

Gerimedica Implementation of System for Cross-domain Identity Management (SCIM)

Abstract

This document details the Gerimedica implementation of 'System for Cross-domain Identity Management' or SCIM. The intended purpose of this document is to describe the details of our implementation including what aspects of the standards we support and how this system can be used. The intended audience for this document is developers who would like to use SCIM for managing Gerimedica users.

For further information or questions regarding the content of this document, please contact support@gerimedica.nl.

Version

Document Version	Release Date
1.0	27-02-2023
1.01	14-03-2023
1.02	24-03-2023
1.03	19-04-2023
1.04	20-06-2023
1.05	27-07-2023
1.06	09-08-2023
1.07	11-08-2023
1.08	23-01-2024
1.09	19-09-2024
1.10	02-04-2025
1.11	10-09-2025
1.12	16-09-2025

1.13	23-09-2025
1.14	24-09-2025
1.15	19-06-2026
1.16	29-07-2026

Changelog

Notable changes to this document will be tracked here.

Version 1.01

- We have updated the list of possible entitlements to now include Module Authorizations. As a result, users can now be granted access to this specific feature or functionality. Please note that this is a new addition and as such, the following values have been added to the list of possible entitlements:

-

```
"AdministratieveOndersteuning",
"Behandeldossier",
"FinancieelVoorGRZBasisGGZELVZPM",
"FinancieelVoorEerstelijnsParamedischeZorgHuisartsenzorgGZSP",
"Gebruikersbeheer",
"Inzicht",
"PortaalFinanciëleKoppeling",
"Zorgdossier"
```

- We have made updates to the request and response of the 'create' and 'update' endpoints, as well as the response of the 'get' function, in accordance with the YSIS extension schema. As a result, the format and structure of these endpoints have been modified to align with the schema. Please take note that only the response of the 'get' function has been changed, while the request and response of the 'create' and 'update' functions have been updated.

So the current usage is like this:

```
ez "urn:ietf:params:scim:schemas:extension:ysis:2.0:User": {
  "ysisInitials": "Y.S",
  "discipline": "MEDICAL",
  "agbCode": "agbCode1"
```

```
}
```

We are currently not supporting custom roles. As such, it is not possible to create or assign custom roles to users at this time.

We have removed the 'honorific prefix' field from the user schema. As a result, this field will no longer be displayed or required when creating or updating user information.

Version 1.02

We have added 'initials' as a new property to the YSIS User extension Schema. This value is required, but unlike 'ysisInitials' does not need to be unique across the organization.

```
"urn:ietf:params:scim:schemas:extension:ysis:2.0:User": {
  "ysisInitials": "Y.S",
  "discipline": "MEDICAL",
  "agbCode": "agbCode1",
  "initials": "YS"
}
```

Version 1.03

- We have added a new schema for User Enterprise.

This User Enterprise schema has an attribute employeeNumber.

```
{
  "id": "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
  "name": "Enterprise",
  "description": "Enterprise for User ResourceType.",
  "attributes": [
    {
      "name": "employeeNumber",
      "type": "string",
      "multiValued": false,
      "description": "The employer's number of the user. The maximum size for this field is 20 character.",
      "required": false,
      "caseExact": true,
      "mutability": "readOnly",
      "returned": "default",
      "uniqueness": "server"
    }
  ],
  "meta": {
    "resourceType": "Schema",

```

```

    "location":
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User"
  }
}

```

- We have added gender, export timeline events and infix to User Schema.

```

{
  "name": "gender",
  "type": "complex",
  "multiValued": false,
  "description": "An indication of the User's gender. If no one selected
it will be given as UNKNOWN by default.",
  "required": false,
  "canonicalValues": [
    "UNKNOWN",
    "MALE",
    "FEMALE"
  ],
  "mutability": "readWrite",
  "returned": "default",
  "uniqueness": "none"
},
{
  "name": "exportTimelineEvents",
  "type": "boolean",
  "multiValued": false,
  "description": "A Boolean value indicating whether to show the status
of the User's events on calendar or not.",
  "required": false,
  "caseExact": true,
  "mutability": "readWrite",
  "returned": "default",
  "uniqueness": "none"
}

```

- We have added **infix** as subAttribute to Name attribute in user schema

```
{
  "name": "name",
  "type": "complex",
  "multiValued": false,
  "description": "The components of the user's real name.",
  "required": false,
  "mutability": "readWrite",
  "returned": "default",
  "uniqueness": "none"
  "subAttributes": [
    {
      "name": "infix",
      "type": "string",
      "multiValued": false,
      "description": "It is a Tussenvoegsel. In a family name affix
positioned between a person's given name and the main part of their family
name. For Example for name: Emma van der Sluss, van der is infix.",
      "required": false,
      "caseExact": true,
      "mutability": "readWrite",
      "returned": "default",
      "uniqueness": "none"
    }
  ]
}
```

- We have added bignumber and position to **Ysis User Extension Schema**

```
{
  "name": "bigNumber",
  "type": "string",
  "multiValued": false,
  "description": "Bignumber of the User. The maximum size for this
field is 11 characters.",
}
```

```
    "required": false,  
    "caseExact": true,  
    "mutability": "readWrite",  
    "returned": "default",  
    "uniqueness": "server"  
  },  
  {  
    "name": "position",  
    "type": "string",  
    "multiValued": false,  
    "description": "The function of the user. The maximum size for this  
field is 255 character.",  
    "required": false,  
    "caseExact": true,  
    "mutability": "readWrite",  
    "returned": "default",  
    "uniqueness": "none"  
  }  
}
```

- To create user request new attributes added

```
"gender": null,
"exportTimelineEvents": true,
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User": {
  "employeeNumber": "2015010213058"
}
"name": {
  "infix": "van"
}
```

- To create user response updated

```
"gender": "UNKNOWN",
"exportTimelineEvents": true,
"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User": {
  "employeeNumber": "2015010213058"
}
"name": {
  "infix": "van"
}
```

- Update User Request updated

```
"gender": null,
"exportTimelineEvents": true
"name": {
  "infix": "van"
}
```

- Update User Response updated

```
"gender": null,
"exportTimelineEvents": true
"name": {
  "infix": "van"
}
```

- We also added this new schema for all request which are contain schemas json

```
"schemas": [
  "urn:ietf:params:scim:schemas:core:2.0:User",
  "urn:ietf:params:scim:schemas:extension:ysis:2.0:User",
  "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User"
]
```

Version 1.04

We have made an enhancement to the discipline field by updating the accepted values. In the previous version, we allowed descriptions of disciplines, but now we only accept enum values for the disciplines. To provide more clarity, here are the new enum values and their corresponding mappings:

Discipline Value	Description
MEDICAL	Medisch
PHYSIOTHERAPIST	Fysiotherapie
PSYCHOLOGIST	Psychologie
ERGOTHERAPIST	Ergotherapie
LOGOPEDIST	Logopedie
DIETITIAN	Diëtetiek
PASTOR	Geestelijke verzorging
SOCIAL_WORK	Maatschappelijk werk
MUSIC_THERAPIST	Muziektherapie
PMT	Psychomotorische therapie
DENTAL_CARE	Mondzorg
ACTIVITY_COACH	Activiteitenbegeleiding
READ_ONLY	Leesaccount
CARE	Zorg
NURSE	Verpleegkunde
ADMINISTRATIVE	Administratie
PSYCHOLOGIST_CONSULTANT	Psychologie consultatie

Please note that this change impacts the create, update, and get endpoints. If you need to verify the current values, you can refer to the schema's endpoint for the most up-to-date information.

- We have made another improvement regarding the field "roles." From now on, two-worded roles should be provided in regular case. This change applies to the create and update endpoints. Furthermore, the response from the get request will include these roles in the specified format. For your reference, here are the affected values and their updated formats resulting from this change:
 - GeestelijkeVerzorging -> Geestelijke verzorging
 - MaatschappelijkWerk -> Maatschappelijk werk
 - PsychomotorischeTherapie -> Psychomotorische therapie
 - PsychologieConsultatie -> Psychologie consultatie

To ensure you have the most accurate and current information, you can also check the schema's endpoint for the latest values.

Version 1.05

- We have introduced the ability to explicitly specify which Ysis Modules the user has access to. This includes adding a new non-mandatory 'modules' field to the `urn:ietf:params:scim:schemas:extension:ysis:2.0:User` schema. The valid set of possible values for this field are:
 - YSIS_CORE
 - YSIS_DBC
 - ECD_CARE
 - FINANCIAL
 - FINANCIAL_EXPORT_PORTAL
 - MANAGEMENT
 - USER_MANAGEMENT
- The following values were previously valid to pass in as authorization ids. These are no longer valid and passing them in will return a validation error. With the exception of 'Inzicht' which has been removed as it is not in use, these values have been migrated to modules:
 - AdministratieveOndersteuning
 - Behandeldossier
 - FinancieelVoorGRZBasisGGZELVZPM
 - FinancieelVoorEerstelijnsParamedischeZorgHuisartsenzorgGZSP
 - Gebruikersbeheer
 - Inzicht
 - PortaalFinanciëleKoppeling
 - Zorgdossier

Version 1.06

- We have updated the User (`urn:ietf:params:scim:schemas:extension:ysis:2.0:User`) schema 'isActive' field

to be mutable. This affects the SCIM User create and update endpoints which have been enhanced to process the value of this field and will internally update the 'blocked' status of the account. Setting the value to false will block the account, so it cannot be used until it is unblocked, Setting the active status to false does not delete the account. Blocked accounts can be unblocked and made active again by setting the 'active' field to true. **Please note, accounts which are deleted are not set to blocked, they are internally archived and cannot be unblocked or re-activated.**

- We have created a new '/users' endpoint, which can return sets of users. This endpoint is not intended to return all available users. The purpose of this endpoint is to determine if a non deleted User exists which matches a given filter. A filter must be provided for this endpoint. Full details of the supported filters can be found in the Filtering section. **The intended use for this functionality is to allow clients to verify that an account does or does not already exist for a given employeeNumber.**
- Filtering is now supported, and the OverviewSchema has been updated to reflect this. Please note, currently filtering is only supported on the new '/users' endpoint.

Version 1.07

We have added a new GET endpoint which can be used to return the available set of Roles for an organization. This will include both the default and custom roles. The ScimRole objects returned from this endpoint are the same object which is currently used in the existing SCIM User endpoints. Please see the SCIM Role Operations section for more information.

We have also modified the existing ScimRole object, we have added a displayName field.

Previous Definition:

Example Error:

```
ScimRole
{
  "value": "The value for this Role."
}
```

New Definition:

```
ScimRole
{
  "value": "The id for this Role.",
  "displayName": "The human readable name of this Role."
}
```

Version 1.08

This document has been updated with more error examples and some minor corrections after a review of the SCIM error messaging.

Version 1.09

The set of available modules has been updated:

MANAGEMENT has been removed.

ECD_ADMIN has been added.

The set of available Entitlements have been updated to reflect changes in the available Gerimedica Authorizations.

The following Entitlements have been added to the available options:

AccorderenWlz,
AccorderenWmo,
DeclarerenWlz,
DeclarerenWmo

Version 1.10

We have updated **retrieving users list with a filter** endpoint; the user schema filter on field EmployeeNumber accepts the PR operator too in addition to EQ.

Version 1.11

We have updated **retrieving users list with a filter** endpoint; the user schema filter accepts on field userName the EQ and PR operators.

Version 1.12

We have created the version v3 for our endpoints because any query resource endpoint will return an object like this:

```
{
  "schemas": [
    "urn:ietf:params:scim:api:messages:2.0:ListResponse"
  ],
}
```

```

"totalResults": 7,
"Resources": [
  ...
]
}

```

The affected endpoints are:

- List resource types
- List schemas
- List users
- List roles

Version 1.13

Password is no longer mandatory to create a user. If the password is not provided, there will be an auto generated password.

Version 1.14

Updating schema that wrongly mentioned the roles could be assigned automatically. Our SCIM provisioning does not allow that.

Version 1.15

Updating the authentication method for our acceptance environment migrating to Auth0. Production is scheduled to migrate and this document will have a new version.

Version 1.16

We have added the ability to re-activate a previously deleted (archived) User. Sending a 'create' request to the Users endpoint with a userName, ysisInitials and discipline that all match a single archived User will restore that User instead of creating a new one, preserving their original userId and applying any other changes in the request. If only some of these fields match an archived User, the request is rejected with a 409 Conflict rather than creating a duplicate or partially re-activating the wrong record. See the new '**Re-activating Users**' section for details.

Abstract	1
Version	1
Changelog	2
Version 1.01	2
Version 1.02	3
Version 1.03	3
Version 1.04	8
Version 1.05	9
Version 1.06	9
Version 1.07	10
Version 1.08	11
Version 1.09	11
Version 1.10	11
Version 1.11	11
Version 1.12	11
Version 1.13	12
Version 1.14	12
Version 1.15	12
Version 1.16	12
SCIM	16
What Is SCIM	16
Accessing SCIM	17
Authentication	18
Access Token	18
How To Get an Access Token	18
Example Request	18
Example Response	18
Example Error Response	19
How To Use Access Token	19
Ip Whitelist	19
Gerimedica SCIM Implementation	21
Service Provider Configuration Schema	21
Example Successful Response	21
Example Error Responses	22
Resource Types Schemas	24
Resource Types	24
Example Successful Response	24
Example Error Responses	25
Resource Type	26
Example Successful Response	26
Example Error Responses	26
Schema Representations	29
Schema Definitions	29

Example Successful Response	29
Example Error Responses	57
Schema Definition	59
Example Successful Response - User	59
Example Successful Response - User Enterprise Extension	68
Example Successful Response - Ysis User Extension	69
Example Successful Response - ResourceType	72
Example Successful Response - ServiceProviderConfig	75
Example Successful Response - Schema	81
Example Successful Response - Error	90
Example Error Responses	91
Scim User Operations	93
Creating Users	93
Example Request	93
Example Successful Response	94
Example Error Responses	96
Re-activating Users	98
Example Request	98
Example Successful Response	99
Example Error Responses	99
Retrieving Users (by userId)	101
Example Request	101
Example Successful Response	101
Example Error Responses	102
Retrieving Users (with Filter)	104
Example Request	104
Example Successful Response	104
Example Error Responses	105
Filtering	108
Supported Filter Structure:	108
Supported Schemalds and AttributeName:	108
Supported Operators:	108
Supported Values:	108
Updating Users	109
Example Request	109
Example Successful Response	110
Example Error Response	110
Deleting Users	114
Example Request	114
Example Successful Response	114
Example Error Responses	114
Scim Role Operations	116
Retrieving Roles	116
Example Request	116

Example Successful Response	116
Example Error Responses	117

SCIM

What Is SCIM

SCIM stands for System for Cross-domain Identity Management.

It is an open source protocol for automating the exchange of user identity information across domains. It provides a standardized way for different systems to share user profiles and groups and avoids the duplicated work of creating the same users in multiple systems manually. SCIM allows our customers to automate and streamline the management of Gerimedica users resulting in a more secure and efficient process.

Gerimedica implements Version 2.0 of SCIM.

Technical information can be found for the SCIM protocol here:

Overview: <https://www.rfc-editor.org/rfc/rfc7642>

Core-Schema: <https://www.rfc-editor.org/rfc/rfc7643>

Protocol: <https://www.rfc-editor.org/rfc/rfc7644>

Accessing SCIM

All SCIM relative endpoints presented in this document are relative to a base URL, which includes a subdomain that is unique per organization who is a customer of Gerimedica. Subdomains will be displayed as **{subdomain}**, customers should substitute this with your own unique subdomain value.

SCIM is available on both our Acceptance and Production environments. Relative SCIM endpoints will be presented in the rest of the document as relative to the **\$BASE_URL**.

The **\$BASE_URL** per environment:

Production: <https://{subdomain}.ysis.nl/gm/api/um/>

Acceptance: <https://{subdomain}.acceptatie2.ysis.nl/gm/api/um/>

Authentication is provided by Auth0 on our Acceptance and our custom Idp on Production environment. Our authentication endpoints will be presented relative to the **\$AUTH_URL**.

The **\$AUTH_URL** per environment:

Production: <https://{subdomain}.ysis.nl/cas/>

Acceptance: <https://auth.acceptatie2.ysis.nl/>

Note: Production will migrate to Auth0 after September 22nd 2026. The **\$AUTH_URL** will be:

Production: <https://auth.ysis.nl/>


```
}
```

Example Error Response

Error Scenario: The caller attempts to retrieve a token with an invalid client id or client secret.

Example Error:

```
HTTP/1.1 401 Unauthorized
{
  "error": "invalid_client",
  "error_description": "Bad client credentials"
}
```

Error Scenario: The caller attempts to retrieve a token with an invalid grant type.

Example Error:

```
HTTP/1.1 400 Bad Request
{
  "error": "unsupported_grant_type",
  "error_description": "Unsupported grant type"
}
```

How To Use Access Token

In the header of all requests the user needs to provide the bearer token.

```
Authorization: Bearer
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiJzY21tLWludGVncmF0aW9uIiwiaXNjbGInVt
cy1hcGkiXSwic2NvcGUiOiJic2NpbSdLCjleHAiOiJlY2NzY2MzA3OTYsImF1dGhvcml0aWVzIjpbIjJPTeVfQ
0xJRUSU1I0sImp0aSI6IjdoY2t0ZmVudWRGa2hsZnNtVERjU0g2eVQyayIsImNsaWVudF9pZCI6InNjaW0taW
50ZWdyYXRpb24ifQ.04lcNpuc965_z32h2CaD0YdLAbF_yvrEWP5jQT5Y0mafoCVHoojsipvoIY0ILn7sneZA
AdLmY2Gill-TWJP-9IsBx51bKBxuWRi96VOH4r5p8u60IfsQfHXxt8w7ImZnf_8aHdIX_86-Q2yy0Sd2QUOzv
meSFJoGHjIIwJYzJdzyvib-R_K2-W3KDNLXothoR6hHzt70CQsm2U6R0K_B9PoFPrLmim3F_oitSD4D3efIfc
7FS8nFB-08bn2xrgHH_E0Sh1CxXnGSgI994Cxd6emAaGyJ8HTuY0zJ28GWHCTz4Jes97-4mRe07PKNHn-W9e
1LtHSe440jSNxHu25031W_TVvFvpKtclErhGu0ycvpz3fJ7t3SwsJ0dswvtRFgh2eDQpiTRT-aGVXPTedX4XP
AjMyQ4Vj1iUW3_GPtq1g4zcEbI14AqZK921ycNGiz-oDt3g0RL115hF_in86X3bL-71TDkAXECFeK1ezb9a7Z
nR2--ornRou1Rg1wtEvYGFaZSTgeDR5fUURZwtVeJO9b4BQPJ1P2hIwZDvwcG2au-XquCPa7-mcI2mfsh15Z_
Be5UvwQcCIuoPGX1CSHeogeDqGGwsBD2PN03hG0c3cyii6FcSvPWkfPJWZIHlgJEjNoRlqPZReTf85Aq0vP74
sdAOidRnqbrcJHpvjtw
```

IP Whitelist

To help protect access to our access token endpoint, Gerimedica uses IP whitelisting. This means that we only allow requests to the access token endpoint from specific IP addresses.

If you are a customer and need to call the access token endpoint, you will need to provide Gerimedica with a list of your public IP addresses. We will then add those IP addresses to our whitelist, so that your requests can be authenticated.

To provide us with your list of public IP addresses, please contact our support team at support@gerimedica.nl. Once we have received your list, we will add your IP addresses to our whitelist as soon as possible.

Please note that if your IP address changes or if you need to use a different IP address to make requests to the access token endpoint, you will need to notify us so that we can update our whitelist accordingly.

Gerimedica SCIM Implementation

Service Provider Configuration Schema

Endpoint: **\$BASE_URL/scim/v3/service-provider-config**
HTTP Operation: **GET**

This endpoint will return a JSON object which describes the SCIM features supported by Gerimedica. The returned object will match the schema

urn:ietf:params:scim:schemas:core:2.0:ServiceProviderConfig.

This endpoint is read-only.

Filtering, sorting, and pagination are not supported.

Example Successful Response

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "schemas": ["urn:ietf:params:scim:schemas:core:2.0:ServiceProviderConfig"],
  "documentationUri": "TODO fill in URL to this document",
  "patch": {
    "supported": false
  },
  "bulk": {
    "supported": false
  },
  "filter": {
    "supported": true
  },
  "changePassword": {
    "supported": false
  },
  "sort": {
    "supported": false
  },
  "etag": {
    "supported": true
  },
  "authenticationSchemes": [
    {
      "name": "OAuth Bearer Token",
      "description":
        "Authentication scheme using the OAuth Bearer Token Standard",
      "specUri": "http://www.rfc-editor.org/info/rfc6750",
      "documentationUri": "http://example.com/help/oauth.html",
      "type": "oauthbearertoken",
      "primary": true
    }
  ]
}
```

```

    }
  ],
  "meta": {
    "location": "$BASE_URL/scim/v3/service-provider-config",
    "resourceType": "ServiceProviderConfig",
    "created": "2023-01-01T04:56:22Z",
    "lastModified": "2023-02-13T04:42:34Z",
    "version": "w/\037fbfb28473d1590ba2211690684a3de\"
  }
}

```

Example Error Responses

Error Scenario: A request to this endpoint which contains filtering options will result in an error.

Example Error:

```

HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "scimType": "invalidFilter",
  "detail": "Filtering is not supported."
}

```

Error Scenario: A request to this endpoint is made which does not contain the correct authentication.

Example Error:

```

HTTP/1.1 403 UNAUTHORIZED
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "detail": "Operation is not permitted based on the supplied authorization.",
  "status": 401
}

```

Error Scenario: A request is made to this endpoint from an organization with multiple groups.

Example Error:

```

HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations with multiple groups."
}

```

Error Scenario: A request is made to this endpoint from an organization with no group.

Example Error:

```

HTTP/1.1 400 BAD REQUEST
{

```

```
"schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],  
"status": 400,  
"detail": "SCIM is not currently supported for organizations without groups."  
}
```

Resource Types Schemas

Resource Types

Endpoint: **\$BASE_URL/scim/v3/resource-types**

HTTP Operation: **GET**

A HTTP GET call to this endpoint can be used to return a list of Resource Types which are supported by Gerimedita. Gerimedita currently supports the 'User' Resource Type and does not support the 'Group' or 'Me' Resource Types.

The data returned by this endpoint will be a JSON array of Resource Type objects matching the **urn:ietf:params:scim:schemas:core:2.0:ResourceType** schema.

This endpoint is read only.

Filtering, sorting and pagination are not supported.

Example Successful Response

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "schemas": [
    "urn:ietf:params:scim:api:messages:2.0:ListResponse"
  ],
  "totalResults": 1,
  "Resources": [
    {
      "schemas": ["urn:ietf:params:scim:schemas:core:2.0:ResourceType"],
      "id": "user",
      "name": "User",
      "endpoint": "/user",
      "description": "User Account",
      "schema": "urn:ietf:params:scim:schemas:core:2.0:User",
      "schemaExtensions": [
        {
          "schema": "urn:ietf:params:scim:schemas:extension:ysis:2.0:User",
          "required": true
        },
        {
          "schema": "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
          "required": false
        }
      ]
    },
    {
      "meta": {
        "location": "$BASE_URL/scim/v3/resource-types/user",
        "resourceType": "ResourceType"
      }
    }
  ]
}
```

```
]
```

Example Error Responses

Error Scenario: A request to this endpoint which contains filtering options will result in an error.

Example Error:

```
HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": "403",
  "scimType": "invalidFilter",
  "detail": "Filtering is not supported."
}
```

Error Scenario: A request to this endpoint is made for an invalid organization.

Example Error:

```
HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "detail": "Operation is not permitted based on the supplied authorization."
}
```

Error Scenario: A request is made to this endpoint from an organization with multiple groups.

Example Error:

```
HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations with multiple groups."
}
```

Error Scenario: A request is made to this endpoint from an organization with no group.

Example Error:

```
HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations without groups."
}
```

Resource Type

Endpoint: **\$BASE_URL/resource-types/{resourceType}**
 HTTP Operation: **GET**

A HTTP GET call to this endpoint will return details of a single Resource Type which is supported by Gerimedica. Passing an invalid value for the resourceType parameter will result in an error.

The data returned by this endpoint will be a single JSON object matching the **urn:ietf:params:scim:schemas:core:2.0:ResourceType** schema.

This endpoint is read only.

Filtering, sorting and pagination are not supported.

The following values are valid for the path parameter **resourceType**:

- **User**

Example Successful Response

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "schemas": ["urn:ietf:params:scim:schemas:core:2.0:ResourceType"],
  "id": "user",
  "name": "User",
  "endpoint": "/user",
  "description": "User Account",
  "schema": "urn:ietf:params:scim:schemas:core:2.0:User",
  "schemaExtensions": [
    {
      "schema": "urn:ietf:params:scim:schemas:extension:ysis:2.0:User",
      "required": true
    },
    {
      "schema": "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
      "required": false
    }
  ],
  "meta": {
    "location": "$BASE_URL/scim/v3/resource-types/user",
    "resourceType": "ResourceType"
  }
}
```

Example Error Responses

Error Scenario: A request to this endpoint which contains filtering options will result in an error.

Example Error:

```
HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "scimType": "invalidFilter",
  "detail": "Filtering is not supported."
}
```

Error Scenario: A request to this endpoint which contains an invalid resourceType.

Example Error:

```
HTTP/1.1 400 BAD_REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "scimType": "invalidValue",
  "detail": "The path parameter 'departments' for 'resourceType' is not supported."
}
```

Error Scenario: A request to this endpoint is made for an invalid organization.

Example Error:

```
HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "detail": "Operation is not permitted based on the supplied authorization."
}
```

Error Scenario: A request is made to this endpoint from an organization with multiple groups.

Example Error:

```
HTTP/1.1 400 BAD_REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations with multiple groups."
}
```

Error Scenario: A request is made to this endpoint from an organization with no group.

Example Error:

```
HTTP/1.1 400 BAD_REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations without groups."
}
```


Schema Representations

Schema Definitions

Endpoint: **\$BASE_URL/scim/v3/schemas**

HTTP Operation: **GET**

A HTTP GET to this endpoint can be used to return a complete list of the SCIM Schemas supported by Gerimedica.

The data returned by this endpoint will be in a JSON array format of objects matching the **urn:ietf:params:scim:schemas:core:2.0:Schema** schema.

This endpoint is read only.

Filtering, sorting and pagination are not supported.

Example Successful Response

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "schemas": [
    "urn:ietf:params:scim:api:messages:2.0:ListResponse"
  ],
  "totalResults": 1,
  "Resources": [
    {
      "id": "urn:ietf:params:scim:schemas:core:2.0:User",
      "name": "User",
      "description": "User Account",
      "attributes": [
        {
          "name": "userName",
          "type": "string",
          "multiValued": false,
          "description": "Unique identifier for the User, typically
            used by the user to directly authenticate to the service provider.
            Each User MUST include a non-empty userName value. This identifier
            MUST be unique across the service provider's entire set of Users.
            The maximum size for this field is 40 characters.",
          "required": true,
          "caseExact": false,
          "mutability": "readWrite",
          "returned": "default",
          "uniqueness": "server"
        }
      ],
    }
  ],
}
```

```

{
  "name": "name",
  "type": "complex",
  "multiValued": false,
  "description": "The components of the user's real name.",
  "required": false,
  "mutability": "readWrite",
  "returned": "default",
  "uniqueness": "none"
  "subAttributes": [
    {
      "name": "familyName",
      "type": "string",
      "multiValued": false,
      "description": "The family name of the User, or
last name in most Western languages (e.g., 'Jensen' given the full
name 'Ms. Barbara Jensen'). The maximum size for this field is 100
characters.",
      "required": true,
      "caseExact": true,
      "mutability": "readWrite",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "givenName",
      "type": "string",
      "multiValued": false,
      "description": "The given name of the User, or
first name in most Western languages (e.g., 'Barbara' given the
full name 'Ms. Barbara Jensen'). The maximum size for this field is 80
characters.",
      "required": true,
      "caseExact": true,
      "mutability": "readWrite",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "infix",
      "type": "string",
      "multiValued": false,
      "description": "It is a Tussenvoegsel in Dutch. In a family name
affix positioned between a person's given name and the main part of their
family name. For Example for name: Emma van der Sluss, van der is infix.",
    },
  ],
}

```

```

        "required": false,
        "caseExact": false,
        "mutability": "readWrite",
        "returned": "default",
        "uniqueness": "none"
    }
]
},
{
    "name": "active",
    "type": "boolean",
    "multiValued": false,
    "description": "A Boolean value indicating the User's administrative
status.",
    "required": false,
    "mutability": "readWrite",
    "returned": "default"
},
{
    "name": "password",
    "type": "string",
    "multiValued": false,
    "description": "The User's cleartext password. This
attribute is intended to be used as a means to specify an initial
password when creating a new User. A password is OPTIONAL when creating a
user and must not be present otherwise. If the password is not present at
creation, it will be auto generated. A password must be a minimum of 12
characters and a maximum of 64 characters. All characters are valid except
'/' and '\\'.",
    "required": false,
    "caseExact": true,
    "mutability": "writeOnly",
    "returned": "never",
    "uniqueness": "none"
},
{
    "name": "emails",
    "type": "complex",
    "multiValued": true,
    "description": "Email addresses for the user. Gerimedica can only
store one email address per user. Ideally one email address will be marked as
'primary' and if so, this will be saved. If no email address is marked as
primary and more than one email address is supplied, only one will be saved.
The email address saved will prioritize the 'work' over the 'home' and
'other' email addresses. If more than one of a given type is supplied, the
first email address with the highest priority based upon its type will be

```

```

accepted.",
  "required": false,
  "mutability": "readWrite",
  "returned": "default",
  "uniqueness": "none",
  "subAttributes" : [
    {
      "name": "value",
      "type": "string",
      "multiValued": false,
      "description": "The email address value of the user. The maximum
size for this field is 255 characters",
      "required": false,
      "caseExact": false,
      "mutability": "readWrite",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "primary",
      "type": "boolean",
      "multiValued": false,
      "description": "A Boolean value indicating the 'primary'
or preferred email address of this user. The primary attribute value 'true'
MUST appear no more than once.",
      "required": false,
      "mutability": "readWrite",
      "returned": "never"
    }
  ]
},
{
  "name": "phoneNumbers",
  "type": "complex",
  "multiValued": true,
  "description": "Phone numbers for the User. Gerimedica can save both
a mobile phone number and another phone number. Canonical type values of
'work' and 'mobile' will be accepted. If more than one value of a given type
is supplied, one should be marked with 'primary' and will be prioritized. If
more than one value of a given type is supplied without being marked as
'primary', the first will be accepted. If more than one phone number of a
given type is supplied and more than one is marked as 'primary', the first
supplied will be accepted.",
  "required": false,
  "mutability": "readWrite",
  "returned": "default",

```

```

"subAttributes": [
  {
    "name": "value",
    "type": "string",
    "multiValued": false,
    "description": "Phone number of the User.",
    "required": false,
    "caseExact": false,
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "type",
    "type": "string",
    "multiValued": false,
    "description": "A label indicating the attribute's function,
e.g., 'work', or 'mobile'.",
    "required": false,
    "caseExact": false,
    "canonicalValues": [
      "work",
      "mobile"
    ],
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "primary",
    "type": "boolean",
    "multiValued": false,
    "description": "A Boolean value indicating the 'primary'
or preferred phone number. The primary attribute value 'true' MUST appear no
more than once per type.",
    "required": false,
    "mutability": "readWrite",
    "returned": "never"
  }
]
},
{
  "name": "gender",
  "type": "complex",
  "multiValued": false,
  "description": "An indication of the User's gender. If no one selected

```

```

it will be given as UNKNOWN by default.",
  "required": false,
  "canonicalValues": [
    "UNKNOWN",
    "MALE",
    "FEMALE"
  ],
  "mutability": "readWrite",
  "returned": "default",
  "uniqueness": "none"
},
{
  "name": "exportTimelineEvents",
  "type": "boolean",
  "multiValued": false,
  "description": "A Boolean value indicating whether to show the status
of the User's events on calendar or not.",
  "required": false,
  "caseExact": false,
  "mutability": "readWrite",
  "returned": "default",
  "uniqueness": "none"
},
{
  "name": "roles",
  "type": "complex",
  "multiValued": true,
  "description": "A list of roles for the User that collectively
represent what activities the User will be undertaking while using this
system. A role is used to identify a set of authorizations the user is
granted within the system.",
  "required": false,
  "mutability": "readWrite",
  "returned": "default",
  "subAttributes": [
    {
      "name": "value",
      "type": "string",
      "multiValued": false,
      "description": "The unique identifier for the role",
      "required": false,
      "caseExact": true,
      "mutability": "readWrite",
      "returned": "default",
      "uniqueness": "none"
    }
  ],
},

```

```

    {
      "name": "displayName",
      "type": "string",
      "multiValued": false,
      "description": "The name of the role.",
      "required": false,
      "caseExact": true,
      "mutability": "readWrite",
      "returned": "default",
      "uniqueness": "none"
    }
  ]
},
{
  "name": "entitlements",
  "type": "complex",
  "multiValued": true,
  "description": "A list of authorizations that the user will be able
to undertake within the system. Valid entitlement values must be one of the
canonical values.",
  "required": false,
  "mutability": "readWrite",
  "returned": "default",
  "subAttributes": [
    {
      "name": "value",
      "type": "string",
      "multiValued": false,
      "description": "The value of an entitlement.",
      "required": false,
      "caseExact": false,
      "canonicalValues": [
        "AccorderenGRZBasisGGZELV",
        "AccorderenWlz",
        "AccorderenWmo",
        "BehandelplanAccorderen",
        "BehandelrelatieWijzigen",
        "BeheerAdviessjablonen",
        "BeheerBriefsjablonen",
        "BeheerMedewerkers",
        "BeheerOverig",
        "BeheerTijdregistratie",
        "BeheerUitvalbak",
        "DBCZorgactiviteitenRegistreren",
        "DBCZorgtrajectenAanmakenBewerkenVerwijderen",
        "DeclarerenGRZBasiGGZELVZPM",

```

```

"DeclarerenEerstelijnsParamedischeZorgHuisartsenzorGZSP",
"DeclarerenWlz",
"DeclarerenWmo",
"DocumentenBeherenVoorAlleDisciplines",
"DocumentenBeherenVoorEigenDiscipline",
"DocumentenLezenVoorEigenDiscipline",
"ELVTijdRegistreren",
"HADeclaratie",
"HAVerrichtingenWijzigen",
"JuridischeInformatieBewerken",
"LaboratoriumresultatenLezen",
"MedischTabblad",
"MeetinstrumentenInvullenBekijken",
"MetingenVerwijderenNa24Uur",
"MultidisciplinairTabblad",
"PatiëntenMutereren",
"RapportageGenereren",
"RapportagesVerwijderenNa24Uur",
"ToegangBonstat",
"ToegangFarmedRx",
"ToegangFierit",
"ToegangHINQPortaal",
"ToegangMedimo",
"ToegangNCare",
"ToegangQuestManagerROMPortaal",
"ToegangRoosterplatform",
"ToegangVIPLive",
"ToegangViPharma",
"ToegangZorgDomein",
"ToevoegenBewerkenVerwijderenFinanciering",
"ToevoegenBewerkenVerwijderenVerzekeringsgegevens",
"VerslagenLezenActiviteitenbegeleiding",
"VerslagenLezenDiëtetiek",
"VerslagenLezenErgotherapie",
"VerslagenLezenFysiotherapie",
"VerslagenLezenGeestelijkeVerzorging",
"VerslagenLezenLogopedie",
"VerslagenLezenMaatschappelijkWerk",
"VerslagenLezenMedischeDiscipline",
"VerslagenLezenMondzorg",
"VerslagenLezenMuziektherapie",
"VerslagenLezenPsychologie",
"VerslagenLezenPsychologieConsultatie",
"VerslagenLezenPsychomotorischeTherapie",
"VerslagenLezenVerpleegkunde",
"VerslagenLezenZorg",

```

```

        "VerslagenToevoegen",
        "VertrouwelijkVerslagLezenActiviteitenbegeleiding",
        "VertrouwelijkVerslagLezenDiëtetiek",
        "VertrouwelijkVerslagLezenErgotherapie",
        "VertrouwelijkVerslagLezenFysiotherapie",
        "VertrouwelijkVerslagLezenGeestelijkeVerzorging",
        "VertrouwelijkVerslagLezenLogopedie",
        "VertrouwelijkVerslagLezenMaatschappelijkWerk",
        "VertrouwelijkVerslagLezenMedischeDiscipline",
        "VertrouwelijkVerslagLezenMondzorg",
        "VertrouwelijkVerslagLezenMuziektherapie",
        "VertrouwelijkVerslagLezenPsychologie",
        "VertrouwelijkVerslagLezenPsychologieConsultatie",
        "VertrouwelijkVerslagLezenPsychomotorischeTherapie",
        "VertrouwelijkVerslagLezenVerpleegkunde",
        "VerwerkenExterneVerwijzingen",
        "VerwijderenVragenlijsten",
        "VerwijzenNaarAndereAisciplines",
        "VolledigDossierPrinten",
        "ZorgplanAccorderen"
    ],
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "none"
}
]
}
],
"meta" : {
    "resourceType": "Schema",
    "location":
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:schemas:core:2.0:User"
}
},
{
    "id": "urn:ietf:params:scim:schemas:extension:ysis:2.0:User",
    "name": "User",
    "description": "Ysis extension for User ResourceType",
    "attributes": [
        {
            "name": "discipline",
            "type": "string",
            "multiValued": false,
            "description": "Identifier to indicate the Ysis Discipline of this
User. Each User MUST include a non-empty discipline value, which must belong
to the set of canonical values.",

```

```

    "required": true,
    "caseExact": false,
    "canonicalValues": [
        "ACTIVITY_COACH",
        "ADMINISTRATIVE",
        "CARE",
        "DENTAL_CARE",
        "DIETITIAN",
        "ERGOTHERAPIST",
        "LOGOPEDIST",
        "MEDICAL",
        "MUSIC_THERAPIST",
        "NURSE",
        "PASTOR",
        "PHYSIOTHERAPIST",
        "PMT",
        "PSYCHOLOGIST",
        "PSYCHOLOGIST_CONSULTANT",
        "READ_ONLY",
        "SOCIAL_WORK"
    ],
    "mutability": "immutable",
    "returned": "default",
    "uniqueness": "none"
},
{
    "name": "ysisInitials",
    "type": "string",
    "multiValued": false,
    "description": "Ysis Initials are used to uniquely identify a user
within the Ysis system. Each User MUST have a unique non-empty Ysis initials
value. The value should consist of alphanumeric characters. the maximum size
for this field is 10 characters",
    "required": true,
    "caseExact": true,
    "mutability": "immutable",
    "returned": "default",
    "uniqueness": "server"
},
{
    "name": "initials",
    "type": "string",
    "multiValued": false,
    "description": "Initials are used to identify a user within the Ysis
system. Each User MUST have a unique non-empty value. The value should
consist of alphanumeric characters. the maximum size for this field is 10
characters",
    "required": true,

```

```

    "caseExact": true,
    "mutability": "immutable",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "agbCode",
    "type": "string",
    "multiValued": false,
    "description": "AGB Code of this User. AGB Codes should be numeric
and 8 characters in length.",
    "required": false,
    "caseExact": true,
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "bigNumber",
    "type": "string",
    "multiValued": false,
    "description": "Bignumber of the User. The maximum size for this
field is 11 characters.",
    "required": false,
    "caseExact": true,
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "server"
  },
  {
    "name": "position",
    "type": "string",
    "multiValued": false,
    "description": "The function of the user. The maximum size for this
field is 255 character.",
    "required": false,
    "caseExact": true,
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "modules",
    "type": "string",
    "multiValued": true,
    "description": "Set of identifiers representing the YSIS Modules to be

```

```

enabled for this User. The set of modules can be empty.",
    "required": false,
    "caseExact": true,
    "canonicalValues": [
        "ECD_ADMIN",
        "ECD_CARE",
        "FINANCIAL",
        "FINANCIAL_EXPORT_PORTAL",
        "USER_MANAGEMENT",
        "YSIS_CORE",
        "YSIS_DBC"
    ],
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "none"
},
{
    "meta" : {
        "resourceType": "Schema",
        "location":
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:schemas:extension:ysis:2.0:User"
    }
},
{
    "id": "urn:ietf:params:scim:schemas:core:2.0:ResourceType",
    "name": "ResourceType",
    "description": "Describes a SCIM Resource Type.",
    "attributes": [
        {
            "name": "id",
            "type": "string",
            "multiValued": false,
            "description": "The resource type's server unique id.",
            "required": false,
            "caseExact": false,
            "mutability": "readOnly",
            "returned": "default",
            "uniqueness": "server"
        },
        {
            "name": "name",
            "type": "string",
            "multiValued": false,
            "description": "The resource type's name. e.g., 'user'.",
            "required": true,

```

```

    "caseExact": false,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "description",
    "type": "string",
    "multiValued": false,
    "description": "The resource type's human-readable description.",
    "required": false,
    "caseExact": false,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "endpoint",
    "type": "reference",
    "referenceTypes": ["uri"],
    "multiValued": false,
    "description": "The resource type's HTTP-addressable endpoint
relative to the Base URL, e.g., '/users'.",
    "required": true,
    "caseExact": false,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "schema",
    "type": "reference",
    "referenceTypes": ["uri"],
    "multiValued": false,
    "description": "The resource type's primary/base schema URI.",
    "required": true,
    "caseExact": true,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "schemaExtensions",
    "type": "complex",
    "multiValued": false,
    "description": "A list of URIs of the resource type's schema

```

```

extensions.",
  "required": true,
  "mutability": "readOnly",
  "returned": "default",
  "subAttributes" : [
    {
      "name": "schema",
      "type": "reference",
      "referenceTypes": ["uri"],
      "multiValued": false,
      "description": "The URI of a schema extension.",
      "required": true,
      "caseExact": true,
      "mutability": "readOnly",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "required",
      "type": "boolean",
      "multiValued": false,
      "description": "A Boolean value that specifies whether
        or not the schema extension is required for the
        resource type. If true, a resource of this type MUST
        include this schema extension and also include any
        attributes declared as required in this schema extension.
        If false, a resource of this type MAY omit this schema
        extension.",
      "required": true,
      "mutability": "readOnly",
      "returned": "default"
    }
  ]
},
{
  "meta" : {
    "resourceType": "Schema",
    "location" :
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:schemas:core:2.0:ResourceType
"
  }
},
{
  "id": "urn:ietf:params:scim:schemas:core:2.0:ServiceProviderConfig",
  "name": "ServiceProviderConfig",
  "description": "Schema representing the service provider's

```

```

configuration.",
  "attributes" : [
    {
      "name": "documentationUri",
      "type": "reference",
      "referenceTypes" : ["external"],
      "multiValued": false,
      "description": "An HTTP-addressable URL pointing to the
        service provider's human-consumable help documentation.",
      "required": false,
      "caseExact": false,
      "mutability": "readOnly",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "patch",
      "type": "complex",
      "multiValued": false,
      "description": "A complex type that specifies PATCH configuration
options.",
      "required": true,
      "mutability": "readOnly",
      "returned": "default",
      "subAttributes" : [
        {
          "name": "supported",
          "type": "boolean",
          "multiValued": false,
          "description": "A Boolean value specifying whether or not
            the operation is supported.",
          "required": true,
          "mutability": "readOnly",
          "returned": "default"
        }
      ]
    },
    {
      "name": "bulk",
      "type": "complex",
      "multiValued": false,
      "description": "A complex type that specifies bulk configuration
options.",
      "required": true,
      "mutability": "readOnly",
      "returned": "default",

```

```

"subAttributes" : [
  {
    "name": "supported",
    "type": "boolean",
    "multiValued": false,
    "description": "A Boolean value specifying whether or not
      the operation is supported.",
    "required": true,
    "mutability": "readOnly",
    "returned": "default"
  },
  {
    "name": "maxOperations",
    "type": "integer",
    "multiValued": false,
    "description": "An integer value specifying the maximum
      number of operations.",
    "required": true,
    "mutability": "readOnly",
    "uniqueness": "none",
    "returned": "default"
  },
  {
    "name": "maxPayloadSize",
    "type": "integer",
    "multiValued": false,
    "description": "An integer value specifying the maximum
      payload size in bytes.",
    "required": true,
    "mutability": "readOnly",
    "uniqueness": "none",
    "returned": "default",      }
]
},
{
  "name": "filter",
  "type": "complex",
  "multiValued": false,
  "description": "A complex type that specifies FILTER options.",
  "required": true,
  "mutability": "readOnly",
  "returned": "default",
  "subAttributes": [
    {
      "name": "supported",
      "type": "boolean",

```

```

        "multiValued": false,
        "description": "A Boolean value specifying whether or not
            the operation is supported.",
        "required": true,
        "mutability": "readOnly",
        "returned": "default"
    },
    {
        "name": "maxResults",
        "type": "integer",
        "multiValued": false,
        "description": "An integer value specifying the maximum
            number of resources returned in a response.",
        "required": true,
        "mutability": "readOnly",
        "uniqueness": "none",
        "returned": "default"
    }
]
},
{
    "name": "changePassword",
    "type": "complex",
    "multiValued": false,
    "description": "A complex type that specifies configuration
        options related to changing a password.",
    "required": true,
    "mutability": "readOnly",
    "returned": "default",
    "subAttributes": [
        {
            "name": "supported",
            "type": "boolean",
            "multiValued": false,
            "description": "A Boolean value specifying whether or not
                the operation is supported.",
            "required": true,
            "mutability": "readOnly",
            "returned": "default"
        }
    ]
},
{
    "name": "sort",
    "type": "complex",
    "multiValued": false,

```

```

    "description": "A complex type that specifies sort result options.",
    "required": true,
    "mutability": "readOnly",
    "returned": "default",
    "subAttributes": [
      {
        "name": "supported",
        "type": "boolean",
        "multiValued": false,
        "description": "A Boolean value specifying whether or not
          the operation is supported.",
        "required": true,
        "mutability": "readOnly",
        "returned": "default"
      }
    ]
  },
  {
    "name": "authenticationSchemes",
    "type": "complex",
    "multiValued": true,
    "description": "A complex type that specifies supported
      authentication scheme properties.",
    "required": true,
    "mutability": "readOnly",
    "returned": "default",
    "subAttributes": [
      {
        "name": "name",
        "type": "string",
        "multiValued": false,
        "description": "The common authentication scheme name, e.g., HTTP
Basic.",
        "required": true,
        "caseExact": false,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
      },
      {
        "name": "description",
        "type": "string",
        "multiValued": false,
        "description": "A description of the authentication scheme.",
        "required": true,
        "caseExact": false,

```

```

        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "specUri",
        "type": "reference",
        "referenceTypes": ["external"],
        "multiValued": false,
        "description": "An HTTP-addressable URL pointing to the
            authentication scheme's specification.",
        "required": false,
        "caseExact": false,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "documentationUri",
        "type": "reference",
        "referenceTypes": ["external"],
        "multiValued": false,
        "description": "An HTTP-addressable URL pointing to the
            authentication scheme's usage documentation.",
        "required": false,
        "caseExact": false,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    }
]
},
"meta" : {
    "resourceType": "Schema",
    "location" :
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:schemas:core:2.0:ServiceProv
iderConfig"
}
},
{
    "id": "urn:ietf:params:scim:schemas:core:2.0:Schema",
    "name": "Schema",
    "description": "Specifies the schema that describes a SCIM schema.",
    "attributes": [
        {

```

```

    "name": "id",
    "type": "string",
    "multiValued": false,
    "description": "The unique URI of the schema.",
    "required": true,
    "caseExact": true,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "name",
    "type": "string",
    "multiValued": false,
    "description": "The schema's human-readable name. e.g., 'user'.",
    "required": true,
    "caseExact": false,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "description",
    "type": "string",
    "multiValued": false,
    "description": "A human-readable description of the schema.",
    "required": false,
    "caseExact": false,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "attributes",
    "type": "complex",
    "multiValued": true,
    "description": "A complex attribute that includes the attributes of a
schema.",
    "required": true,
    "mutability": "readOnly",
    "returned": "default",
    "subAttributes": [
      {
        "name": "name",
        "type": "string",
        "multiValued": false,

```

```

    "description": "The attribute's name.",
    "required": true,
    "caseExact": true,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "type",
    "type": "string",
    "multiValued": false,
    "description": "The attribute's data type.
      Valid values include 'string', 'complex', 'boolean',
      'decimal', 'integer', 'dateTime', 'reference'.",
    "required": true,
    "caseExact": false,
    "canonicalValues": [
      "string",
      "complex",
      "boolean",
      "decimal",
      "integer",
      "dateTime",
      "reference"
    ],
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "multiValued",
    "type": "boolean",
    "multiValued": false,
    "description": "A Boolean value indicating an attribute's
plurality.",
    "required": true,
    "mutability": "readOnly",
    "returned": "default"
  },
  {
    "name": "description",
    "type": "string",
    "multiValued": false,
    "description": "A human-readable description of the attribute.",
    "required": false,
    "caseExact": true,

```

```

        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "required",
        "type": "boolean",
        "multiValued": false,
        "description": "A Boolean value indicating whether or not the
attribute is required.",
        "required": false,
        "mutability": "readOnly",
        "returned": "default"
    },
    {
        "name": "canonicalValues",
        "type": "string",
        "multiValued": true,
        "description": "A collection of canonical values.",
        "required": false,
        "caseExact": true,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "caseExact",
        "type": "boolean",
        "multiValued": false,
        "description": "A Boolean value indicating whether or
not a string attribute is case sensitive.",
        "required": false,
        "mutability": "readOnly",
        "returned": "default"
    },
    {
        "name": "mutability",
        "type": "string",
        "multiValued": false,
        "description": "Indicates whether or not an attribute is
modifiable.",
        "required": false,
        "caseExact": true,
        "canonicalValues": [
            "readOnly",
            "readWrite",

```

```

        "immutable",
        "writeOnly"
    ],
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
},
{
    "name": "returned",
    "type": "string",
    "multiValued": false,
    "description": "Indicates when an attribute is returned
        in a response (e.g., to a query).",
    "required": false,
    "caseExact": true,
    "canonicalValues": [
        "always",
        "default",
        "never",
        "request"
    ],
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
},
{
    "name": "uniqueness",
    "type": "string",
    "multiValued": false,
    "description": "Indicates how unique a value must be.",
    "required": false,
    "caseExact": true,
    "canonicalValues": [
        "server",
        "global",
        "none",
    ],
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
},
{
    "name": "referenceTypes",
    "type": "string",
    "multiValued": true,
    "description": "Used only with an attribute of type

```

```

        'reference'. Specifies a SCIM resourceType that a
        reference attribute MAY refer to, e.g., 'User'.",
        "required": false,
        "caseExact": true,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "subAttributes",
        "type": "complex",
        "multiValued": true,
        "description": "Used to define the sub-attributes of a complex
attribute.",
        "required": false,
        "mutability": "readOnly",
        "returned": "default",
        "subAttributes" : [
            {
                "name": "name",
                "type": "string",
                "multiValued": false,
                "description": "The attribute's name.",
                "required": true,
                "caseExact": true,
                "mutability": "readOnly",
                "returned": "default",
                "uniqueness": "none"
            },
            {
                "name": "type",
                "type": "string",
                "multiValued": false,
                "description": "The attribute's data type.
Valid values include 'string', 'complex', 'boolean',
'decimal', 'integer', 'dateTime', 'reference'.",
                "required": true,
                "caseExact": false,
                "canonicalValues": [
                    "string",
                    "complex",
                    "boolean",
                    "decimal",
                    "integer",
                    "dateTime",
                    "reference"
                ]
            }
        ]
    }
]

```

```

    ],
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "multiValued",
    "type": "boolean",
    "multiValued": false,
    "description": "A Boolean value indicating an attribute's
plurality.",
    "required": true,
    "mutability": "readOnly",
    "returned": "default"
  },
  {
    "name": "description",
    "type": "string",
    "multiValued": false,
    "description": "A human-readable description of the
attribute.",
    "required": false,
    "caseExact": true,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "required",
    "type": "boolean",
    "multiValued": false,
    "description": "A Boolean value indicating whether or
not the attribute is required.",
    "required": false,
    "mutability": "readOnly",
    "returned": "default"
  },
  {
    "name": "canonicalValues",
    "type": "string",
    "multiValued": true,
    "description": "A collection of canonical values.",
    "required": false,
    "caseExact": true,
    "mutability": "readOnly",
    "returned": "default",

```

```

    "uniqueness": "none"
  },
  {
    "name": "caseExact",
    "type": "boolean",
    "multiValued": false,
    "description": "A Boolean value indicating whether or
      not a string attribute is case sensitive.",
    "required": false,
    "mutability": "readOnly",
    "returned": "default"
  },
  {
    "name": "mutability",
    "type": "string",
    "multiValued": false,
    "description": "Indicates whether or not an
      attribute is modifiable.",
    "required": false,
    "caseExact": true,
    "canonicalValues": [
      "readOnly",
      "readWrite",
      "immutable",
      "writeOnly"
    ],
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "returned",
    "type": "string",
    "multiValued": false,
    "description": "Indicates when an attribute is
      returned in a response (e.g., to a query).",
    "required": false,
    "caseExact": true,
    "canonicalValues": [
      "always",
      "default",
      "never",
      "request"
    ],
    "mutability": "readOnly",
    "returned": "default",

```

```

        "uniqueness": "none"
      },
      {
        "name": "uniqueness",
        "type": "string",
        "multiValued": false,
        "description": "Indicates how unique a value must be.",
        "required": false,
        "caseExact": true,
        "canonicalValues": [
          "none",
          "server",
          "global"
        ],
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
      },
      {
        "name": "referenceTypes",
        "type": "string",
        "multiValued": true,
        "description": "Used only with an attribute of type
          'reference'. Specifies a SCIM resourceType that a
          reference attribute MAY refer to, e.g., 'User'.",
        "required": false,
        "caseExact": true,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
      }
    ]
  }
],
"meta" : {
  "resourceType": "Schema",
  "location" :
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:schemas:core:2.0:Schema"
},
{
  "id": "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
  "name": "Enterprise",
  "description": "Enterprise for User ResourceType.",

```

```

"attributes": [
  {
    "name": "employeeNumber",
    "type": "string",
    "multiValued": false,
    "description": "The employee number of the user. The maximum size for
this field is 20 character.",
    "required": false,
    "caseExact": true,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "server"
  }
],
"meta": {
  "resourceType": "Schema",
  "location":
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:schemas:extension:enterprise:
2.0:User"
}
},
{
  "id": "urn:ietf:params:scim:api:messages:2.0:Error",
  "name": "Error",
  "description": "An error encountered during a SCIM operation.",
  "attributes": [
    {
      "name": "id",
      "type": "string",
      "multiValued": false,
      "description": "A unique identifier for the error.",
      "required": false,
      "caseExact": true,
      "mutability": "immutable",
      "returned": "never",
      "uniqueness": "none"
    },
    {
      "name": "status",
      "type": "integer",
      "multiValued": false,
      "description": "The HTTP status code for the error.",
      "required": true,
      "caseExact": true,
      "mutability": "immutable",
      "returned": "default",

```

```

    "uniqueness": "none"
  },
  {
    "name": "detail",
    "type": "string",
    "multiValued": false,
    "description": "A human-readable detailed error message.",
    "required": false,
    "caseExact": false,
    "mutability": "immutable",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "scimType",
    "type": "string",
    "multiValued": false,
    "description": "The SCIM error code.",
    "required": false,
    "caseExact": true,
    "mutability": "immutable",
    "returned": "default",
    "uniqueness": "none"
  },
],
"meta" : {
  "resourceType": "Schema",
  "location" :
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:api:messages:2.0:Error"
}
}
]

```

Example Error Responses

Error Scenario: A request to this endpoint which contains filtering options will result in an error.

Example Error:

```

HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "scimType": "invalidFilter",
  "detail": "Filtering is not supported."
}

```

Error Scenario: A request to this endpoint is made for an invalid organization.

Example Error:

```
HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "detail": "Operation is not permitted based on the supplied authorization."
}
```

Error Scenario: A request is made to this endpoint from an organization with multiple groups.

Example Error:

```
HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations with multiple groups."
}
```

Error Scenario: A request is made to this endpoint from an organization with no group.

Example Error:

```
HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations without groups."
}
```

Schema Definition

Endpoint: **/scim/v3/schemas/{schemaUri}**
 HTTP Operation: **GET**

A HTTP GET to this endpoint can be used to return a single SCIM Schema supported by Gerimedica.

This endpoint is read only.

Filtering, sorting and pagination are not supported.

The following values are valid for the path parameter **schemaUri**:

Schema Name	schemaUri parameter value
User	urn:ietf:params:scim:schemas:core:2.0:User
Ysis User Extension	urn:ietf:params:scim:schemas:extension:ysis:2.0:User
User Enterprise	urn:ietf:params:scim:schemas:extension:enterprise:2.0:User
ResourceType	urn:ietf:params:scim:schemas:core:2.0:ResourceType
Service Provider Config	urn:ietf:params:scim:schemas:core:2.0:ServiceProviderConfig
Schema	urn:ietf:params:scim:schemas:core:2.0:Schema
Error	urn:ietf:params:scim:api:messages:2.0:Error

Example Successful Response - User

Example response for request **/scim/v3/schemas/urn:ietf:params:scim:schemas:core:2.0:User**

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "id": "urn:ietf:params:scim:schemas:core:2.0:User",
  "name": "User",
  "description": "User Account",
  "attributes": [
    {
      "name": "userName",
      "type": "string",
      "multiValued": false,
      "description": "Unique identifier for the User, typically
used by the user to directly authenticate to the service provider."
    }
  ]
}
```

```

Each User MUST include a non-empty userName value. This identifier
MUST be unique across the service provider's entire set of Users.
The maximum size for this field is 40 characters.",
    "required": true,
    "caseExact": false,
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "server"
},
{
    "name": "name",
    "type": "complex",
    "multiValued": false,
    "description": "The components of the user's real name.",
    "required": false,
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "none"
    "subAttributes": [
        {
            "name": "familyName",
            "type": "string",
            "multiValued": false,
            "description": "The family name of the User, or
last name in most Western languages (e.g., 'Jensen' given the full
name 'Ms. Barbara Jensen'). The maximum size for this field is 100
characters.",
            "required": true,
            "caseExact": true,
            "mutability": "readWrite",
            "returned": "default",
            "uniqueness": "none"
        },
        {
            "name": "givenName",
            "type": "string",
            "multiValued": false,
            "description": "The given name of the User, or
first name in most Western languages (e.g., 'Barbara' given the
full name 'Ms. Barbara Jensen'). The maximum size for this field is 80
characters.",
            "required": true,
            "caseExact": true,
            "mutability": "readWrite",
            "returned": "default",
            "uniqueness": "none"
        }
    ]
}

```

```

    },
    {
      "name": "infix",
      "type": "string",
      "multiValued": false,
      "description": "It is a Tussenvoegsel in Dutch. In a family name
affix positioned between a person's given name and the main part of their
family name. For Example for name: Emma van der Sluss, van der is infix.",
      "required": false,
      "caseExact": false,
      "mutability": "readWrite",
      "returned": "default",
      "uniqueness": "none"
    }
  ]
},
{
  "name": "active",
  "type": "boolean",
  "multiValued": false,
  "description": "A Boolean value indicating the User's administrative
status.",
  "required": false,
  "mutability": "readWrite",
  "returned": "default"
},
{
  "name": "password",
  "type": "string",
  "multiValued": false,
  "description": "The User's cleartext password. This
attribute is intended to be used as a means to specify an initial
password when creating a new User. A password is OPTIONAL when creating a
user and must not be present otherwise. If the password is not present at
creation, it will be auto generated. A password must be a minimum of 12
characters and a maximum of 64 characters. All characters are valid except
'/' and '\\'.
",
  "required": false,
  "caseExact": true,
  "mutability": "writeOnly",
  "returned": "never",
  "uniqueness": "none"
},
{

```

```

    "name": "bigNumber",
    "type": "string",
    "multiValued": false,
    "description": "Bignumber of the User. The maximum size for this
field is 11 characters.",
    "required": false,
    "caseExact": true,
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "server"
  },
  {
    "name": "position",
    "type": "string",
    "multiValued": false,
    "description": "The function of the user. The maximum size for this
field is 255 character.",
    "required": false,
    "caseExact": true,
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "emails",
    "type": "complex",
    "multiValued": true,
    "description": "Email addresses for the user. Gerimedica can only
store one email address per user. Ideally one email address will be marked as
'primary' and if so, this will be saved. If no email address is marked as
primary and more than one email address is supplied, only one will be saved.
The email address saved will prioritize the 'work' over the 'home' and
'other' email addresses. If more than one of a given type is supplied, the
first email address with the highest priority based upon its type will be
accepted.",
    "required": false,
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "none",
    "subAttributes" : [
      {
        "name": "value",
        "type": "string",
        "multiValued": false,
        "description": "The email address value of the user. The maximum
size for this field is 255 characters",

```

```

        "required": false,
        "caseExact": false,
        "mutability": "readWrite",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "primary",
        "type": "boolean",
        "multiValued": false,
        "description": "A Boolean value indicating the 'primary'
or preferred email address of this user. The primary attribute value 'true'
MUST appear no more than once.",
        "required": false,
        "mutability": "readWrite",
        "returned": "never"
    }
]
},
{
    "name": "phoneNumbers",
    "type": "complex",
    "multiValued": true,
    "description": "Phone numbers for the User. Gerimedica can save both
a mobile phone number and another phone number. Canonical type values of
'work' and 'mobile' will be accepted. If more than one value of a given type
is supplied, one should be marked with 'primary' and will be prioritized. If
more than one value of a given type is supplied without being marked as
'primary', the first will be accepted. If more than one phone number of a
given type is supplied and more than one is marked as 'primary', the first
supplied will be accepted.",
    "required": false,
    "mutability": "readWrite",
    "returned": "default",
    "subAttributes": [
        {
            "name": "value",
            "type": "string",
            "multiValued": false,
            "description": "Phone number of the User.",
            "required": false,
            "caseExact": false,
            "mutability": "readWrite",
            "returned": "default",
            "uniqueness": "none"
        }
    ]
},

```

```

    {
      "name": "type",
      "type": "string",
      "multiValued": false,
      "description": "A label indicating the attribute's function,
e.g., 'work', or 'mobile'.",
      "required": false,
      "caseExact": false,
      "canonicalValues": [
        "work",
        "mobile"
      ],
      "mutability": "readWrite",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "primary",
      "type": "boolean",
      "multiValued": false,
      "description": "A Boolean value indicating the 'primary'
or preferred phone number. The primary attribute value 'true' MUST appear no
more than once per type.",
      "required": false,
      "mutability": "readWrite",
      "returned": "never"
    }
  ]
},
{
  "name": "roles",
  "type": "complex",
  "multiValued": true,
  "description": "A list of roles for the User that collectively
represent what activities the User will be undertaking while using this
system. A role is used to identify a set of authorizations the user is
granted within the system",
  "required": false,
  "mutability": "readWrite",
  "returned": "default",
  "subAttributes": [
    {
      "name": "value",
      "type": "string",
      "multiValued": false,
      "description": "The unique identifier for the role.",

```

```

        "required": false,
        "caseExact": true,
        "mutability": "readWrite",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "displayName",
        "type": "string",
        "multiValued": false,
        "description": "The name of the role.",
        "required": false,
        "caseExact": true,
        "mutability": "readWrite",
        "returned": "default",
        "uniqueness": "none"
    }
]
},
{
    "name": "entitlements",
    "type": "complex",
    "multiValued": true,
    "description": "A list of authorizations that the user will be able
to undertake within the system. Valid entitlement values must be one of the
canonical values.",
    "required": false,
    "mutability": "readWrite",
    "returned": "default",
    "subAttributes": [
        {
            "name": "value",
            "type": "string",
            "multiValued": false,
            "description": "The value of an entitlement.",
            "required": false,
            "caseExact": false,
            "canonicalValues": [
                "AccorderenGRZBasisGGZELV",
                "AccorderenWlz",
                "AccorderenWmo",
                "BehandelplanAccorderen",
                "BehandelrelatieWijzigen",
                "BeheerAdviessjablonen",
                "BeheerBriefsjablonen",
                "BeheerMedewerkers",
            ]
        }
    ]
}

```

```

"BeheerOverig",
"BeheerTijdregistratie",
"BeheerUitvalbak",
"DBCZorgactiviteitenRegistreren",
"DBCZorgtrajectenAanmakenBewerkenVerwijderen",
"DeclarerenGRZBasiGGZELVZPM",
"DeclarerenEerstelijnsParamedischeZorgHuisartsenzorgGZSP",
"DeclarerenWlz",
"DeclarerenWmo",
"DocumentenBeherenVoorAlleDisciplines",
"DocumentenBeherenVoorEigenDiscipline",
"DocumentenLezenVoorEigenDiscipline",
"ELVTijdRegistreren",
"HADeclaratie",
"HAVerrichtingenWijzigen",
"JuridischeInformatieBewerken",
"LaboratoriumresultatenLezen",
"MedischTabblad",
"MeetinstrumentenInvullenBekijken",
"MetingenVerwijderenNa24Uur",
"MultidisciplinairTabblad",
"PatiëntenMutereren",
"RapportageGenereren",
"RapportagesVerwijderenNa24Uur",
"ToegangBonstat",
"ToegangFarmedRx",
"ToegangFierit",
"ToegangHINQPortaal",
"ToegangMedimo",
"ToegangNCare",
"ToegangQuestManagerROMPortaal",
"ToegangRoosterplatform",
"ToegangVIPLive",
"ToegangViPharma",
"ToegangZorgDomein",
"ToevoegenBewerkenVerwijderenFinanciering",
"ToevoegenBewerkenVerwijderenVerzekeringsgegevens",
"VerslagenLezenActiviteitenbegeleiding",
"VerslagenLezenDiëtetiek",
"VerslagenLezenErgotherapie",
"VerslagenLezenFysiotherapie",
"VerslagenLezenGeestelijkeVerzorging",
"VerslagenLezenLogopedie",
"VerslagenLezenMaatschappelijkWerk",
"VerslagenLezenMedischeDiscipline",
"VerslagenLezenMondzorg",

```

```

        "VerslagenLezenMuziektherapie",
        "VerslagenLezenPsychologie",
        "VerslagenLezenPsychologieConsultatie",
        "VerslagenLezenPsychomotorischeTherapie",
        "VerslagenLezenVerpleegkunde",
        "VerslagenLezenZorg",
        "VerslagenToevoegen",
        "VertrouwelijkVerslagLezenActiviteitenbegeleiding",
        "VertrouwelijkVerslagLezenDiëtetiek",
        "VertrouwelijkVerslagLezenErgotherapie",
        "VertrouwelijkVerslagLezenFysiotherapie",
        "VertrouwelijkVerslagLezenGeestelijkeVerzorging",
        "VertrouwelijkVerslagLezenLogopedie",
        "VertrouwelijkVerslagLezenMaatschappelijkWerk",
        "VertrouwelijkVerslagLezenMedischeDiscipline",
        "VertrouwelijkVerslagLezenMondzorg",
        "VertrouwelijkVerslagLezenMuziektherapie",
        "VertrouwelijkVerslagLezenPsychologie",
        "VertrouwelijkVerslagLezenPsychologieConsultatie",
        "VertrouwelijkVerslagLezenPsychomotorischeTherapie",
        "VertrouwelijkVerslagLezenVerpleegkunde",
        "VerwerkenExterneVerwijzingen",
        "VerwijderenVragenlijsten",
        "VerwijzenNaarAndereAisciplines",
        "VolledigDossierPrinten",
        "ZorgplanAccorderen"
    ],
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "none"
}
]
}
],
"meta" : {
    "resourceType": "Schema",
    "location" :
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:schemas:core:2.0:User"
}
}

```

Example Successful Response - User Enterprise Extension

Example response for request:

scim/v3/schemas/urn:ietf:params:scim:schemas:extension:enterprise:2.0:User

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "id": "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User",
  "name": "Enterprise",
  "description": "Enterprise for User ResourceType.",
  "attributes": [
    {
      "name": "employeeNumber",
      "type": "string",
      "multiValued": false,
      "description": "The employee number of the user. The maximum size for
this field is 20 character.",
      "required": false,
      "caseExact": true,
      "mutability": "readOnly",
      "returned": "default",
      "uniqueness": "server"
    }
  ],
  "meta": {
    "resourceType": "Schema",
    "location": "$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:schemas:extension:
enterprise:2.0:User"
  }
}
```

Example Successful Response - Ysis User Extension

Example response for request

/scim/v3/schemas/urn:ietf:params:scim:schemas:extension:ysis:2.0:User

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "id": "urn:ietf:params:scim:schemas:extension:ysis:2.0:User",
  "name": "User",
  "description": "Ysis extension for User ResourceType.",
  "attributes": [
    {
      "name": "discipline",
      "type": "string",
      "multiValued": false,
      "description": "Identifier to indicate the Ysis Discipline of this
User. Each User MUST include a non-empty discipline value, which must belong
to the set of canonical values.",
      "required": true,
      "caseExact": true,
      "canonicalValues": [
        "ACTIVITY_COACH",
        "ADMINISTRATIVE",
        "CARE",
        "DENTAL_CARE",
        "DIETITIAN",
        "ERGOTHERAPIST",
        "LOGOPEDIST",
        "MEDICAL",
        "MUSIC_THERAPIST",
        "NURSE",
        "PASTOR",
        "PHYSIOTHERAPIST",
        "PMT",
        "PSYCHOLOGIST",
        "PSYCHOLOGIST_CONSULTANT",
        "READ_ONLY",
        "SOCIAL_WORK"
      ],
      "mutability": "immutable",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "ysisInitials",
      "type": "string",
      "multiValued": false,
```

```

        "description": "Ysis Initials are used to uniquely identify a user
within the Ysis system. Each User MUST have a unique non-empty Ysis initials
value. The value should consist of alphanumeric characters. the maximum size
for this field is 10 characters.",
        "required": true,
        "caseExact": true,
        "mutability": "immutable",
        "returned": "default",
        "uniqueness": "server"
    },
    {
        "name": "initials",
        "type": "string",
        "multiValued": false,
        "description": "Initials are used to identify a user within the Ysis
system. Each User MUST have a unique non-empty value. The value should
consist of alphanumeric characters. the maximum size for this field is 10
characters",
        "required": true,
        "caseExact": true,
        "mutability": "immutable",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "agbCode",
        "type": "string",
        "multiValued": false,
        "description": "AGB Code of this User. AGB Codes should be numeric
and 8 characters in length.",
        "required": false,
        "caseExact": true,
        "mutability": "readWrite",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "modules",
        "type": "string",
        "multiValued": true,
        "description": "Set of identifiers representing the YSIS Modules to
enable for this User. The set of modules can be empty.",
        "required": false,
        "caseExact": true,
        "canonicalValues": [
            "ECD_ADMIN",

```

```

        "ECD_CARE",
        "FINANCIAL",
        "FINANCIAL_EXPORT_PORTAL",
        "USER_MANAGEMENT"
        "YSIS_CORE",
        "YSIS_DBC"
    ],
    "mutability": "readWrite",
    "returned": "default",
    "uniqueness": "none"
}
],
"meta" : {
    "resourceType": "Schema",
    "location":
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:schemas:extension:ysis:2.0:User"
}
}

```

Example Successful Response - ResourceType

Example response for request

/scim/v3/schemas/urn:ietf:params:scim:schemas:core:2.0:ResourceType

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "id": "urn:ietf:params:scim:schemas:core:2.0:ResourceType",
  "name": "ResourceType",
  "description": "Describes a SCIM Resource Type.",
  "attributes": [
    {
      "name": "id",
      "type": "string",
      "multiValued": false,
      "description": "The resource type's server unique id.",
      "required": false,
      "caseExact": false,
      "mutability": "readOnly",
      "returned": "default",
      "uniqueness": "server"
    },
    {
      "name": "name",
      "type": "string",
      "multiValued": false,
      "description": "The resource type's name. e.g., 'user'.",
      "required": true,
      "caseExact": false,
      "mutability": "readOnly",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "description",
      "type": "string",
      "multiValued": false,
      "description": "The resource type's human-readable description.",
      "required": false,
      "caseExact": false,
      "mutability": "readOnly",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "endpoint",
      "type": "reference",

```

```

    "referenceTypes": ["uri"],
    "multiValued": false,
    "description": "The resource type's HTTP-addressable endpoint
relative to the Base URL, e.g., '/users'.",
    "required": true,
    "caseExact": false,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "schema",
    "type": "reference",
    "referenceTypes" : ["uri"],
    "multiValued": false,
    "description": "The resource type's primary/base schema URI.",
    "required": true,
    "caseExact": true,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "schemaExtensions",
    "type": "complex",
    "multiValued": false,
    "description": "A list of URIs of the resource type's schema
extensions.",
    "required": true,
    "mutability": "readOnly",
    "returned": "default",
    "subAttributes" : [
      {
        "name": "schema",
        "type": "reference",
        "referenceTypes": ["uri"],
        "multiValued": false,
        "description": "The URI of a schema extension.",
        "required": true,
        "caseExact": true,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
      },
      {
        "name": "required",

```

```

        "type": "boolean",
        "multiValued": false,
        "description": "A Boolean value that specifies whether
                        or not the schema extension is required for the
                        resource type. If true, a resource of this type MUST
                        include this schema extension and also include any
                        attributes declared as required in this schema extension.
                        If false, a resource of this type MAY omit this schema
                        extension.",
        "required": true,
        "mutability": "readOnly",
        "returned": "default"
    }
]
},
"meta" : {
    "resourceType": "Schema",
    "location" :
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:schemas:core:2.0:ResourceType
"
}
}

```

Example Successful Response - ServiceProviderConfig

Example response for request

/scim/v3/schemas/urn:ietf:params:scim:schemas:core:2.0:ServiceProviderConfig

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "id": "urn:ietf:params:scim:schemas:core:2.0:ServiceProviderConfig",
  "name": "ServiceProviderConfig",
  "description": "Schema representing the service provider's
configuration.",
  "attributes" : [
    {
      "name": "documentationUri",
      "type": "reference",
      "referenceTypes" : ["external"],
      "multiValued": false,
      "description": "An HTTP-addressable URL pointing to the
service provider's human-consumable help documentation.",
      "required": false,
      "caseExact": false,
      "mutability": "readOnly",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "patch",
      "type": "complex",
      "multiValued": false,
      "description": "A complex type that specifies PATCH configuration
options.",
      "required": true,
      "mutability": "readOnly",
      "returned": "default",
      "subAttributes" : [
        {
          "name": "supported",
          "type": "boolean",
          "multiValued": false,
          "description": "A Boolean value specifying whether or not
the operation is supported.",
          "required": true,
          "mutability": "readOnly",
          "returned": "default"
        }
      ]
    }
  ],
}
```

```

{
  "name": "bulk",
  "type": "complex",
  "multiValued": false,
  "description": "A complex type that specifies bulk configuration
options.",
  "required": true,
  "mutability": "readOnly",
  "returned": "default",
  "subAttributes" : [
    {
      "name": "supported",
      "type": "boolean",
      "multiValued": false,
      "description": "A Boolean value specifying whether or not
the operation is supported.",
      "required": true,
      "mutability": "readOnly",
      "returned": "default"
    },
    {
      "name": "maxOperations",
      "type": "integer",
      "multiValued": false,
      "description": "An integer value specifying the maximum
number of operations.",
      "required": true,
      "mutability": "readOnly",
      "uniqueness": "none",
      "returned": "default"
    },
    {
      "name": "maxPayloadSize",
      "type": "integer",
      "multiValued": false,
      "description": "An integer value specifying the maximum
payload size in bytes.",
      "required": true,
      "mutability": "readOnly",
      "uniqueness": "none",
      "returned": "default",
    }
  ]
},
{
  "name": "filter",
  "type": "complex",

```

```

"multiValued": false,
"description": "A complex type that specifies FILTER options.",
"required": true,
"mutability": "readOnly",
"returned": "default",
"subAttributes": [
  {
    "name": "supported",
    "type": "boolean",
    "multiValued": false,
    "description": "A Boolean value specifying whether or not
      the operation is supported.",
    "required": true,
    "mutability": "readOnly",
    "returned": "default"
  },
  {
    "name": "maxResults",
    "type": "integer",
    "multiValued": false,
    "description": "An integer value specifying the maximum
      number of resources returned in a response.",
    "required": true,
    "mutability": "readOnly",
    "uniqueness": "none",
    "returned": "default"
  }
]
},
{
  "name": "changePassword",
  "type": "complex",
  "multiValued": false,
  "description": "A complex type that specifies configuration
    options related to changing a password.",
  "required": true,
  "mutability": "readOnly",
  "returned": "default",
  "subAttributes": [
    {
      "name": "supported",
      "type": "boolean",
      "multiValued": false,
      "description": "A Boolean value specifying whether or not
        the operation is supported.",
      "required": true,

```

```

        "mutability": "readOnly",
        "returned": "default"
    }
]
},
{
    "name": "sort",
    "type": "complex",
    "multiValued": false,
    "description": "A complex type that specifies sort result options.",
    "required": true,
    "mutability": "readOnly",
    "returned": "default",
    "subAttributes": [
        {
            "name": "supported",
            "type": "boolean",
            "multiValued": false,
            "description": "A Boolean value specifying whether or not
                the operation is supported.",
            "required": true,
            "mutability": "readOnly",
            "returned": "default"
        }
    ]
},
{
    "name": "authenticationSchemes",
    "type": "complex",
    "multiValued": true,
    "description": "A complex type that specifies supported
        authentication scheme properties.",
    "required": true,
    "mutability": "readOnly",
    "returned": "default",
    "subAttributes": [
        {
            "name": "name",
            "type": "string",
            "multiValued": false,
            "description": "The common authentication scheme name, e.g., HTTP
Basic.",
            "required": true,
            "caseExact": false,
            "mutability": "readOnly",
            "returned": "default",

```

```

        "uniqueness": "none"
    },
    {
        "name": "description",
        "type": "string",
        "multiValued": false,
        "description": "A description of the authentication scheme.",
        "required": true,
        "caseExact": false,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "specUri",
        "type": "reference",
        "referenceTypes": ["external"],
        "multiValued": false,
        "description": "An HTTP-addressable URL pointing to the
            authentication scheme's specification.",
        "required": false,
        "caseExact": false,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "documentationUri",
        "type": "reference",
        "referenceTypes": ["external"],
        "multiValued": false,
        "description": "An HTTP-addressable URL pointing to the
            authentication scheme's usage documentation.",
        "required": false,
        "caseExact": false,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    }
]
},
"meta" : {
    "resourceType": "Schema",
    "location" :
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:schemas:core:2.0:ServiceProvi

```

```
derConfig"  
  }  
}
```

Example Successful Response - Schema

Example response for request

/scim/v3/schemas/urn:ietf:params:scim:schemas:core:2.0:Schema

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "id": "urn:ietf:params:scim:schemas:core:2.0:Schema",
  "name": "Schema",
  "description": "Specifies the schema that describes a SCIM schema.",
  "attributes": [
    {
      "name": "id",
      "type": "string",
      "multiValued": false,
      "description": "The unique URI of the schema.",
      "required": true,
      "caseExact": true,
      "mutability": "readOnly",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "name",
      "type": "string",
      "multiValued": false,
      "description": "The schema's human-readable name. e.g., 'user'.",
      "required": true,
      "caseExact": false,
      "mutability": "readOnly",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "description",
      "type": "string",
      "multiValued": false,
      "description": "A human-readable description of the schema.",
      "required": false,
      "caseExact": false,
      "mutability": "readOnly",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "attributes",
      "type": "complex",
```

```

    "multiValued": true,
    "description": "A complex attribute that includes the attributes of a
schema.",
    "required": true,
    "mutability": "readOnly",
    "returned": "default",
    "subAttributes": [
      {
        "name": "name",
        "type": "string",
        "multiValued": false,
        "description": "The attribute's name.",
        "required": true,
        "caseExact": true,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
      },
      {
        "name": "type",
        "type": "string",
        "multiValued": false,
        "description": "The attribute's data type.
Valid values include 'string', 'complex', 'boolean',
'decimal', 'integer', 'dateTime', 'reference'.",
        "required": true,
        "caseExact": false,
        "canonicalValues": [
          "string",
          "complex",
          "boolean",
          "decimal",
          "integer",
          "dateTime",
          "reference"
        ],
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
      },
      {
        "name": "multiValued",
        "type": "boolean",
        "multiValued": false,
        "description": "A Boolean value indicating an attribute's
plurality.",

```

```

        "required": true,
        "mutability": "readOnly",
        "returned": "default"
    },
    {
        "name": "description",
        "type": "string",
        "multiValued": false,
        "description": "A human-readable description of the attribute.",
        "required": false,
        "caseExact": true,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "required",
        "type": "boolean",
        "multiValued": false,
        "description": "A Boolean value indicating whether or not the
attribute is required.",
        "required": false,
        "mutability": "readOnly",
        "returned": "default"
    },
    {
        "name": "canonicalValues",
        "type": "string",
        "multiValued": true,
        "description": "A collection of canonical values.",
        "required": false,
        "caseExact": true,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "caseExact",
        "type": "boolean",
        "multiValued": false,
        "description": "A Boolean value indicating whether or
not a string attribute is case sensitive.",
        "required": false,
        "mutability": "readOnly",
        "returned": "default"
    },

```

```

    {
      "name": "mutability",
      "type": "string",
      "multiValued": false,
      "description": "Indicates whether or not an attribute is
modifiable.",
      "required": false,
      "caseExact": true,
      "canonicalValues": [
        "readOnly",
        "readWrite",
        "immutable",
        "writeOnly"
      ],
      "mutability": "readOnly",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "returned",
      "type": "string",
      "multiValued": false,
      "description": "Indicates when an attribute is returned
        in a response (e.g., to a query).",
      "required": false,
      "caseExact": true,
      "canonicalValues": [
        "always",
        "default",
        "never",
        "request"
      ],
      "mutability": "readOnly",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "uniqueness",
      "type": "string",
      "multiValued": false,
      "description": "Indicates how unique a value must be.",
      "required": false,
      "caseExact": true,
      "canonicalValues": [
        "server",
        "global"
      ]
    }
  ]
}

```

```

        "none",
    ],
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
},
{
    "name": "referenceTypes",
    "type": "string",
    "multiValued": true,
    "description": "Used only with an attribute of type
        'reference'. Specifies a SCIM resourceType that a
        reference attribute MAY refer to, e.g., 'User'.",
    "required": false,
    "caseExact": true,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
},
{
    "name": "subAttributes",
    "type": "complex",
    "multiValued": true,
    "description": "Used to define the sub-attributes of a complex
attribute.",
    "required": false,
    "mutability": "readOnly",
    "returned": "default",
    "subAttributes" : [
        {
            "name": "name",
            "type": "string",
            "multiValued": false,
            "description": "The attribute's name.",
            "required": true,
            "caseExact": true,
            "mutability": "readOnly",
            "returned": "default",
            "uniqueness": "none"
        },
        {
            "name": "type",
            "type": "string",
            "multiValued": false,
            "description": "The attribute's data type.
                Valid values include 'string', 'complex', 'boolean',

```

```

        'decimal', 'integer', 'dateTime', 'reference'.",
    "required": true,
    "caseExact": false,
    "canonicalValues": [
        "string",
        "complex",
        "boolean",
        "decimal",
        "integer",
        "dateTime",
        "reference"
    ],
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
},
{
    "name": "multiValued",
    "type": "boolean",
    "multiValued": false,
    "description": "A Boolean value indicating an attribute's
plurality.",
    "required": true,
    "mutability": "readOnly",
    "returned": "default"
},
{
    "name": "description",
    "type": "string",
    "multiValued": false,
    "description": "A human-readable description of the
attribute.",
    "required": false,
    "caseExact": true,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
},
{
    "name": "required",
    "type": "boolean",
    "multiValued": false,
    "description": "A Boolean value indicating whether or
        not the attribute is required.",
    "required": false,
    "mutability": "readOnly",

```

```

    "returned": "default"
  },
  {
    "name": "canonicalValues",
    "type": "string",
    "multiValued": true,
    "description": "A collection of canonical values.",
    "required": false,
    "caseExact": true,
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "caseExact",
    "type": "boolean",
    "multiValued": false,
    "description": "A Boolean value indicating whether or
      not a string attribute is case sensitive.",
    "required": false,
    "mutability": "readOnly",
    "returned": "default"
  },
  {
    "name": "mutability",
    "type": "string",
    "multiValued": false,
    "description": "Indicates whether or not an
      attribute is modifiable.",
    "required": false,
    "caseExact": true,
    "canonicalValues": [
      "readOnly",
      "readWrite",
      "immutable",
      "writeOnly"
    ],
    "mutability": "readOnly",
    "returned": "default",
    "uniqueness": "none"
  },
  {
    "name": "returned",
    "type": "string",
    "multiValued": false,
    "description": "Indicates when an attribute is

```

```

        returned in a response (e.g., to a query).",
        "required": false,
        "caseExact": true,
        "canonicalValues": [
            "always",
            "default",
            "never",
            "request"
        ],
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "uniqueness",
        "type": "string",
        "multiValued": false,
        "description": "Indicates how unique a value must be.",
        "required": false,
        "caseExact": true,
        "canonicalValues": [
            "none",
            "server",
            "global"
        ],
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    },
    {
        "name": "referenceTypes",
        "type": "string",
        "multiValued": true,
        "description": "Used only with an attribute of type
            'reference'. Specifies a SCIM resourceType that a
            reference attribute MAY refer to, e.g., 'User'.",
        "required": false,
        "caseExact": true,
        "mutability": "readOnly",
        "returned": "default",
        "uniqueness": "none"
    }
]
}

```

```
    ],  
    "meta" : {  
      "resourceType": "Schema",  
      "location" :  
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:schemas:core:2.0:Schema"  
    }  
  }  
}
```

Example Successful Response - Error

Example response for request `/scim/v3/schemas/urn:ietf:params:scim:api:messages:2.0:Error`

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "id": "urn:ietf:params:scim:api:messages:2.0:Error",
  "name": "Error",
  "description": "An error encountered during a SCIM operation.",
  "attributes": [
    {
      "name": "id",
      "type": "string",
      "multiValued": false,
      "description": "A unique identifier for the error.",
      "required": false,
      "caseExact": true,
      "mutability": "immutable",
      "returned": "never",
      "uniqueness": "none"
    },
    {
      "name": "status",
      "type": "integer",
      "multiValued": false,
      "description": "The HTTP status code for the error.",
      "required": true,
      "caseExact": true,
      "mutability": "immutable",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "detail",
      "type": "string",
      "multiValued": false,
      "description": "A human-readable detailed error message.",
      "required": false,
      "caseExact": false,
      "mutability": "immutable",
      "returned": "default",
      "uniqueness": "none"
    },
    {
      "name": "scimType",
      "type": "string",
      "multiValued": false,
```

```

        "description": "The SCIM error code.",
        "required": false,
        "caseExact": true,
        "mutability": "immutable",
        "returned": "default",
        "uniqueness": "none"
    }
],
    "meta" : {
        "resourceType": "Schema",
        "location" :
"$BASE_URL/scim/v3/schemas/urn:ietf:params:scim:api:messages:2.0:Error"
    }
}

```

Example Error Responses

Error Scenario: A request is sent to this endpoint containing filtering options.

Error Example:

```

HTTP/1.1 403 FORBIDDEN
{
    "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
    "status": 403,
    "scimType": "invalidFilter",
    "detail": "Filtering of ResourceTypes is not supported."
}

```

Error Scenario: A request is sent to this endpoint with an invalid path parameter.

Error Example:

```

HTTP/1.1 400 BAD_REQUEST
{
    "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
    "status": 400,
    "scimType": "invalidValue",
    "detail": "The path parameter 'urn:ietf:params:scim:schemas:core:2.0:NA' for
'schemaUri' is not supported."
}

```

Error Scenario: A request to this endpoint is made for an invalid organization.

Example Error:

```

HTTP/1.1 403 FORBIDDEN
{
    "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],

```

```

"status": 403,
"detail": "Operation is not permitted based on the supplied authorization."
}

```

Error Scenario: A request is made to this endpoint from an organization with multiple groups.

Example Error:

```

HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations with multiple groups."
}

```

Error Scenario: A request is made to this endpoint from an organization with no group.

Example Error:

```

HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations without groups."
}

```

Scim User Operations

Creating Users

Endpoint: **\$BASE_URL/scim/v3/users**
 HTTP Operation: **POST**
 Authorization: **Bearer Token**

A HTTP POST to this endpoint can be used to create a Gerimedica User for the caller's organization. Error responses will be returned if invalid values are attempted to be set. See the 'Schema Definition - User' and 'Schema Definition - Error' sections for the full set of field requirements.

Filtering, sorting, and pagination are not supported.

Example Request

```
{
  "schemas": [
    "urn:ietf:params:scim:schemas:core:2.0:User",
    "urn:ietf:params:scim:schemas:extension:ysis:2.0:User",
    "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User"
  ],
  "userName": "bjensen",
  "name": {
    "familyName": "Jensen",
    "givenName": "Barbara",
    "infix": "van"
  },
  "gender": null,
  "exportTimelineEvents": true,
  "bigNumber": null,
  "urn:ietf:params:scim:schemas:extension:ysis:2.0:User": {
    "ysisInitials": "Y.S",
    "discipline": "MEDICAL",
    "initials": "YS",
    "agbCode": "agbCode1",
    "modules": ["YSIS_CORE", "USER_MANAGEMENT"]
  },
  "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User": {
    "employeeNumber": "2015010213058"
  },
  "emails": [
    {
      "value": "bjensen@example.com",
      "primary": true,
      "type": "home"
    }
  ]
}
```

```

],
"phoneNumbers": [
  {
    "value": "555-555-5555",
    "type": "work"
  },
  {
    "value": "555-555-4444",
    "type": "mobile"
  }
],
"password": "t1meMa$heen",
"roles":[
  {
    "value": "51bc045c-d17c-4d29-bd3f-9b1avcf5a011",
    "displayName": "Fysiotherapie"
  },
  {
    "value": "13y79ab5-ec65-4733-95e5-49b5977733c7",
    "displayName": "Psychologie"
  }
],
"entitlements":[
  {
    "value": "BehandelplanAccorderen"
  },
  {
    "value": "BehandelrelatieWijzigen"
  }
]
}

```

Example Successful Response

```

{
  "schemas": ["urn:ietf:params:scim:schemas:core:2.0:User"],
  "id": "2819c223-7f76-453a-919d-413861904646",
  "userName": "bjensen",
  "name": {
    "familyName": "Jensen",
    "givenName": "Barbara",
    "infix": "van"
  },
  "gender": "UNKNOWN",
  "exportTimelineEvents": true,
  "bigNumber": null,
  "urn:ietf:params:scim:schemas:extension:ysis:2.0:User": {
    "ysisInitials": "Y.S",

```

```

    "discipline": "MEDICAL",
    "agbCode": "agbCode1",
    "Initials": "YS",
    "modules": ["YSIS_CORE", "USER_MANAGEMENT"]
  },
  "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User": {
    "employeeNumber": "2015010213058"
  },
  "emails": [
    {
      "value": "bjensen@example.com"
    }
  ],
  "phoneNumbers": [
    {
      "value": "555-555-5555",
      "type": "work"
    },
    {
      "value": "555-555-4444",
      "type": "mobile"
    }
  ],
  "active": true,
  "roles": [
    {
      "value": "51bc045c-d17c-4d29-bd3f-9b1avcf5a011",
      "displayName": "Fysiotherapie"
    },
    {
      "value": "13y79ab5-ec65-4733-95e5-49b5977733c7",
      "displayName": "Psychologie"
    }
  ],
  "entitlements": [
    {
      "value": "BehandelplanAccorderen"
    },
    {
      "value": "BehandelrelatieWijzigen"
    }
  ],
  "meta": {
    "resourceType": "User",
    "created": "2010-01-23T04:56:22Z",
    "lastModified": "2011-05-13T04:42:34Z",
    "location": "$BASE_URL/v3/users/2819c223-7f76-453a-919d-413861904646"
  }
}

```

Example Error Responses

Error Scenario: The caller attempts to create a User with a userName which already exists.

Example Error:

```
HTTP/1.1 409 Conflict
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 409,
  "scimType": "uniqueness",
  "detail": "A User with the 'userName' 'bjensen' already exists."
}
```

Error Scenario: A request to this endpoint is made for an invalid organization.

Example Error:

```
HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "detail": "Operation is not permitted based on the supplied authorization."
}
```

Error Scenario: The caller attempts to create a User with a value which is invalid for a validated field.

Example Error:

```
HTTP/1.1 400 BAD_REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "scimType": "invalidValue",
  "detail": "Invalid ysis initials, it should be max 10 character."
}
```

Error Scenario: A request is sent to this endpoint containing filtering options.

Error Example:

```
HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "scimType": "invalidFilter",
  "detail": "Filtering of ResourceTypes is not supported."
}
```

Error Scenario: A request is made to this endpoint from an organization with multiple groups.

Example Error:

```
HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
```

```
"status": 400,  
"detail": "SCIM is not currently supported for organizations with multiple groups."  
}
```

Error Scenario: A request is made to this endpoint from an organization with no group.

Example Error:

```
HTTP/1.1 400 BAD REQUEST  
{  
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],  
  "status": 400,  
  "detail": "SCIM is not currently supported for organizations without groups."  
}
```

Re-activating Users

Endpoint: **\$BASE_URL/scim/v3/users**
 HTTP Operation: **POST**
 Authorization: **Bearer Token**

This same endpoint used for creating Users (a HTTP POST to \$BASE_URL/scim/v3/users) is also used to re-activate a previously deleted (archived) Gerimedica User. If the userName, ysisInitials and discipline in the request all match a single archived User, that User is restored instead of a new User being created. The User's original userId is preserved, and any other mutable fields supplied in the request are applied in the same way as a normal Update.

The 'active' attribute in the request determines whether the re-activated User is enabled (true) or remains blocked (false). If userName, ysisInitials or discipline only partially match an archived User (for example the userName matches but the ysisInitials does not), the request is rejected with a 409 Conflict rather than falling through to a normal create or silently re-activating the wrong User.

Filtering, sorting, and pagination are not supported.

Example Request

```
{
  "schemas": [
    "urn:ietf:params:scim:schemas:core:2.0:User",
    "urn:ietf:params:scim:schemas:extension:ysis:2.0:User"
  ],
  "userName": "mjansen",
  "active": true,
  "name": {
    "familyName": "Jansen",
    "givenName": "Maria"
  },
  "gender": "UNKNOWN",
  "emails": [
    {
      "value": "maria.jansen@example.com",
      "primary": true,
      "type": "home"
    }
  ],
  "urn:ietf:params:scim:schemas:extension:ysis:2.0:User": {
    "ysisInitials": "M.J",
    "discipline": "MEDICAL",
    "initials": "MJ",
    "agbCode": "agbCode2",
    "modules": ["YSIS_CORE"]
  }
}
```

Example Successful Response

```
{
  "schemas": ["urn:ietf:params:scim:schemas:core:2.0:User"],
  "id": "3919c223-7f76-453a-919d-413861904647",
  "userName": "mjansen",
  "name": {
    "familyName": "Jansen",
    "givenName": "Maria"
  },
  "gender": "UNKNOWN",
  "urn:ietf:params:scim:schemas:extension:ysis:2.0:User": {
    "ysisInitials": "M.J",
    "discipline": "MEDICAL",
    "agbCode": "agbCode2",
    "initials": "MJ",
    "modules": ["YSIS_CORE"]
  },
  "emails": [
    {
      "value": "maria.jansen@example.com"
    }
  ],
  "active": true,
  "meta": {
    "resourceType": "User",
    "created": "2022-03-11T09:15:02Z",
    "lastModified": "2026-07-29T10:02:11Z",
    "location": "$BASE_URL/v3/users/3919c223-7f76-453a-919d-413861904647"
  }
}
```

Example Error Responses

Error Scenario: The caller attempts to re-activate a User where the userName matches an archived User, but the ysisInitials does not.

Example Error:

```
HTTP/1.1 409 Conflict
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 409,
  "scimType": "uniqueness",
  "detail": "Cannot create user: userName 'mjansen' matches a previously deleted user with ysisInitials 'M.J', but the request provides ysisInitials 'X.Y'. To re-activate, use the matching ysisInitials."
}
```

Error Scenario: The caller attempts to re-activate a User where the ysisInitials matches an archived User, but the userName does not.

Example Error:

```
HTTP/1.1 409 Conflict
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 409,
  "scimType": "uniqueness",
  "detail": "Cannot create user: ysisInitials 'M.J' matches a previously deleted user with userName 'mjansen', but the request provides userName 'ojansen'. To re-activate, use the matching userName."
}
```

Error Scenario: The caller attempts to re-activate a User where the userName and ysisInitials match an archived User, but the discipline does not.

Example Error:

```
HTTP/1.1 409 Conflict
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 409,
  "scimType": "uniqueness",
  "detail": "Cannot create user: userName 'mjansen' and ysisInitials 'M.J' match an archived user, but the requested discipline 'PHYSIOTHERAPIST' does not match the archived user's discipline 'MEDICAL'. To re-activate, use the matching discipline."
}
```

Retrieving Users (by userId)

Endpoint: **\$BASE_URL/scim/v3/users/{user_id}**
 HTTP Operation: **GET**
 Authorization: **Bearer Token**

A HTTP GET call to this endpoint can be used to return an existing Gerimedita user in the SCIM JSON format. Only Users from the caller's organization can be returned. Attempting to return a user which does not exist will return a 404 error.

Filtering, sorting and pagination are not supported when returning a single user.

Example Request

```
HTTP/1.1 GET $BASE_URL/scim/v3/users/2819c223-7f76-453a-919d-413861904646
Authorization: Bearer xxx
Content-Type: application/scim+json
```

Example Successful Response

```
HTTP/1.1 200 OK
Content-Type: application/json
Location: $BASE_URL/scim/v3/users/2819c223-7f76-453a-919d-413861904646
{
  "schemas": [
    "urn:ietf:params:scim:schemas:core:2.0:User",
    "urn:ietf:params:scim:schemas:extension:ysis:2.0:User",
    "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User"
  ],
  "id": "2819c223-7f76-453a-919d-413861904646",
  "userName": "bjensen",
  "name": {
    "familyName": "Jensen",
    "givenName": "Barbara"
  },
  "emails": [
    {
      "value": "bjensen@example.com",
    }
  ],
  "phoneNumbers": [
    {
      "value": "555-555-5555",
      "type": "work"
    },
    {
      "value": "555-555-4444",
      "type": "mobile"
    }
  ],
  "active": true,
```

```

"roles": [
  {
    "value": "51bc045c-d17c-4d29-bd3f-9b1avcf5a011",
    "displayName": "Fysiotherapie"
  },
  {
    "value": "13y79ab5-ec65-4733-95e5-49b5977733c7",
    "displayName": "Psychologie"
  }
],
"entitlements": [
  {
    "value": "BehandelplanAccorderen"
  },
  {
    "value": "BehandelrelatieWijzigen"
  }
],
"urn:ietf:params:scim:schemas:extension:ysis:2.0:User": {
  "ysisInitials": "Y.S",
  "discipline": "MEDICAL",
  "agbCode": "agbCode1",
  "initials": "YS"
},
"meta": {
  "resourceType": "User",
  "created": "2010-01-23T04:56:22Z",
  "lastModified": "2011-05-13T04:42:34Z",
  "location": "$BASE_URL/v3/users/2819c223-7f76-453a-919d-413861904646"
}
}

```

Example Error Responses

Error Scenario: The caller attempts to get a user that does not exist.

Error Example:

```

HTTP/1.1 404 Not Found
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 404,
  "detail": "Resource 2819c223-7f76-453a-919d-413861904646 not found"
}

```

Error Scenario: A request to this endpoint is made for an invalid organization.

Example Error:

```

HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],

```

```

"status": 403,
"detail": "Operation is not permitted based on the supplied authorization."
}

```

Error Scenario: A request is sent to this endpoint containing filtering options.

Error Example:

```

HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "scimType": "invalidFilter",
  "detail": "Filtering of ResourceTypes is not supported."
}

```

Error Scenario: A request is made to this endpoint from an organization with multiple groups.

Example Error:

```

HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations with multiple groups."
}

```

Error Scenario: A request is made to this endpoint from an organization with no group.

Example Error:

```

HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations without groups."
}

```

Retrieving Users (with Filter)

Endpoint: **\$BASE_URL/scim/v3/users**
 HTTP Operation: **GET**
 Authorization: **Bearer Token**

A HTTP GET call to this endpoint can be used to return a list of existing Gerimedita users in the SCIM JSON format. Only Users from the caller's organization can be returned. A filter must be provided for this endpoint. The currently supported filters can be found in the filtering section below.

Sorting and pagination are not supported.

Example Request

```
HTTP/1.1 GET
$BASE_URL/scim/v3/users?filter=urn:ietf:params:scim:schemas:extension:enterprise:2.0:
User:employeeNumber%20eq%20%221234%22
Authorization: Bearer xxx
Content-Type: application/scim+json
```

Example Successful Response

```
HTTP/1.1 200 OK
Content-Type: application/json
Location: $BASE_URL/scim/v3/users/2819c223-7f76-453a-919d-413861904646
{
  "schemas": [
    "urn:ietf:params:scim:api:messages:2.0:ListResponse"
  ],
  "totalResults": 1,
  "Resources": [
    {
      "schemas": [
        "urn:ietf:params:scim:schemas:core:2.0:User",
        "urn:ietf:params:scim:schemas:extension:ysis:2.0:User",
        "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User"
      ],
      "id": "2819c223-7f76-453a-919d-413861904646",
      "userName": "bjensen",
      "name": {
        "familyName": "Jensen",
        "givenName": "Barbara"
      },
      "emails": [
        {
          "value": "bjensen@example.com",
        }
      ],
      "phoneNumbers": [
        {
```

```

        "value": "555-555-5555",
        "type": "work"
      },
      {
        "value": "555-555-4444",
        "type": "mobile"
      }
    ],
    "active": true,
    "roles": [
      {
        "value": "51bc045c-d17c-4d29-bd3f-9b1avcf5a011",
        "displayName": "Fysiotherapie"
      },
      {
        "value": "13y79ab5-ec65-4733-95e5-49b5977733c7",
        "displayName": "Psychologie"
      }
    ],
    "entitlements": [
      {
        "value": "BehandelplanAccorderen"
      },
      {
        "value": "BehandelrelatieWijzigen"
      }
    ],
    "urn:ietf:params:scim:schemas:extension:ysis:2.0:User": {
      "ysisInitials": "Y.S",
      "discipline": "MEDICAL",
      "agbCode": "agbCode1",
      "initials": "YS"
    },
    "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User": {
      "employeeNumber": "1234"
    },
    "meta": {
      "resourceType": "User",
      "created": "2010-01-23T04:56:22Z",
      "lastModified": "2011-05-13T04:42:34Z",
      "location": "$BASE_URL/v3/users/2819c223-7f76-453a-919d-413861904646"
    }
  }
]

```

Example Error Responses

Error Scenario: The caller attempts to call without a filter: \$BASE_URL/scim/v3/users

Error Example:

```

HTTP/1.1 404 Forbidden
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],

```

```

    "status": 403,
    "detail": "Empty filters are not supported."
  }

```

Error Scenario: The caller attempts to call with an incorrectly structured filter:

`$BASE_URL/scim/v3/users?filter=bad%20filter`

Error Example:

```

HTTP/1.1 403 Forbidden
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "detail": "The filter 'bad filter' is not structured correctly."
}

```

Error Scenario: The caller attempts to with an invalid filter operator:

`$BASE_URL/scim/v3/users?filter=urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employeeNumber%20regex%20%221234%22`

Error Example:

```

HTTP/1.1 403 Forbidden
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "Detail": "The operator 'regex' is not supported."
}

```

Error Scenario: The caller attempts to with a filter value which is not enclosed in double quotes:

`$BASE_URL/scim/v3/users?filter=urn:ietf:params:scim:schemas:extension:enterprise:2.0:User:employeeNumber%20eq%201234`

Error Example:

```

HTTP/1.1 403 Forbidden
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "detail": "The value to be filtered '1234' is not valid."
}

```

Error Scenario: A request is made to this endpoint from an organization with multiple groups.

Example Error:

```

HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations with multiple groups."
}

```

Error Scenario: A request is made to this endpoint from an organization with no group.

Example Error:

```
HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations without groups."
}
```

Filtering

The current SCIM implementation supports a limited subset of filtering functionality. Filtering allows a user to pass in a filter as a query parameter to a supported endpoint. Any SCIM filtering option not specifically mentioned in this section is not currently supported.

Supported Filter Structure:

The SCIM implementation supports the following filter structure:

Schemald:attributeName operator "value"

Supported Schemalds and AttributeName:

We currently only support filtering on a single attribute at a time. The specified attribute must be supported and must belong to the specified schema. The current set of supported attributes and their associated schemas are:

Schema Id	Attribute Name
urn:ietf:params:scim:schemas:extension:enterprise:2.0:User	employeeNumber
urn:ietf:params:scim:schemas:extension:enterprise:2.0:User	userName

Supported Operators:

Operators are case insensitive. All other operators such as "regex", "co", "(,)" etc. are not supported and will cause undesired results or errors if attempted.

The current set of supported operators for this endpoint are:

Operator	Description
eq	Equals, exact match. When used in a filter, the 'eq' operator will match any objects which exactly match the filter value by querying the filter attribute within the specified filter schema. If the attribute contains multiple values, if any of the values match, the result will be returned.
pr	Present (has value). When used in a filter, the 'pr' operator will match any object which has a value for the specified attribute within the specified filter schema.

Supported Values:

The value must be enclosed in double quotes, an error will be returned if double quotes are missing. If an empty value is passed in eg "", no results will be returned.

This only applies to operators that accept a value.

Updating Users

Endpoint: **\$BASE_URL/scim/v3/users/{user_id}**
 HTTP Operation: **PUT**
 Authorization: **Bearer Token**

A HTTP PUT call to this endpoint can be used to update mutable attributes for a User. The intent of a PUT call is to replace the entire object rather than update individual fields. PUT calls cannot be used to create new resources. Only Users from the caller's organization can be updated.

If an attribute is immutable, the PUT call must contain the same value (or no value) as the existing object. A 400 bad request will be returned if modification of an immutable attribute is attempted.

If an attribute is mutable, supplying new value(s) will replace the old values. Supplying a null or empty value will remove the existing value if the attribute is optional and will result in an error if the attribute is mandatory. If no value currently exists for a mutable attribute, the new value(s) will be applied.

“writeOnly” attributes such as the User password cannot currently be modified via a PUT call. They should not be supplied and a 400 bad request error will be returned.

Filtering, sorting, and pagination are not supported.

Example Request

```
HTTP/1.1 PUT $BASE_URL/scim/v3/users/2819c223-7f76-453a-919d-413861904646
Authorization: Bearer xxx
Content-Type: application/scim+json
{
  "schemas": ["urn:ietf:params:scim:schemas:core:2.0:User"],
  "userName": "bjensen",
  "name": {
    "familyName": "Jensen-Smith",
    "givenName": "Barbara",
    "infix": "van"
  },
  "gender": "UNKNOWN",
  "exportTimelineEvents": true,
  "roles": [],
  "emails": [
    {
      "value": "bjensen@example.com"
    },
    {
      "value": "babs@jensen.org"
    }
  ]
}
```

Example Successful Response

```
{
  "schemas": ["urn:ietf:params:scim:schemas:core:2.0:User"],
  "id": "2819c223-7f76-453a-919d-413861904646",
  "userName": "bjensen",
  "name": {
    "familyName": "Jensen-Smith",
    "givenName": "Barbara",
    "infix": "van"
  },
  "gender": "UNKNOWN",
  "exportTimelineEvents": true,
  "emails": [
    {
      "value": "bjensen@example.com"
    },
    {
      "value": "babs@jensen.org"
    }
  ],
  "modules": [],
  "meta": {
    "resourceType": "User",
    "created": "2011-08-08T04:56:22Z",
    "lastModified": "2011-08-08T08:00:12Z",
    "location": "$BASE_URL/scim/v3/users/2819c223-7f76-453a-919d-413861904646"
  }
}
```

Example Error Response

Error Scenario: Attempting to update an attribute which must be unique to a value which already exists.

Example Error:

```
HTTP/1.1 409 Conflict
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "detail": "A User with the 'userName' 'bjensen' already exists.",
  "scimType": "uniqueness",
  "status": 409
}
```

Error Scenario: Attempting to pass in a JSON object which is not parseable.

Example Error:

```
HTTP/1.1 400 Bad Request
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
```

```

    "detail": "Request is unparsable, syntactically incorrect, or violates
schema.",
    "scimType": "invalidSyntax",
    "status": 400
}

```

Error Scenario: Caller attempts to clear the value of a mandatory attribute.

Example Error:

```

HTTP/1.1 400 Bad Request
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "detail": "A value for the 'userName' attribute is required.",
  "scimType": "mutability",
  "status": 400
}

```

Error Scenario: Caller attempts to update an attribute which is readOnly

Example Error:

```

HTTP/1.1 400 Bad Request
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "detail": "The 'id' attribute is readOnly.",
  "scimType": "mutability",
  "status": 400
}

```

Error Scenario: Caller attempts to update an attribute which is immutable.

Example Error:

```

HTTP/1.1 400 Bad Request
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "detail": "The 'discipline' attribute is immutable.",
  "scimType": "mutability",
  "status": 400
}

```

Error Scenario: Caller attempts to update an attribute to an invalid value.

Example Error:

HTTP/1.1 400 Bad Request

```
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "detail": "The value 'DETECTIVE' is not valid for the 'roles' attribute.",
  "scimType": "invalidValue",
  "status": 400
}
```

Error Scenario: Caller attempting to update a User which does not exist.

Example Error:

HTTP/1.1 404 Not Found

```
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "detail": "Resource '2819c223-7f76-453a-919d-413861904646' not found.",
  "status": 404
}
```

Error Scenario: A request is sent to this endpoint containing filtering options.

Error Example:

HTTP/1.1 403 FORBIDDEN

```
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "scimType": "invalidFilter",
  "detail": "Filtering of ResourceTypes is not supported."
}
```

Error Scenario: A request to this endpoint is made for an invalid organization.

Example Error:

HTTP/1.1 403 FORBIDDEN

```
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "detail": "Operation is not permitted based on the supplied authorization."
}
```

Error Scenario: A request is made to this endpoint from an organization with multiple groups.

Example Error:

HTTP/1.1 400 BAD REQUEST

```
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations with multiple groups."
}
```

Error Scenario: A request is made to this endpoint from an organization with no group.

Example Error:

```
HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations without groups."
}
```

Deleting Users

Endpoint: **\$BASE_URL/scim/v3/users/{user_id}**
 HTTP Operation: **DELETE**
 Authorization: **Bearer Token**

A HTTP DELETE call to this endpoint can be used to delete a Gerimedica User. Only Users from the caller's organization can be deleted. Attempting to delete a User which does not exist for the caller's organization will result in a 404 not found.

Filtering, sorting and pagination are not supported.

Example Request

```
HTTP/1.1 DELETE $BASE_URL/scim/v3/users/2819c223-7f76-453a-919d-413861904646
Authorization: Bearer xxx
```

Example Successful Response

```
HTTP/1.1 204 No Content
```

Example Error Responses

Error Scenario: Attempting to delete a User which does not exist.

Example Error:

```
HTTP/1.1 404 Not Found
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "detail": "Resource '2819c223-7f76-453a-919d-413861904646' not found.",
  "status": 404
}
```

Error Scenario: A request is sent to this endpoint containing filtering options.

Error Example:

```
HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "scimType": "invalidFilter",
  "detail": "Filtering of ResourceTypes is not supported."
}
```

Error Scenario: A request to this endpoint is made for an invalid organization.

Example Error:

```
HTTP/1.1 403 FORBIDDEN
{
```

```
"schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
"status": 403,
"detail": "Operation is not permitted based on the supplied authorization."
}
```

Error Scenario: A request is made to this endpoint from an organization with multiple groups.

Example Error:

```
HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations with multiple groups."
}
```

Error Scenario: A request is made to this endpoint from an organization with no group.

Example Error:

```
HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations without groups."
}
```

Scim Role Operations

Retrieving Roles

Endpoint: **\$BASE_URL/scim/v3/roles**
HTTP Operation: **GET**
Authorization: **Bearer Token**

An HTTP GET to this endpoint can be used to retrieve roles.

Example Request

```
HTTP/1.1 GET $BASE_URL/scim/v3/roles
Authorization: Bearer xxx
Content-Type: application/scim+json
```

Example Successful Response

```
"schemas": [
  "urn:ietf:params:scim:api:messages:2.0:ListResponse"
],
"totalResults": 1,
"Resources": [
  {
    "value": "39c0be3a-891a-4c77-a65e-49ee16f3538d",
    "displayName": "Diëtetiek"
  },
  {
    "value": "b3497775-9482-4e77-a240-8607db84fee5",
    "displayName": "Ergotherapie"
  },
  {
    "value": "ceb5c7bf-2092-4377-975a-7303af8e42a8",
    "displayName": "Geestelijke verzorging"
  },
  {
    "value": "135f9ab7-ec65-4777-95e5-49b5977c33c7",
    "displayName": "Muziektherapie"
  },
  {
    "value": "9c4ca071-eb2c-4767-a228-0150c727e3f4",
    "displayName": "Custom Role"
  },
]
```

Example Error Responses

Error Scenario: A request is made to this endpoint which contains filtering options will result in an error.

Example Error:

```
HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "scimType": "invalidFilter",
  "detail": "Filtering is not supported."
}
```

Error Scenario: A request is made to this endpoint from an organization with multiple groups.

Example Error:

```
HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations with multiple groups."
}
```

Error Scenario: A request is made to this endpoint from an organization with no group.

Example Error:

```
HTTP/1.1 400 BAD REQUEST
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 400,
  "detail": "SCIM is not currently supported for organizations without groups."
}
```

Error Scenario: A request to this endpoint is made for an invalid organization.

Example Error:

```
HTTP/1.1 403 FORBIDDEN
{
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:Error"],
  "status": 403,
  "detail": "Operation is not permitted based on the supplied authorization."
}
```