
۱	Content-Type	String	بله	نوع محتوای ورودی
۲	Pid	String	بله	شناسه سرویس
۳	Authorization	String	بله	مقدار توکن بازگشتی از تابع دریافت یا بروزرسانی توکن

تابع Terminology

از این تابع به منظور دریافت مقدار متناظر یک کد از یک ترمینولوژی مشخص استفاده می شود.

- قالب ورودی

```
GET /api/client/apim/v1/maksaservice/gwmaksa/Terminology
HTTP/1.1
Host: apigateway.behdasht.gov.ir
pid: {Your Pid}
Content-Type: application/json
Authorization: Bearer {{token}}
{
  "terminologyId": "thrtaEHR.Religion",
  "codedString": "1"
}
```

- نمونه ورودی

```
curl --location --
request GET 'https://apigateway.behdasht.gov.ir/api/client/apim/v1/maksaservice/gw
maksaservice/gwmaksa/Terminology' \
--header 'pid: {Your Pid}' \
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer {{token}}' \
```



```
--data-raw '{
  "terminologyId": "thrtaEHR.Religion",
  "codedString": "1"
}'
```

جدول ۸- پارامترهای ورودی Body تابع Terminology

ردیف	نام پارامتر	نوع داده	الزامی	توضیحات
۱	terminologyId	String	بله	عنوان ترمینولوژی
۲	codedString	String	بله	کد

- نمونه خروجی موفق

```
{
  "result": {
    "data": "\"اسلام\"",
    "status": {
      "statusCode": 200
    }
  },
  "status": {
    "statusCode": 200,
    "message": "OK"
  }
}
```

- نمونه خروجی ناموفق

```
{
  "result": {
    "status": {
      "statusCode": 404
    }
  },
  "status": {
    "statusCode": 200,
    "message": "OK"
  }
}
```

خروجی این تابع رشته‌ای است که مقدار متناظر با کد و ترمینولوژی مربوطه را برمی‌گرداند.

تابع SearchTerminology

این تابع به منظور دریافت کدها و مقادیر مربوط به یک ترمینولوژی از یک تاریخ مشخص به بعد ایجاد شده‌اند استفاده می‌شود.

- قالب ورودی

```
POST /api/client/apim/v1/maksaservice/gwmaksa/SearchTerminology
HTTP/1.1
Host: apigateway.behdasht.gov.ir
pid: {Your Pid}
```



```
Content-Type: application/json
Authorization: Bearer {{token}}
{
  "TerminologyId": "ContactType",
  "FromDate": {
    "isoStringField": "",
    "yearField": 1390,
    "monthField": 10,
    "dayField": 10,
    "hourField": 10,
    "minuteField": 10,
    "secondField": 10
  },
  "Category": null,
  "PageNumber": 1,
  "PageSize": 100
}
```

- نمونه ورودی

```
curl --location --
request POST 'https://apigateway.behdasht.gov.ir/api/client/apim/v1/maksaservice/g
wmaksa/SearchTerminology' \
--header 'pid: {Your Pid}' \
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer {{token}}' \
--data-raw '{
  "TerminologyId": "ContactType",
  "FromDate": {
    "isoStringField": "",
    "yearField": 1390,
    "monthField": 10,
    "dayField": 10,
    "hourField": 10,
    "minuteField": 10,
    "secondField": 10
  },
  "Category": null,
  "PageNumber": 1,
  "PageSize": 100
}'
```

جدول ۹- پارامترهای ورودی Body تابع SearchTerminology

ردیف	نام پارامتر	نوع داده	الزامی	توضیحات
۱	TerminologyId	String	بله	عنوان ترمینولوژی
۲	FromDate	Object	اختیاری	از تاریخ، فیلهای آن شامل yearField برای سال، monthField برای ماه، dayField برای



ردیف	نام پارامتر	نوع داده	الزامی	توضیحات
				روز، hourField برای ساعت، minuteField برای دقیقه و secondField برای ثانیه می باشند.
۳	Category	String	اختیاری	عنوان دسته بندی
۴	PageNumber	Int	بله	شماره صفحه - پیش فرض ۱ می باشد.
۵	PageSize	Int	بله	اندازه صفحه - پیش فرض ۱ می باشد و حداکثر مقدار می تواند ۲۰۰ باشد.

- نمونه خروجی موفق

```
{
  "result": {
    "data": [
      {
        "valueField": "تلفن منزل",
        "terminology_idField": "ContactType",
        "coded_stringField": "1",
        "description": null,
        "categories": [],
        "parent": null
      },
      {
        "valueField": "پیجر شخصی",
        "terminology_idField": "ContactType",
        "coded_stringField": "10",
        "description": null,
        "categories": [],
        "parent": null
      },
      {
        "valueField": "پیجر محل کار",
        "terminology_idField": "ContactType",
        "coded_stringField": "11",
        "description": null,
        "categories": [],
        "parent": null
      },
      {
        "valueField": "تلفن همراه محل کار",
        "terminology_idField": "ContactType",
        "coded_stringField": "12",
        "description": null,
        "categories": [],
        "parent": null
      },
      {
        "valueField": "نشانی منزل",
        "terminology_idField": "ContactType",
        "coded_stringField": "13"
      }
    ]
  }
}
```



```
      "description": null,  
      "categories": [],  
      "parent": null  
    },  
    {  
      "valueField": "نشانی محل کار",  
      "terminology_idField": "ContactType",  
      "coded_stringField": "14"  
      "description": null,  
      "categories": [],  
      "parent": null  
    },  
    {  
      "valueField": "تلفن محل کار",  
      "terminology_idField": "ContactType",  
      "coded_stringField": "2"  
      "description": null,  
      "categories": [],  
      "parent": null  
    },  
    {  
      "valueField": "تلفن همراه شخصی",  
      "terminology_idField": "ContactType",  
      "coded_stringField": "3"  
      "description": null,  
      "categories": [],  
      "parent": null  
    },  
    {  
      "valueField": "فاکس منزل",  
      "terminology_idField": "ContactType",  
      "coded_stringField": "4"  
      "description": null,  
      "categories": [],  
      "parent": null  
    },  
    {  
      "valueField": "فاکس محل کار",  
      "terminology_idField": "ContactType",  
      "coded_stringField": "5"  
      "description": null,  
      "categories": [],  
      "parent": null  
    },  
    {  
      "valueField": "پست الکترونیکی شخصی",  
      "terminology_idField": "ContactType",  
      "coded_stringField": "6"  
      "description": null,  
      "categories": [],  
      "parent": null  
    },  
    {  
      "valueField": "پست الکترونیکی محل کار",  
      "terminology_idField": "ContactType",  
      "coded_stringField": "7"
```





```

        "description": null,
        "categories": [],
        "parent": null
    },
    {
        "valueField": "نشانی وبسایت شخصی",
        "terminology_idField": "ContactType",
        "coded_stringField": "8"
        "description": null,
        "categories": [],
        "parent": null
    },
    {
        "valueField": "نشانی وبسایت محل کار",
        "terminology_idField": "ContactType",
        "coded_stringField": "9"
        "description": null,
        "categories": [],
        "parent": null
    }
],
"status": {
    "statusCode": 200
}
},
"status": {
    "statusCode": 200,
    "message": "OK"
}
}

```

- نمونه خروجی ناموفق

```

{
  "result": {
    "status": {
      "statusCode": 404
    }
  },
  "status": {
    "statusCode": 200,
    "message": "OK"
  }
}

```

خروجی این تابع آرایه‌ای از مقادیر، کدها و ترمینولوژی است که از تاریخ مشخص به بعد ایجاد شده‌اند.

تابع ValidateCts

از این تابع به منظور صحت‌سنجی مقدار متناظر با یک کد در یک ترمینولوژی مشخص استفاده می‌شود.



- قالب ورودی

```
POST /api/client/apim/v1/maksaservice/gwmaksa/ValidateCts
HTTP/1.1
Host: apigateway.behdasht.gov.ir
pid: {Your Pid}
Content-Type: application/json
Authorization: Bearer {{token}}
{
  "DoCodedText": {
    "valueField": "پست الکترونیکی شخصی",
    "terminology_idField": "ContactType",
    "coded_stringField": "6"
  }
}
```

- نمونه ورودی

```
curl --location --
request POST 'https://apigateway.behdasht.gov.ir/api/client/apim/v1/maksaservice/gwmaksa/ValidateCts' \
--header 'pid: {Your Pid}' \
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer {{token}}' \
--data-raw '{
  "DoCodedText": {
    "valueField": "پست الکترونیکی شخصی",
    "terminology_idField": "ContactType",
    "coded_stringField": "6"
  }
}'
```

جدول ۱۰- پارامترهای ورودی Body تابع ValidateCts

ردیف	نام پارامتر	نوع داده	الزامی	توضیحات
۱	DoCodedText.valueField	String	بله	مقدار
۲	DoCodedText.terminology_idField	String	بله	ترمینولوژی
۳	DoCodedText.coded_stringField	String	بله	کد

- نمونه خروجی موفق

```
{
  "result": {
    "data": "true",
    "status": {
      "statusCode": 200
    }
  },
  "status": {
    "statusCode": 200,
    "message": "OK"
  }
}
```



```
}  
}
```

- نمونه خروجی ناموفق

```
{  
  "result": {  
    "status": {  
      "statusCode": 404  
    }  
  },  
  "status": {  
    "statusCode": 200,  
    "message": "OK"  
  }  
}
```

خروجی این تابع یک مقدار Boolean است. در صورت معتبر بودن ورودی‌ها مقدار true و در غیر اینصورت مقدار false برگردانده می‌شود.

تابع CtsTerminologies

از این تابع به منظور دریافت لیست ترمینولوژی‌هایی که از تاریخ خاص به بعد تعریف شده‌اند استفاده می‌شود.

- قالب ورودی

```
POST /api/client/apim/v1/maksaservice/gwmaksa/CtsTerminologies  
HTTP/1.1  
Host: apigateway.behdasht.gov.ir  
pid: {Your Pid}  
Content-Type: application/json  
Authorization: Bearer {{token}}  
{  
  "FromDate": {  
    "yearField": 1390,  
    "monthField": 10,  
    "dayField": 10,  
    "hourField": 10,  
    "minuteField": 10,  
    "secondField": 10  
  }  
}
```

- نمونه ورودی

```
curl --location --  
request POST 'https://apigateway.behdasht.gov.ir/api/client/apim/v1/maksaservice/gwmaksa/CtsTerminologies' \  
--header 'pid: {Your Pid}' \  
--header 'Content-Type: application/json' \  
--header 'Authorization: Bearer {{token}}' \  
--data-raw '{  
  "FromDate": {
```





```
"yearField": 1390,
"monthField": 10,
"dayField": 10,
"hourField": 10,
"minuteField": 10,
"secondField": 10
},
}
```

جدول ۱۱- پارامترهای ورودی Body تابع CtsTerminologies

ردیف	نام پارامتر	نوع داده	الزامی	توضیحات
۱	FromDate	Object	اختیاری	از تاریخ، فیلدهای آن شامل yearField برای سال، monthField برای ماه، dayField برای روز، hourField برای ساعت، minuteField برای دقیقه و secondField برای ثانیه می‌باشند.

- نمونه خروجی موفق

```
{
  "result": {
    "data": [
      "ICD10",
      "ICD10-FA",
      "ISO_3166-1",
      "Country",
      "thrtaEHR.admissionType",
      "thrtaEHR.birthDateAccuracy",
      "DateAccuracy",
      "California",
      "thrtaEHR.conditionOnDischarge",
      "thrtaEHR.BedStatus",
      "thrtaEHR.RelatedPerson",
      "thrtaEHR.WardType",
      "ThrtaEHR",
      "CDT",
      "MEQ",
      "SNOMEDCT",
      "LNC",
      "eRX",
      "UCUM",
      "ATC",
      "countryDivisions",
      "IRC",
      "CDT-FA",
      "ADA"
    ],
    "status": {
      "statusCode": 200
    }
  }
}
```



```
},  
"status": {  
  "statusCode": 200,  
  "message": "OK"  
}  
}
```

- نمونه خروجی ناموفق

```
{  
  "result": {  
    "status": {  
      "statusCode": 404  
    }  
  },  
  "status": {  
    "statusCode": 200,  
    "message": "OK"  
  }  
}
```

خروجی این تابع آرایه‌ای از عنوان ترمینولوژی‌هایی است که از تاریخ موردنظر به بعد تعریف شده‌اند.

تابع TerminologyValueByVersion

از این تابع به منظور دریافت مقدار متناظر با یک کد در یک نسخه خاص از ترمینولوژی مشخص استفاده می‌شود.

- قالب ورودی

```
POST /api/client/apim/v1/maksaservice/gwmaksa/TerminologyValueByVersion  
HTTP/1.1  
Host: apigateway.behdasht.gov.ir  
pid: {Your Pid}  
Content-Type: application/json  
Authorization: Bearer {{token}}  
{  
  "terminologyId": "ERX",  
  "codedString": "60002",  
  "version": "10000"  
}
```

- نمونه ورودی

```
curl --location --  
request POST 'https://apigateway.behdasht.gov.ir/api/client/apim/v1/maksaservice/g  
wmaksa/TerminologyValueByVersion' \  
--header 'pid: {Your Pid}' \  
--header 'Content-Type: application/json' \  
--header 'Authorization: Bearer {{token}}' \  
--data-raw '{  
  "terminologyId": "ERX",  
  "codedString": "60002",  
  "version": "10000"  
}'
```



جدول ۱۲- پارامترهای ورودی Body تابع TerminologyValueByVersion

ردیف	نام پارامتر	نوع داده	الزامی	توضیحات
۱	terminologyId	String	بله	ترمینولوژی
۲	codedString	String	بله	کد
۳	version	String	بله	نسخه کدینگ

- نمونه خروجی موفق

```
{
  "result": {
    "data": "\"ACETAMINOPHEN 100 mg / 1mL 15 mL SOLUTION / DROPS ORAL\"",
    "status": {
      "statusCode": 200
    }
  },
  "status": {
    "statusCode": 200,
    "message": "OK"
  }
}
```

- نمونه خروجی ناموفق

```
{
  "status": 0,
  "code": 401,
  "msg": "",
  "violations": []
}
```

خروجی این تابع رشته‌ای است که مقدار متناظر با کد و ترمینولوژی مربوطه را در نسخه موردنظر برمی‌گرداند.

تابع TerminologiesTermByVersion

از این تابع به منظور دریافت آرایه‌ای از کدها و مقادیر یک نسخه خاص از ترمینولوژی مشخص استفاده می‌شود.

- قالب ورودی

```
POST /api/client/apim/v1/maksaservice/gwmaksa/TerminologiesTermByVersion
HTTP/1.1
Host: apigateway.behdasht.gov.ir
pid: {Your Pid}
Content-Type: application/json
Authorization: Bearer {{token}}
{
  "terminologyId": "erX",
```



```
"version": "10000",
"pageNumber": 1,
"pageSize": 100
}
```

- نمونه ورودی

```
curl --location --
request POST 'https://apigateway.behdasht.gov.ir/api/client/apim/v1/maksaservice/g
wmaksa/TerminologiesTermByVersion' \
--header 'pid: {Your Pid}' \
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer {{token}}' \
--data-raw '{
  "terminologyId": "erX",
  "version": "10000",
  "pageNumber": 1,
  "pageSize": 100
}'
```

جدول ۱۳- پارامترهای ورودی Body تابع TerminologiesTermByVersion

ردیف	نام پارامتر	نوع داده	الزامی	توضیحات
۱	terminologyId	String	بله	عنوان ترمینولوژی
۲	version	String	بله	نسخه کدینگ
۳	pageNumber	Int	بله	شماره صفحه - پیش فرض ۱ می باشد.
۴	pageSize	Int	بله	اندازه صفحه - پیش فرض ۱ می باشد و حداکثر مقدار می تواند ۲۰۰ باشد.

- نمونه خروجی موفق

```
{
  "result": {
    "data": [
      {
        "valueField": "MOMETASONE FUROATE 0.1 % 15 g CREAM TOPICAL [MEGACOR
RT]",
        "terminology_idField": "eRx",
        "coded_stringField": "60281",
        "description": null,
        "categories": [],
        "parent": null
      }
    ],
    "status": {
      "statusCode": 200
    }
  },
  "status": {
    "statusCode": 200,
    "message": "OK"
  }
}
```



```
}
}
```

- نمونه خروجی ناموفق

```
{
  "result": {
    "status": {
      "statusCode": 404
    }
  },
  "status": {
    "statusCode": 200,
    "message": "OK"
  }
}
```

جدول ۱۴- پارامترهای خروجی تابع TerminologiesTermByVersion

ردیف	نام پارامتر	نوع داده	توضیحات
۱	valueField	String	مقدار
۲	terminology_idField	String	ترمینولوژی
۳	coded_stringField	String	کد

تابع TerminologyVersions

از این تابع به منظور دریافت آرایه‌ای از اطلاعات مربوط به نسخه‌های مختلف یک ترمینولوژی استفاده می‌شود.

- قالب ورودی

```
POST /api/client/apim/v1/maksaservice/gwmaksa/TerminologyVersions
HTTP/1.1
Host: apigateway.behdasht.gov.ir
pid: {Your Pid}
Content-Type: application/json
Authorization: Bearer {{token}}
{
  "terminologyId" : "ErX"
}
```

- نمونه ورودی

```
curl --location --
request POST 'https://apigateway.behdasht.gov.ir/api/client/apim/v1/maksaservice/gwmaksa/TerminologyVersions' \
--header 'pid: {Your Pid}' \
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer {{token}}' \
--data-raw '{
  "terminologyId" : "ErX"
}'
```



جدول ۱۵- پارامترهای ورودی Body تابع TerminologyVersions

ردیف	نام پارامتر	نوع داده	الزامی	توضیحات
۱	terminologyId	String	بله	ترمینولوژی

- نمونه خروجی موفق

```
{
  "result": {
    "data": [
      {
        "id": 147,
        "TerminologyId": "eRX",
        "Version": "10000",
        "VersionId": 86,
        "IsLast": true
      }
    ],
    "status": {
      "statusCode": 200
    }
  },
  "status": {
    "statusCode": 200,
    "message": "OK"
  }
}
```

- نمونه خروجی ناموفق

```
{
  "result": {
    "status": {
      "statusCode": 404
    }
  },
  "status": {
    "statusCode": 200,
    "message": "OK"
  }
}
```



جدول ۱۶- پارامترهای خروجی Body تابع TerminologyVersions

ردیف	نام پارامتر	نوع داده	توضیحات
۱	id	Integer	شناسه
۲	TerminologyId	String	عنوان ترمینولوژی
۳	Version	String	نسخه کدینگ
۴	VersionId	Integer	شناسه نسخه کدینگ
۵	IsLast	Boolean	آیا این مقدار آخرین نسخه است؟

تابع TerminologyCategories

از این تابع به منظور دریافت آرایه‌ای از دسته بندی های یک ترمینولوژی استفاده می‌شود.

- قالب ورودی

```
POST /api/client/apim/v1/maksaservice/gwmaksa/TerminologyCategories
HTTP/1.1
Host: apigateway.behdasht.gov.ir
pid: {Your Pid}
Content-Type: application/json
Authorization: Bearer {{token}}
{
  "terminologyId" : "ErX"
}
```

- نمونه ورودی

```
curl --location --
request POST 'https://apigateway.behdasht.gov.ir/api/client/apim/v1/maksaservice/gwmaksa/TerminologyCategories\
--header 'pid: {Your Pid}' \
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer {{token}}' \
--data-raw '{
  "terminologyId" : "IRC"
}'
```

جدول ۱۷- پارامترهای ورودی Body تابع TerminologyCategories

ردیف	نام پارامتر	نوع داده	الزامی	توضیحات
۱	terminologyId	String	بله	ترمینولوژی



- نمونه خروجی موفق

```
{
  "result": {
    "data": [
      "دارو",
      "تجهیزات"
    ],
    "status": {
      "statusCode": 200
    }
  },
  "status": {
    "statusCode": 200,
    "message": "OK"
  }
}
```

- نمونه خروجی ناموفق

```
{
  "result": {
    "status": {
      "statusCode": 404
    }
  },
  "status": {
    "statusCode": 200,
    "message": "OK"
  }
}
```



پیوست‌ها

پیوست ۱- ساختار خروجی فراخوانی توابع دیتاس

فراخوانی هر یک از توابع دیتاس (به استثناء توابع دریافت و بروزرسانی توکن دیتاس)، منجر به دریافت خروجی با قالب زیر می‌شود:

```
{
  "result": {
    "data": {obj},
    "status": {
      "statusCode": XXX
    }
  },
  "status": {
    "statusCode": XXX,
    "message": "message"
  }
}
```

ردیف	نام پارامتر	نوع داده	توضیحات
۱	result.data	Object	پارامتر(های) خروجی تابع فراخوانی شده (به توضیحات مربوط به هر تابع در متن سند مراجعه نمایید).
۲	result.status.statusCode	Integer	کد وضعیت پاسخ مربوط به سرویس‌دهنده
۳	status	Object	وضعیت پاسخ دیتاس نسبت به فراخوانی تابع، شامل کد وضعیت (statusCode) و پیام (message) مطابق پیوست ۲





پیوست ۲- کدهای وضعیت پاسخ دیتاس نسبت به فراخوانی توابع

در جدول زیر فهرست کدهای وضعیت پاسخ دیتاس نسبت به فراخوانی توابع، متن پیام و توضیحات مربوطه آمده است.

ردیف	کد وضعیت	پیام	توضیحات
۱	۲۰۰	OK!	عملیات با موفقیت انجام شد.
۲	۲۰۲	Accepted!	درخواست شما دریافت شد.
۳	۴۰۰	Bad request.	خطایی در داده‌های ورودی وجود دارد.
۴	۴۰۱	Credential is not valid!	نام کاربری و یا کلمه عبور نادرست است.
۵	۴۰۳	Your origin IP is permanently blocked!	آی پی شما قبلاً اعلام نشده و مجاز به اتصال نیست.
۶	۴۰۴	Not Found!	پاسخی برای درخواست مورد نظر یافت نشد.
۷	۴۱۰	All resources are moved permanently to the HTTP secure protocol(HTTPS)!	برای دسترسی به خدمات از HTTPS استفاده نمایید.
۸	۴۲۹	Too many requests or Access denied!	درخواست‌ها بیش از حد مجاز است / مجوز استفاده از سرویس صادر نشده است
۹	۴۹۹	Client Closed Request!	در هنگام پردازش، کاربر ارتباط را قطع کرده است.
۱۰	۵۰۰	Internal server error!	بروز خطا در انجام عملیات توسط سرور.
۱۱	۵۰۳	Service provider error!	بروز خطا در انجام عملیات توسط فراهم‌کننده سرویس.
۱۲	۵۰۴	Backend timeout!	سرویس‌دهنده پاسخ‌گو نیست.
۱۳	۵۲۰	Unknown error!	خطای نامشخص.