

DIGILAB API (1.0.0)

License: [MIT](#)

API for managing manuscript, 3D models, textures, annotations, and poses in DIGILAB.

3d-annotations

All 3D Annotation related operations

Retrieve all 3D annotations

AUTHORIZATIONS: > *bearerAuth*

QUERY PARAMETERS

page	integer <code>>= 0</code> Default: <code>0</code> Numero della pagina (facoltativo, predefinito: 0). L'indice parte da 0.
size	integer <code>>= 1</code> Default: <code>20</code> Numero di elementi per pagina (facoltativo, predefinito: 20). Deve essere maggiore di 0.
sort	string Example: <code>sort=name,asc</code> Campo per l'ordinamento (facoltativo). Formato: <code>campo,direzione</code> (es. <code>name,asc</code>). Direzioni supportate: <code>asc</code> , <code>desc</code> .

HEADER PARAMETERS

Accept	string Enum: <code>"application/json"</code> <code>"text/turtle"</code> Formato di risposta desiderato. Valori supportati: <code>application/json</code> , <code>text/turtle</code>
--------	---

Responses

✓ **200** List of all 3D annotations

RESPONSE SCHEMA: application/json ▾

Array [

region string
3D region where the annotation applies

annotation > object (Reference)
Una reference da una risorsa a un'altra

model > object (Reference)
Una reference da una risorsa a un'altra

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

]

— **401** Unauthorized

— **406** Formato non supportato

GET /3d-annotation

Response samples

200

Content type
application/json

Copy Expand all Collapse all

```
[  
- {  
    "region": "string",  
    + "annotation": { ... },  
    + "model": { ... },  
    "id": "string",  
    + "identifiers": [ ... ],  
    + "extensions": [ ... ]  
}  
]  
]
```

Create a new 3D annotation

AUTHORIZATIONS: > *bearerAuth*

REQUEST BODY SCHEMA: application/json

region	string 3D region where the annotation applies
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

— **201** 3D annotation successfully created

— **401** Unauthorized

POST /3d-annotation

Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

```
{  
    "region": "string",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Get a 3D annotation

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

annotationId
required string

Responses

✓ **200** 3D annotation retrieved successfully

RESPONSE SCHEMA: application/json

region	string 3D region where the annotation applies
annotation >	object (Annotation)
model >	object (3DModel)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— 401 Unauthorized

— 404 3D annotation not found

GET /3d-annotation/{annotationId}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "region": "string",
  - "annotation": {
      "type": "text",
      "content": "string",
      "id": "string",
      + "identifiers": [ ... ],
      + "extensions": [ ... ]
    },
  - "model": {
      "name": "string",
      + "textures": [ ... ],
      + "annotations": [ ... ],
      "id": "string",
      + "identifiers": [ ... ],
      + "extensions": [ ... ]
    },
    "id": "string",
  - "identifiers": [
      + { ... }
    ],
  - "extensions": [
      + { ... }
    ]
}
```

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

annotationId
required string

REQUEST BODY SCHEMA: application/json

region string
3D region where the annotation applies

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** 3D annotation replaced successfully

RESPONSE SCHEMA: application/json

region string
3D region where the annotation applies

annotation > object (Annotation)

model > object (3DModel)

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— **401** Unauthorized

— **404** 3D annotation not found

PUT /3d-annotation/{annotationId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "region": "string",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "region": "string",  
    - "annotation": {  
        "type": "text",  
        "content": "string",  
        "id": "string",  
        + "identifiers": [ ... ],  
        + "extensions": [ ... ]  
    - "model": {  
        "name": "string",  
        + "textures": [ ... ],  
        + "annotations": [ ... ],  
        "id": "string",  
        + "identifiers": [ ... ],  
        + "extensions": [ ... ]  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
}
```

```
- "extensions": [
    + { ... }
]
}
```

Update a 3D annotation

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

annotationId string
required

REQUEST BODY SCHEMA: application/json

region string
3D region where the annotation applies

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ 200 3D annotation updated successfully

RESPONSE SCHEMA: application/json

region string
3D region where the annotation applies

annotation > object (Annotation)

model > object (3DModel)

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions >

Array of objects (Extension)

— 401 Unauthorized

— 404 3D annotation not found

PATCH /3d-annotation/{annotationId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "region": "string",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "region": "string",  
    - "annotation": {  
        "type": "text",  
        "content": "string",  
        "id": "string",  
        + "identifiers": [ ... ],  
        + "extensions": [ ... ]  
    },  
}
```

```
- "model": {
    "name": "string",
    + "textures": [ ... ],
    + "annotations": [ ... ],
    "id": "string",
    + "identifiers": [ ... ],
    + "extensions": [ ... ]
},
"id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
```

Delete a 3D annotation

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

annotationId string
required

Responses

— **204** 3D annotation deleted successfully

— **401** Unauthorized

— **404** 3D annotation not found

DELETE /3d-annotation/{annotationId}

3d-annotations/{id}/annotations

All Annotations related to a 3D Annotations

Get annotations of a 3D annotation

returns a list of annotations with support for pagination, sorting and multiple formats.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

annotationId string
required

QUERY PARAMETERS

page integer `>= 0`
Default: `0`
Numero della pagina (facoltativo, predefinito: 0). L'indice parte da 0.

size integer `>= 1`
Default: `20`
Numero di elementi per pagina (facoltativo, predefinito: 20). Deve essere maggiore di 0.

sort string
Example: `sort=createdDate,desc`
Campo per l'ordinamento (facoltativo). Formato: `campo,direzione` (es. `createdDate,desc`). Direzioni supportate: `asc`, `desc`.

HEADER PARAMETERS

Accept string
Enum: `"application/json"`, `"text/turtle"`
Formato di risposta desiderato. Valori supportati: `application/json`, `text/turtle`.

Responses

✓ 200 Lista di annotazioni restituita con successo.

RESPONSE SCHEMA: application/json ▾

Array [

type string
 Enum: `"text"` `"image"` `"video"` `"audio"`
 Annotation type (text, image, video, audio)

content string
 Content of the annotation (text, image URL, video URL, audio URL)

id string
 Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

]

✗ **406** Formato non supportato

RESPONSE SCHEMA: application/json

error string

supported_formats Array of strings

GET /3d-annotation/{annotationId}/annotations

Response samples

200

406

Content type

application/json

Copy Expand all Collapse all

[
 - {
 "type": "text",
 "content": "string",
 "id": "string",
 + "identifiers": [...],
 + "extensions": [...]
 }

]

Add a new annotation to a 3D annotation

AUTHORIZATIONS:

bearerAuth

PATH PARAMETERS

annotationId
required string

REQUEST BODY SCHEMA:

application/json

type string
Enum: `"text"` `"image"` `"video"` `"audio"`
Annotation type (text, image, video, audio)

content string
Content of the annotation (text, image URL, video URL, audio URL)

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ 201 Annotation created successfully

RESPONSE SCHEMA:

application/json

type string
Enum: `"text"` `"image"` `"video"` `"audio"`
Annotation type (text, image, video, audio)

content string
Content of the annotation (text, image URL, video URL, audio URL)

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions >

Array of objects (Extension)

— 401 Unauthorized

POST /3d-annotation/{annotationId}/annotations

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "type": "text",  
  "content": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

201

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "type": "text",  
    "content": "string",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Get an annotation of a 3D annotation

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

3dAnnotationId string
required

annotationId string
required

Responses

✓ 200 Annotation retrieved successfully

RESPONSE SCHEMA: application/json

type string
Enum: "text" "image" "video" "audio"
Annotation type (text, image, video, audio)

content string
Content of the annotation (text, image URL, video URL, audio URL)

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— 401 Unauthorized

— 404 Annotation not found

GET /3d-annotation/{3dAnnotationId}/annotations/{annotationId}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "type": "text",
  "content": "string",
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

Replace an annotation of a 3D annotation

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

3dAnnotationId string
required

annotationId string
required

REQUEST BODY SCHEMA: application/json

type string
Enum: `"text"` `"image"` `"video"` `"audio"`
Annotation type (text, image, video, audio)

<code>content</code>	string Content of the annotation (text, image URL, video URL, audio URL)
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

Responses

✓ **200** Annotation replaced successfully

RESPONSE SCHEMA: `application/json`

<code>type</code>	string Enum: <code>"text"</code> <code>"image"</code> <code>"video"</code> <code>"audio"</code> Annotation type (text, image, video, audio)
<code>content</code>	string Content of the annotation (text, image URL, video URL, audio URL)
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

— **401** Unauthorized

— **404** Annotation not found

PUT /3d-annotation/{3dAnnotationId}/annotations/{annotationId}

Request samples

Payload

Content type
`application/json`

Copy Expand all Collapse all

```
{  
  "type": "text",  
  "content": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

200

Content type

application/json

Copy Expand all Collapse all

```
{  
  "type": "text",  
  "content": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Update an annotation of a 3D annotation

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

3dAnnotationId
required string

annotationId
required string

REQUEST BODY SCHEMA: application/json

type string
Enum: "text" "image" "video" "audio"
Annotation type (text, image, video, audio)

content string
Content of the annotation (text, image URL, video URL, audio URL)

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** Annotation updated successfully

RESPONSE SCHEMA: application/json

type string
Enum: "text" "image" "video" "audio"
Annotation type (text, image, video, audio)

content string
Content of the annotation (text, image URL, video URL, audio URL)

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— **401** Unauthorized

— **404** Annotation not found

PATCH /3d-annotation/{3dAnnotationId}/annotations/{annotationId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "type": "text",  
  "content": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

200**Content type**

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "type": "text",  
  "content": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Delete an annotation of a 3D annotation

AUTHORIZATIONS: >*bearerAuth***PATH PARAMETERS**

3dAnnotationId required	string
--	--------

annotationId	string
required	

Responses

— **204** Annotation deleted successfully

— **401** Unauthorized

— **404** Annotation not found

DELETE /3d-annotation/{3dAnnotationId}/annotations/{annotationId}

3d-annotations/{id}/3d-models

All 3D Models related operations

Get a 3D model of a 3D annotation

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

3dAnnotationId	string
required	

3dMoldeId	string
required	

Responses

✓ **200** 3D model retrieved successfully

RESPONSE SCHEMA: application/json

<code>name</code>	string Name of the 3D model
<code>textures ></code>	Array of objects (Texture) List of textures associated with the 3D model
<code>annotations ></code>	Array of objects (3DAnnotation) List of 3D annotations linked to the model
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

— **401** Unauthorized

— **404** 3D model not found

GET /3d-annotation/{3dAnnotationId}/3d-model/{3dMoldeId}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "name": "string",  
    - "textures": [  
        + { ... }  
    ],  
    - "annotations": [  
        + { ... }  
    ],  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Replace a 3D model of a 3D annotation

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

3dAnnotationId string
required

3dModelId string
required

REQUEST BODY SCHEMA: **application/json**

name string
Name of the 3D model

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** 3D model replaced successfully

<code>name</code>	string Name of the 3D model
<code>textures ></code>	Array of objects (Texture) List of textures associated with the 3D model
<code>annotations ></code>	Array of objects (3DAnnotation) List of 3D annotations linked to the model
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

— **401 Unauthorized**

— **404 3D model not found**

PUT /3d-annotation/{3dAnnotationId}/3d-model/{3dModelId}

Request samples

Payload

Content type
application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "name": "string",
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

Response samples

200

Content type

application/json

Copy Expand all Collapse all

```
{  
    "name": "string",  
    - "textures": [  
        + { ... }  
    ],  
    - "annotations": [  
        + { ... }  
    ],  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Update a 3D model of a 3D annotation

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

3dAnnotationId
required string

3dMoldeId
required string

REQUEST BODY SCHEMA: application/json

name string
Name of the 3D model

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** 3D model updated successfully

RESPONSE SCHEMA: application/json

name	string	Name of the 3D model
textures >	Array of objects (Texture)	List of textures associated with the 3D model
annotations >	Array of objects (3DAnnotation)	List of 3D annotations linked to the model
id	string	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)	
extensions >	Array of objects (Extension)	

— **401** Unauthorized

— **404** 3D model not found

PATCH /3d-annotation/{3dAnnotationId}/3d-model/{3dMoldeId}

Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

```
{  
  "name": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
}
```

```
- "extensions": [
    + { ... }
]
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
    "name": "string",
- "textures": [
    + { ... }
],
- "annotations": [
    + { ... }
],
    "id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Delete a 3D model of a 3D annotation

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

3dAnnotationId
required

string

3dMoldeId
required

string

Responses

— **204** 3D model deleted successfully

— **401** Unauthorized

— **404** 3D model not found

DELETE /3d-annotation/{3dAnnotationId}/3d-model/{3dMoldeId}

Annotations

All Annotation related operations

Get annotations

returns returns a list of annotations with support for pagination, sorting and multiple formats.

AUTHORIZATIONS: >

bearerAuth

QUERY PARAMETERS

page

integer `>= 0`

Default: `0`

Numero della pagina (facoltativo, predefinito: 0). L'indice parte da 0.

size

integer `>= 1`

Default: `20`

Numero di elementi per pagina (facoltativo, predefinito: 20). Deve essere maggiore di 0.

sort

string

Example: `sort=createdDate,desc`

Campo per l'ordinamento (facoltativo). Formato: `campo,direzione` (es. `createdDate,desc`). Direzioni supportate: `asc`, `desc`.

HEADER PARAMETERS

Accept

string

Enum: `"application/json"` `"text/turtle"`

Responses

- ✓ **200** Lista di annotazioni restituita con successo.

RESPONSE SCHEMA: application/json ▾

Array [

type string

 Enum: ["text"](#) ["image"](#) ["video"](#) ["audio"](#)

 Annotation type (text, image, video, audio)

content string

 Content of the annotation (text, image URL, video URL, audio URL)

id string

 Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

]

- ✗ **406** Formato non supportato

RESPONSE SCHEMA: application/json

error string

supported_formats Array of strings

GET /annotations

Response samples

[200](#)

[406](#)

Content type

application/json

```
[  
- {  
  "type": "text",  
  "content": "string",  
  "id": "string",  
  + "identifiers": [ ... ],  
  + "extensions": [ ... ]  
}  
]  
]
```

Create a new annotation

AUTHORIZATIONS: > *bearerAuth*

REQUEST BODY SCHEMA: application/json

type	string Enum: <code>"text"</code> <code>"image"</code> <code>"video"</code> <code>"audio"</code> Annotation type (text, image, video, audio)
content	string Content of the annotation (text, image URL, video URL, audio URL)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ 201 Annotation created successfully

RESPONSE SCHEMA: application/json

type	string Enum: <code>"text"</code> <code>"image"</code> <code>"video"</code> <code>"audio"</code> Annotation type (text, image, video, audio)
content	string Content of the annotation (text, image URL, video URL, audio URL)
id	string

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— 401 Unauthorized

POST /annotations

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "type": "text",  
  "content": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

201

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "type": "text",  
  "content": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

```
- "extensions": [
  + { ... }
]
}
```

Get an annotation

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id string
required

Responses

✓ **200** Annotation retrieved successfully

RESPONSE SCHEMA: application/json

type	string Enum: <code>"text"</code> <code>"image"</code> <code>"video"</code> <code>"audio"</code> Annotation type (text, image, video, audio)
<hr/>	
content	string Content of the annotation (text, image URL, video URL, audio URL)
<hr/>	
id	string Unique identifier for the object. It could not be send in POST
<hr/>	
identifiers >	Array of objects (Identifier)
<hr/>	
extensions >	Array of objects (Extension)

— **401** Unauthorized

— **404** Annotation not found

GET /annotations/{id}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "type": "text",  
    "content": "string",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Replace an annotation

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

id string
required

REQUEST BODY SCHEMA: application/json

type string
Enum: `"text"` `"image"` `"video"` `"audio"`
Annotation type (text, image, video, audio)

content string
Content of the annotation (text, image URL, video URL, audio URL)

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** Annotation replaced successfully

RESPONSE SCHEMA: application/json

type	string Enum: "text" "image" "video" "audio" Annotation type (text, image, video, audio)
content	string Content of the annotation (text, image URL, video URL, audio URL)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— **401** Unauthorized

— **404** Annotation not found

PUT /annotations/{id}

Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

```
{  
    "type": "text",  
    "content": "string",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "type": "text",  
    "content": "string",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Update an annotation

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id
required string

REQUEST BODY SCHEMA: application/json

type string
Enum: `"text"` `"image"` `"video"` `"audio"`
Annotation type (text, image, video, audio)

content	string Content of the annotation (text, image URL, video URL, audio URL)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ 200 Annotation updated successfully

RESPONSE SCHEMA: application/json

type	string Enum: <code>"text"</code> <code>"image"</code> <code>"video"</code> <code>"audio"</code> Annotation type (text, image, video, audio)
content	string Content of the annotation (text, image URL, video URL, audio URL)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— 401 Unauthorized

— 404 Annotation not found

PATCH /annotations/{id}

Request samples

Payload

Content type
application/json

Copy Expand all Collapse all

```
{  
  "type": "text",  
  "content": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

200

Content type

application/json

Copy Expand all Collapse all

```
{  
  "type": "text",  
  "content": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Delete an annotation

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id
required string

Responses

— **204** Annotation deleted successfully

— **401** Unauthorized

— **404** Annotation not found

DELETE /annotations/{id}

events

All Events related operations

Get all events

This operation retrieves all events.

AUTHORIZATIONS: > *bearerAuth*

Responses

✓ **200** A list of events

RESPONSE SCHEMA: `application/json`

Array [

type string

Period > object (Period)

Location > object (Location)

id string

 Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

`extensions >`

Array of objects (Extension)

]

— **401 Unauthorized**

GET /events

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
- {  
    "type": "string",  
    + "Period": { ... },  
    + "Location": { ... },  
    "id": "string",  
    + "identifiers": [ ... ],  
    + "extensions": [ ... ]  
}  
]
```

Create a new event

This operation creates a new event.

AUTHORIZATIONS: >

bearerAuth

REQUEST BODY SCHEMA: application/json

`type`

string

`Period >`

object (Period)

`Location >`

object (Location)

<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

Responses

✓ 201 Event created

RESPONSE SCHEMA: `application/json`

<code>type</code>	string
<code>Period ></code>	object (Period)
<code>Location ></code>	object (Location)
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

— 401 Unauthorized

POST /events

Request samples

Payload

Content type
`application/json`

Copy Expand all Collapse all

```
{  
  "type": "string",
```

```
- "Period": {
    "start": "2019-08-24T14:15:22Z",
    "end": "2019-08-24T14:15:22Z"
},
- "Location": {
    "name": "string",
    "description": "string",
    "address": "string",
    + "coordinates": [ ... ],
    "id": "string",
    + "identifiers": [ ... ],
    + "extensions": [ ... ]
},
"id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
```

}

Response samples

201

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "type": "string",
- "Period": {
    "start": "2019-08-24T14:15:22Z",
    "end": "2019-08-24T14:15:22Z"
},
- "Location": {
    "name": "string",
    "description": "string",
    "address": "string",
    + "coordinates": [ ... ],
    "id": "string",
    + "identifiers": [ ... ],
    + "extensions": [ ... ]
},
"id": "string",
- "identifiers": [
    + { ... }
],
```

```
- "extensions": [
    + { ... }
]
}
```

Get an event

This operation retrieves an event based on its ID.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id
required string

Responses

✓ 200 An event

RESPONSE SCHEMA: application/json

type	string
Period >	object (Period)
Location >	object (Location)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— 401 Unauthorized

— 404 Event not found

GET /events/{id}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "type": "string",  
    - "Period": {  
        "start": "2019-08-24T14:15:22Z",  
        "end": "2019-08-24T14:15:22Z"  
    },  
    - "Location": {  
        "name": "string",  
        "description": "string",  
        "address": "string",  
        + "coordinates": [ ... ],  
        "id": "string",  
        + "identifiers": [ ... ],  
        + "extensions": [ ... ]  
    },  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Update an event

This operation updates an event based on its ID.

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

id required	string
------------------------------	--------

REQUEST BODY SCHEMA: application/json

type	string
Period >	object (Period)
Location >	object (Location)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ 200 Event updated

RESPONSE SCHEMA: application/json

type	string
Period >	object (Period)
Location >	object (Location)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— 401 Unauthorized

— 404 Event not found

PATCH /events/{id}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "type": "string",  
    - "Period": {  
        "start": "2019-08-24T14:15:22Z",  
        "end": "2019-08-24T14:15:22Z"  
    },  
    - "Location": {  
        "name": "string",  
        "description": "string",  
        "address": "string",  
        + "coordinates": [ ... ],  
        "id": "string",  
        + "identifiers": [ ... ],  
        + "extensions": [ ... ]  
    },  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Response samples

200**Content type**

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "type": "string",  
    - "Period": {  
        "start": "2019-08-24T14:15:22Z",  
        "end": "2019-08-24T14:15:22Z"  
    },  
}
```

```
- "Location": {  
    "name": "string",  
    "description": "string",  
    "address": "string",  
    + "coordinates": [ ... ],  
    "id": "string",  
    + "identifiers": [ ... ],  
    + "extensions": [ ... ]  
},  
"id": "string",  
- "identifiers": [  
    + { ... }  
],  

```

Delete an event

This operation deletes an event based on its ID.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

<code>id</code> <small>required</small>	<code>string</code>
--	---------------------

Responses

— **204** Event deleted

— **401** Unauthorized

— **404** Event not found

DELETE /events/{id}

Replace an event

This operation replaces an event based on its ID.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id
required string

REQUEST BODY SCHEMA: application/json

type string
Period > object (Period)
Location > object (Location)
id string
Unique identifier for the object. It could not be send in POST
identifiers > Array of objects (Identifier)
extensions > Array of objects (Extension)

Responses

✓ 200 Event replaced

RESPONSE SCHEMA: application/json

type string
Period > object (Period)
Location > object (Location)
id string
Unique identifier for the object. It could not be send in POST
identifiers > Array of objects (Identifier)
extensions > Array of objects (Extension)

— 401 Unauthorized

— 404 Event not found

PUT /events/{id}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "type": "string",
  - "Period": {
      "start": "2019-08-24T14:15:22Z",
      "end": "2019-08-24T14:15:22Z"
    },
  - "Location": {
      "name": "string",
      "description": "string",
      "address": "string",
      + "coordinates": [ ... ],
      "id": "string",
      + "identifiers": [ ... ],
      + "extensions": [ ... ]
    },
    "id": "string",
  - "identifiers": [
      + { ... }
    ],
  - "extensions": [
      + { ... }
    ]
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "type": "string",  
    - "Period": {  
        "start": "2019-08-24T14:15:22Z",  
        "end": "2019-08-24T14:15:22Z"  
    },  
    - "Location": {  
        "name": "string",  
        "description": "string",  
        "address": "string",  
        + "coordinates": [ ... ],  
        "id": "string",  
        + "identifiers": [ ... ],  
        + "extensions": [ ... ]  
    },  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

events/{id}/locations

All location related to event

Get Locations of an Event

Get locations of an Event.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

eventId required	string
-----------------------------------	--------

QUERY PARAMETERS

<code>id</code>	Array of strings Filtrare le location per ID, valori multipli separati da virgola ad es. <code>1,2,3</code>
<code>name</code>	string Filtrare le location per nome
<code>address</code>	string Filtrare le location per indirizzo
<code>coordinates</code>	Array of numbers Filtrare le location per coordinate, valori separati da virgola ad es. <code>1.0,2.0</code>
<code>identifiers</code>	Array of strings Filtrare i manoscritti per identificatore, valori multipli separati da virgola ad es. <code>1,2,3</code>
<code>page</code>	integer <code>>= 0</code> Default: <code>0</code> Numero della pagina (facoltativo, predefinito: 0). L'indice parte da 0.
<code>size</code>	integer <code>>= 1</code> Default: <code>20</code> Numero di elementi per pagina (facoltativo, predefinito: 20). Deve essere maggiore di 0.
<code>sort</code>	string Example: <code>sort=name,asc</code> Campo per l'ordinamento (facoltativo). Formato: <code>campo,direzione</code> (es. <code>name,asc</code>). Direzioni supportate: <code>asc</code> , <code>desc</code> .

HEADER PARAMETERS

<code>Accept</code>	string Enum: <code>"application/json"</code> <code>"text/turtle"</code> Formato di risposta desiderato. Valori supportati: <code>application/json</code> , <code>text/turtle</code>
---------------------	---

Responses

✓ 200 Successful operation

RESPONSE SCHEMA: `application/json`

Array [

`name` string

`description` string

`address` string

coordinates	Array of numbers	= 2 items
id	string	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)	
extensions >	Array of objects (Extension)	
]		

— 401 Unauthorized

GET /events/{eventId}/locations

Response samples

200

Content type
application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
- {  
  "id": "1",  
  "name": "Location 1",  
  "description": "Description 1",  
  "address": "Address 1",  
  + "coordinates": [ ... ]  
},  
- {  
  "id": "2",  
  "name": "Location 2",  
  "description": "Description 2",  
  "address": "Address 2",  
  + "coordinates": [ ... ]  
}  
]
```

Add Location to Event

Add a location to an event

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

eventId
required string

REQUEST BODY SCHEMA: application/json

name string

description string

address string

coordinates Array of numbers = 2 items

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

— 201 Location created successfully

— 401 Unauthorized

POST /events/{eventId}/locations

Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

```
{  
  "name": "string",  
  "description": "string",  
  "address": "string",  
  "coordinates": [
```

```
- "coordinates": [
  0,
  0
],
"id": "string",
- "identifiers": [
  + { ... }
],
- "extensions": [
  + { ... }
]
}
```

Get Location of an Event

Get a specific location by its ID

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

eventId
required string

locationId
required string

Responses

✓ 200 Successful operation

RESPONSE SCHEMA: application/json

name string

description string

address string

coordinates Array of numbers [= 2 items]

id string

Unique identifier for the object. It could not be send in POST

`identifiers` > Array of objects (Identifier)

`extensions` > Array of objects (Extension)

— **401 Unauthorized**

— **404 Location not found**

GET /events/{eventId}/locations/{locationId}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "name": "string",  
    "description": "string",  
    "address": "string",  
    - "coordinates": [  
        0,  
        0  
    ],  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Update Location of an Event

Update a location of an event with new information

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

eventId string
required

locationId string
required

REQUEST BODY SCHEMA: application/json

name string

description string

address string

coordinates Array of numbers **= 2 items**

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** Location updated successfully

RESPONSE SCHEMA: application/json

name string

description string

address string

coordinates Array of numbers **= 2 items**

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— **401** Unauthorized

— **404** Location not found

```
PUT /events/{eventId}/locations/{locationId}
```

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "name": "string",
  "description": "string",
  "address": "string",
  - "coordinates": [
    0,
    0
  ],
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "name": "string",
  "description": "string",
  "address": "string",
  - "coordinates": [
    0,
    0
  ],
  "id": "string",
```

```
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Patch Location of an Event

partial update a location with new information

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

eventId
required string

locationId
required string

REQUEST BODY SCHEMA: **application/json**

name string

description string

address string

coordinates Array of numbers **= 2 items**

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** Location patched successfully

RESPONSE SCHEMA: **application/json**

<code>name</code>	string
<code>description</code>	string
<code>address</code>	string
<code>coordinates</code>	Array of numbers <small>= 2 items</small>
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

— **401 Unauthorized**

— **404 Location not found**

PATCH /events/{eventId}/locations/{locationId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "name": "string",
  "description": "string",
  "address": "string",
  - "coordinates": [
    0,
    0
  ],
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "name": "string",  
    "description": "string",  
    "address": "string",  
    - "coordinates": [  
        0,  
        0  
    ],  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Delete Location of an Event

Delete a location of an event. This operation deletes a specific location by its ID.

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

eventId
required string

locationId
required string

Responses

— **204** Location deleted successfully

— **401** Unauthorized

DELETE /events/{eventId}/locations/{locationId}

Illuminated Scene

All Illuminated Scene related operations

Retrieve all illuminated manuscript scenes

AUTHORIZATIONS: > *bearerAuth*

QUERY PARAMETERS

<code>id</code>	Array of strings Filtra illuminated manuscript scenes per ID, valori multipli separati da virgola ad es. <code>1,2,3</code>
<code>title</code>	string Filtra illuminated manuscript scenes per titolo
<code>description</code>	string Filtra illuminated manuscript scenes per descrizione
<code>status</code>	string Enum: <code>"draft"</code> <code>"in_review"</code> <code>"revisions_required"</code> <code>"accepted"</code> <code>"in_editing"</code> <code>"published"</code> <code>"rejected"</code> <code>"withdrawn"</code> Filtra illuminated manuscript scenes in base al loro stato: <ul style="list-style-type: none">• <code>draft</code> : Bozza• <code>in_review</code> : In revisione• <code>revisions_required</code> : Revisione necessaria• <code>accepted</code> : Accettato• <code>in_editing</code> : In fase di editing• <code>published</code> : Pubblicato• <code>rejected</code> : Rifiutato• <code>withdrawn</code> : Ritirato
<code>locations</code>	string

locationId	Array of strings Filtra illuminated manuscript scenes per location ID, valori multipli separati da virgola ad es. <code>1,2,3</code>
authors	string Filtra illuminated manuscript scenes per nome autore
authorId	Array of strings filtra illuminated manuscript scenes per autore ID, valori multipli separati da virgola ad es. <code>1,2,3</code>
editors	string Filtra illuminated manuscript scenes per nome editor
editorId	Array of strings filtra illuminated manuscript scenes per editor ID, valori multipli separati da virgola ad es. <code>1,2,3</code>
identifiers	Array of strings Filtra illuminated manuscript scenes per identificatore, valori multipli separati da virgola ad es. <code>1,2,3</code>
page	integer <code>>= 0</code> Default: <code>0</code> Numero della pagina (facoltativo, predefinito: 0). L'indice parte da 0.
size	integer <code>>= 1</code> Default: <code>20</code> Numero di elementi per pagina (facoltativo, predefinito: 20). Deve essere maggiore di 0.
sort	string Example: <code>sort=createdDate,desc</code> Campo per l'ordinamento (facoltativo). Formato: <code>campo,direzione</code> (es. <code>createdDate,desc</code>). Direzioni supportate: <code>asc</code> , <code>desc</code> .

HEADER PARAMETERS

Accept	string Enum: <code>"application/json"</code> <code>"text/turtle"</code> Formato di risposta desiderato. Valori supportati: <code>application/json</code> , <code>text/turtle</code> .
--------	---

Responses

✓ 200 Successful response

Array [

title	string Title of the scene
description	string Description of the scene
panorama	string <uri> Panoramic view of the scene
previewImage	string <uri> Preview image of the scene
authors >	Array of objects (Reference) List of authors of the scene
editors >	Array of objects (Reference) List of editors
visibility	string Enum: <input type="button" value="public"/> <input type="button" value="private"/> <input type="button" value="restricted"/> Visibility status of the scene
project >	object (Reference) Una reference da una risorsa a un'altra
manuscript >	object (Reference) Una reference da una risorsa a un'altra
poses >	Array of objects (Reference) List of poses
annotations3D >	Array of objects (Reference) List of 3D annotations

]

— **401** Unauthorized— **406** Formato non supportato

GET /illuminated-scenes

Response samples

200

Content type

application/json

Copy Expand all Collapse all

```
[  
- {  
    "title": "string",  
    "description": "string",  
    "panorama": "http://example.com",  
    "previewImage": "http://example.com",  
+   "authors": [ ... ],  
+   "editors": [ ... ],  
    "visibility": "public",  
+   "project": { ... },  
+   "manuscript": { ... },  
+   "poses": [ ... ],  
+   "annotations3D": [ ... ]  
}  
]
```

Create a new illuminated manuscript scene

AUTHORIZATIONS: > *bearerAuth*

REQUEST BODY SCHEMA: application/json

title	string Title of the scene
description	string Description of the scene
panorama	string <uri> Panoramic view of the scene
previewImage	string <uri> Preview image of the scene
authors >	Array of objects (Reference) List of authors of the scene
editors >	Array of objects (Reference) List of editors

<code>visibility</code>	string Enum: "public" "private" "restricted" Visibility status of the scene
<code>project ></code>	object (Reference) Una reference da una risorsa a un'altra
<code>manuscript ></code>	object (Reference) Una reference da una risorsa a un'altra
<code>poses ></code>	Array of objects (Reference) List of poses
<code>annotations3D ></code>	Array of objects (Reference) List of 3D annotations

Responses

— **201** Scene created successfully

— **401** Unauthorized

POST /illuminated-scenes

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "title": "string",
  "description": "string",
  "panorama": "http://example.com",
  "previewImage": "http://example.com",
  - "authors": [
    + { ... }
  ],
  - "editors": [
    + { ... }
  ],
  "visibility": "public",
```

```
- "project": {
    "reference": "Risorsa/1234",
    "type": "Risorsa",
    + "identifier": { ... },
    "display": "descrizione della risorsa"
},
- "manuscript": {
    "reference": "Risorsa/1234",
    "type": "Risorsa",
    + "identifier": { ... },
    "display": "descrizione della risorsa"
},
- "poses": [
    + { ... }
],
- "annotations3D": [
    + { ... }
]
}
```

Retrieve a specific illuminated manuscript scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id
required string

HEADER PARAMETERS

Accept string
Enum: `"application/json"` `"text/turtle"`
Formato di risposta desiderato. Valori supportati: `application/json`,
`text/turtle`.

Responses

✓ **200** Successful response

RESPONSE SCHEMA: `application/json` ▾

Array [

title string

Title of the scene

description	string
	Description of the scene
panorama	string <uri>
	Panoramic view of the scene
previewImage	string <uri>
	Preview image of the scene
authors >	Array of objects (Reference) List of authors of the scene
editors >	Array of objects (Reference) List of editors
visibility	string Enum: <code>"public"</code> <code>"private"</code> <code>"restricted"</code> Visibility status of the scene
project >	object (Reference) Una reference da una risorsa a un'altra
manuscript >	object (Reference) Una reference da una risorsa a un'altra
poses >	Array of objects (Pose) List of poses
annotations3D >	Array of objects (3DAnnotation) List of 3D annotations

]

— **401** Unauthorized

— **404** Scene not found

— **406** Formato non supportato

GET /illuminated-scenes/{id}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
- {  
    "title": "string",  
    "description": "string",  
    "panorama": "http://example.com",  
    "previewImage": "http://example.com",  
    + "authors": [ ... ],  
    + "editors": [ ... ],  
    "visibility": "public",  
    + "project": { ... },  
    + "manuscript": { ... },  
    + "poses": [ ... ],  
    + "annotations3D": [ ... ]  
}  
]
```

Update an entire illuminated manuscript scene

AUTHORIZATIONS: >*bearerAuth***PATH PARAMETERS****id**
required

string

REQUEST BODY SCHEMA: application/json**title**

string

Title of the scene

description

string

Description of the scene

panorama

string <uri>

Panoramic view of the scene

previewImage

string <uri>

Preview image of the scene

authors >

Array of objects (Reference)

List of authors of the scene

editors >

Array of objects (Reference)

List of editors

visibility	string Enum: "public" "private" "restricted" Visibility status of the scene
project >	object (Reference) Una reference da una risorsa a un'altra
manuscript >	object (Reference) Una reference da una risorsa a un'altra
poses >	Array of objects (Reference) List of poses
annotations3D >	Array of objects (Reference) List of 3D annotations

Responses

— **200** Scene updated successfully

— **401** Unauthorized

— **404** Scene not found

PUT /illuminated-scenes/{id}

Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

```
{  
  "title": "string",  
  "description": "string",  
  "panorama": "http://example.com",  
  "previewImage": "http://example.com",  
  - "authors": [  
    + { ... }  
  ],
```

```
- "editors": [
    + { ... }
],
"visibility": "public",
- "project": {
    "reference": "Risorsa/1234",
    "type": "Risorsa",
    + "identifier": { ... },
    "display": "descrizione della risorsa"
},
- "manuscript": {
    "reference": "Risorsa/1234",
    "type": "Risorsa",
    + "identifier": { ... },
    "display": "descrizione della risorsa"
},
- "poses": [
    + { ... }
],
- "annotations3D": [
    + { ... }
]
```

Partially update an illuminated manuscript scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id
required string

REQUEST BODY SCHEMA: application/json

title string
Title of the scene

description string
Description of the scene

panorama string <uri>
Panoramic view of the scene

previewImage string <uri>
Preview image of the scene

<code>authors ></code>	Array of objects (Reference) List of authors of the scene
<code>editors ></code>	Array of objects (Reference) List of editors
<code>visibility</code>	string Enum: <code>"public"</code> <code>"private"</code> <code>"restricted"</code> Visibility status of the scene
<code>project ></code>	object (Reference) Una reference da una risorsa a un'altra
<code>manuscript ></code>	object (Reference) Una reference da una risorsa a un'altra
<code>poses ></code>	Array of objects (Reference) List of poses
<code>annotations3D ></code>	Array of objects (Reference) List of 3D annotations

Responses

— **200** Scene patched successfully

— **401** Unauthorized

— **404** Scene not found

PATCH /illuminated-scenes/{id}

Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

```
{
  "title": "string",
  "description": "string",
  "panorama": "http://example.com",
```

```
"previewImage": "http://example.com",  
- "authors": [  
    + { ... }  
,  
- "editors": [  
    + { ... }  
,  
    "visibility": "public",  
- "project": {  
    "reference": "Risorsa/1234",  
    "type": "Risorsa",  
    + "identifier": { ... },  
    "display": "descrizione della risorsa"  
},  
- "manuscript": {  
    "reference": "Risorsa/1234",  
    "type": "Risorsa",  
    + "identifier": { ... },  
    "display": "descrizione della risorsa"  
},  
- "poses": [  
    + { ... }  
,  
- "annotations3D": [  
    + { ... }  
]  
}
```

Delete an illuminated manuscript scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id
required string

Responses

— **204** Scene deleted successfully

— **401** Unauthorized

```
DELETE /illuminated-scenes/{id}
```

3d annotations of an Illuminated Scene

3D Annotations related to Illuminated Scene

Get 3D annotations of an illuminated scene

returns a list of 3D annotations with support for pagination, sorting and multiple formats.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId string
required

QUERY PARAMETERS

page integer `>= 0`
Default: `0`
Numero della pagina (facoltativo, predefinito: 0). L'indice parte da 0.

size integer `>= 1`
Default: `20`
Numero di elementi per pagina (facoltativo, predefinito: 20). Deve essere maggiore di 0.

sort string
Example: `sort=createdDate,desc`
Campo per l'ordinamento (facoltativo). Formato: `campo,direzione` (es. `createdDate,desc`). Direzioni supportate: `asc`, `desc`.

HEADER PARAMETERS

Accept string
Enum: `"application/json"` `"text/turtle"`

Responses

- ✓ **200** Lista di annotazioni 3D restituita con successo.

RESPONSE SCHEMA: [application/json](#) ▾

Array [

region	string
	3D region where the annotation applies
annotation >	object (Annotation)
model >	object (3DModel)
id	string
	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

]

- ✗ **406** Formato non supportato

RESPONSE SCHEMA: [application/json](#)

error	string
supported_formats	Array of strings

GET /illuminated-scenes/{illuminatedSceneId}/3d-annotations

Response samples

[200](#) [406](#)

Content type

[application/json](#)

```
[  
  - {  
      "region": "string",  
      + "annotation": { ... },  
      + "model": { ... },  
      "id": "string",  
      + "identifiers": [ ... ],  
      + "extensions": [ ... ]  
    }  
]
```

Add a new 3D annotation to an illuminated scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId string
required

REQUEST BODY SCHEMA: application/json

region string
3D region where the annotation applies

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ 201 3D annotation created successfully

RESPONSE SCHEMA: application/json

region string
3D region where the annotation applies

annotation > object (Annotation)

model > object (3DModel)

<code>id</code>	string
	Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

— 401 Unauthorized

POST /illuminated-scenes/{illuminatedSceneId}/3d-annotations

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "region": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

201

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "region": "string",  
}
```

```
- "annotation": {
    "type": "text",
    "content": "string",
    "id": "string",
    + "identifiers": [ ... ],
    + "extensions": [ ... ]
},
- "model": {
    "name": "string",
    + "textures": [ ... ],
    + "annotations": [ ... ],
    "id": "string",
    + "identifiers": [ ... ],
    + "extensions": [ ... ]
},
"id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
```

```
}
```

Get a 3D annotation of an illuminated scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId string
required

3dAnnotationId string
required

Responses

✓ 200 3D annotation retrieved successfully

RESPONSE SCHEMA: application/json

region string
3D region where the annotation applies

annotation >	object (Annotation)
model >	object (3DModel)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— **401 Unauthorized**

— **404 3D annotation not found**

GET /illuminated-scenes/{illuminatedSceneId}/3d-annotations/{3dAnnotationId}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "region": "string",
  - "annotation": {
      "type": "text",
      "content": "string",
      "id": "string",
      + "identifiers": [ ... ],
      + "extensions": [ ... ]
    },
  - "model": {
      "name": "string",
      + "textures": [ ... ],
      + "annotations": [ ... ],
      "id": "string",
      + "identifiers": [ ... ],
      + "extensions": [ ... ]
    },
  "id": "string",
}
```

```
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Replace a 3D annotation of an illuminated scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId string
required

3dAnnotationId string
required

REQUEST BODY SCHEMA: application/json

region string
3D region where the annotation applies

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** 3D annotation replaced successfully

RESPONSE SCHEMA: application/json

region string
3D region where the annotation applies

annotation > object (Annotation)

model > object (3DModel)

<code>id</code>	string
	Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

— **401 Unauthorized**

— **404 3D annotation not found**

PUT /illuminated-scenes/{illuminatedSceneId}/3d-annotations/{3dAnnotationId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "region": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "region": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

```
- "annotation": {
    "type": "text",
    "content": "string",
    "id": "string",
    + "identifiers": [ ... ],
    + "extensions": [ ... ]
},
- "model": {
    "name": "string",
    + "textures": [ ... ],
    + "annotations": [ ... ],
    "id": "string",
    + "identifiers": [ ... ],
    + "extensions": [ ... ]
},
"id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
```

Update a 3D annotation of an illuminated scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

`illuminatedSceneId` string
required

`3dAnnotationId` string
required

REQUEST BODY SCHEMA: `application/json`

`region` string
3D region where the annotation applies

`id` string
Unique identifier for the object. It could not be send in POST

`identifiers` > Array of objects (Identifier)

`extensions` > Array of objects (Extension)

Responses

✓ **200** 3D annotation updated successfully

RESPONSE SCHEMA: application/json

region	string	3D region where the annotation applies
annotation >	object (Annotation)	
model >	object (3DModel)	
id	string	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)	
extensions >	Array of objects (Extension)	

— **401** Unauthorized

— **404** 3D annotation not found

PATCH /illuminated-scenes/{illuminatedSceneId}/3d-annotations/{3dAnnotationId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "region": "string",
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
    "region": "string",
    - "annotation": {
        "type": "text",
        "content": "string",
        "id": "string",
        + "identifiers": [ ... ],
        + "extensions": [ ... ]
    },
    - "model": {
        "name": "string",
        + "textures": [ ... ],
        + "annotations": [ ... ],
        "id": "string",
        + "identifiers": [ ... ],
        + "extensions": [ ... ]
    },
    "id": "string",
    - "identifiers": [
        + { ... }
    ],
    - "extensions": [
        + { ... }
    ]
}
```

Delete a 3D annotation of an illuminated scene

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

illuminatedSceneId string
required

3dAnnotationId
required

string

Responses

— **204** 3D annotation deleted successfully

— **401** Unauthorized

— **404** 3D annotation not found

DELETE /illuminated-scenes/{illuminatedSceneId}/3d-annotations/{3dAnnotationId}

Poses of an Illuminated Scene

Poses related to Illuminated Scene

Get poses of an illuminated scene

returns a list of poses with support for pagination, sorting and multiple formats.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId
required

string

QUERY PARAMETERS

page
integer
Default:

Numero della pagina (facoltativo, predefinito: 0). L'indice parte da 0.

size
integer
Default:

Numero di elementi per pagina (facoltativo, predefinito: 20). Deve essere maggiore di 0.

sort

string

Example: `sort=createdDate,desc`

Campo per l'ordinamento (facoltativo). Formato: `campo,direzione` (es. `createdDate,desc`). Direzioni supportate: `asc`, `desc`.

HEADER PARAMETERS

Accept

string

Enum: `"application/json"` `"text/turtle"`

Formato di risposta desiderato. Valori supportati: `application/json`, `text/turtle`.

Responses

✓ **200** Lista di pose restituita con successo.

RESPONSE SCHEMA: `application/json` ▾

Array [

name

string

Name of the pose

3d-models >

Array of objects (3DModel)

List of 3D models associated with the pose

contents >

Array of objects (LighContent)

id

string

Unique identifier for the object. It could not be send in POST

identifiers >

Array of objects (Identifier)

extensions >

Array of objects (Extension)

]

✗ **406** Formato non supportato

RESPONSE SCHEMA: `application/json`

error

string

supported_formats

Array of strings

GET /illuminated-scenes/{illuminatedSceneId}/poses

Response samples

200

406

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
- {  
    "name": "string",  
    + "3d-models": [ ... ],  
    + "contents": [ ... ],  
    "id": "string",  
    + "identifiers": [ ... ],  
    + "extensions": [ ... ]  
}  
]
```

Add a new pose to an illuminated

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId string
required

REQUEST BODY SCHEMA: application/json

name string
Name of the pose

3d-models > Array of objects (3DModel)
List of 3D models associated with the pose

contents > Array of objects (LighContent)

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions >

Array of objects (Extension)

Responses

✓ 201 Pose created successfully

RESPONSE SCHEMA: application/json

name string

Name of the pose

3d-models > Array of objects (3DModel)

List of 3D models associated with the pose

contents > Array of objects (LighContent)

id string

Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— 401 Unauthorized

POST /illuminated-scenes/{illuminatedSceneId}/poses

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "name": "string",  
  - "3d-models": [  
    + { ... }  
  ],  
  - "contents": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

```
"id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Response samples

201

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
    "name": "string",
- "3d-models": [
    + { ... }
],
- "contents": [
    + { ... }
],
    "id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Get a pose of an illuminated scene

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

illuminatedSceneId required	string
poseId required	string

Responses

✓ **200** Pose retrieved successfully

RESPONSE SCHEMA: application/json

name	string
	Name of the pose
3d-models >	Array of objects (3DModel) List of 3D models associated with the pose
contents >	Array of objects (LighContent)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— **401** Unauthorized

— **404** Pose not found

GET /illuminated-scenes/{illuminatedSceneId}/poses/{poseId}

Response samples

200

Content type
application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "name": "string",  
  - "3d-models": [  
    + { ... }  
  ],  
  - "contents": [  
    + { ... }  
  ],  
  "id": "string",
```

```
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Replace a pose of an illuminated scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId string
required

poseId string
required

REQUEST BODY SCHEMA: application/json

name	string
	Name of the pose
3d-models >	Array of objects (3DModel) List of 3D models associated with the pose
contents >	Array of objects (LighContent)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ 200 Pose replaced successfully

RESPONSE SCHEMA: application/json

name	string
	Name of the pose

<code>3d-models ></code>	Array of objects (3DModel) List of 3D models associated with the pose
<code>contents ></code>	Array of objects (LighContent)
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

— **401 Unauthorized**

— **404 Pose not found**

PUT /illuminated-scenes/{illuminatedSceneId}/poses/{poseId}

Request samples

Payload

Content type
application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "name": "string",
  - "3d-models": [
    + { ... }
  ],
  - "contents": [
    + { ... }
  ],
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

Response samples

200

Content type

application/json

Copy Expand all Collapse all

```
{  
    "name": "string",  
    - "3d-models": [  
        + { ... }  
    ],  
    - "contents": [  
        + { ... }  
    ],  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Update a pose of an illuminated scene

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

illuminatedSceneId string
required

poseId string
required

REQUEST BODY SCHEMA: application/json

name string
Name of the pose

3d-models > Array of objects (3DModel)
List of 3D models associated with the pose

contents > Array of objects (LighContent)

id string
Unique identifier for the object. It could not be send in POST

`identifiers >` Array of objects (Identifier)

`extensions >` Array of objects (Extension)

Responses

✓ **200** Pose updated successfully

RESPONSE SCHEMA: `application/json`

`name` string

Name of the pose

`3d-models >` Array of objects (3DModel)

List of 3D models associated with the pose

`contents >` Array of objects (LighContent)

`id` string

Unique identifier for the object. It could not be send in POST

`identifiers >` Array of objects (Identifier)

`extensions >` Array of objects (Extension)

— **401** Unauthorized

— **404** Pose not found

PATCH /illuminated-scenes/{illuminatedSceneId}/poses/{poseId}

Request samples

Payload

Content type

`application/json`

Copy Expand all Collapse all

{

`"name": "string",`

```
- "3d-models": [
    + { ... }
],
- "contents": [
    + { ... }
],
"id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "name": "string",
- "3d-models": [
    + { ... }
],
- "contents": [
    + { ... }
],
"id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Delete a pose of an illuminated scene

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

`illuminatedSceneId` string
required

`poseId` string
required

Responses

— **204** Pose deleted successfully

— **401** Unauthorized

— **404** Pose not found

DELETE /illuminated-scenes/{illuminatedSceneId}/poses/{poseId}

Authors of an Illuminated Scene

Authors related to Illuminated Scene

Get authors of an illuminated scene

returns a list of authors with support for pagination, sorting and multiple formats.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

`illuminatedSceneId` string
required

QUERY PARAMETERS

`page` integer `>= 0`
Default: `0`

Numero della pagina (facoltativo, predefinito: 0). L'indice parte da 0.

size integer `>= 1`
Default: 20
Numero di elementi per pagina (facoltativo, predefinito: 20). Deve essere maggiore di 0.

sort string
Example: `sort=createdDate,desc`
Campo per l'ordinamento (facoltativo). Formato: `campo,direzione` (es. `createdDate,desc`). Direzioni supportate: `asc`, `desc`.

HEADER PARAMETERS

Accept string
Enum: `"application/json"` `"text/turtle"`
Formato di risposta desiderato. Valori supportati: `application/json`, `text/turtle`.

Responses

✓ **200** Lista di autori restituita con successo.

RESPONSE SCHEMA: `application/json` ▾

Array [

firstName string
required
The person's first name

lastName string
required
The person's last name

department string
required
The person's department

birthDate string `<date>`
The person's birthDate

deathDate string `<date>`
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

✓ 406 Formato non supportato

RESPONSE SCHEMA: application/json

error string

supported_formats Array of strings

GET /illuminated-scenes/{illuminatedSceneId}/authors

Response samples

200

406

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
  - {  
      "firstName": "string",  
      "lastName": "string",  
      "department": "string",  
      "birthDate": "2019-08-24",  
      "deathDate": "2019-08-24",  
      "id": "string",  
      + "identifiers": [ ... ],  
      + "extensions": [ ... ]  
    }  
]
```

Add a new author to an illuminated

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId string
required

REQUEST BODY SCHEMA: application/json

firstName	string
required	The person's first name
lastName	string
required	The person's last name
department	string
required	The person's department
birthDate	string <date>
	The person's birthDate
deathDate	string <date>
	The person's deathDate
id	string
	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ 201 Author created successfully

RESPONSE SCHEMA: application/json

firstName	string
required	The person's first name
lastName	string
required	The person's last name
department	string
required	The person's department
birthDate	string <date>
	The person's birthDate
deathDate	string <date>
	The person's deathDate
id	string
	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)

extensions >

Array of objects (Extension)

— 401 Unauthorized

POST /illuminated-scenes/{illuminatedSceneId}/authors

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Response samples

201

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

```
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Get an author of an illuminated scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId string
required

authorId string
required

Responses

✓ 200 Author retrieved successfully

RESPONSE SCHEMA: application/json

firstName string
required
The person's first name

lastName string
required
The person's last name

department string
required
The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

`extensions` >

Array of objects (Extension)

— **401** Unauthorized

— **404** Author not found

GET /illuminated-scenes/{illuminatedSceneId}/authors/{authorId}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Replace an author of an illuminated scene

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

`illuminatedSceneId` string
required

`authorId` string
required

firstName <small>required</small>	string The person's first name
lastName <small>required</small>	string The person's last name
department <small>required</small>	string The person's department
birthDate	string <date> The person's birthDate
deathDate	string <date> The person's deathDate
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ 200 Author replaced successfully

firstName <small>required</small>	string The person's first name
lastName <small>required</small>	string The person's last name
department <small>required</small>	string The person's department
birthDate	string <date> The person's birthDate
deathDate	string <date> The person's deathDate
id	string Unique identifier for the object. It could not be send in POST

`identifiers` > Array of objects (Identifier)

`extensions` > Array of objects (Extension)

— 401 Unauthorized

— 404 Author not found

PUT /illuminated-scenes/{illuminatedSceneId}/authors/{authorId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",
```

```
"birthDate": "2019-08-24",
"deathDate": "2019-08-24",
"id": "string",
- "identifiers": [
  + { ... }
],
- "extensions": [
  + { ... }
]
}
```

Update an author of an illuminated scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId string
required

authorId string
required

REQUEST BODY SCHEMA: application/json

firstName string
required
The person's first name

lastName string
required
The person's last name

department string
required
The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ 200 Author updated successfully

RESPONSE SCHEMA: application/json

firstName	string	
required		The person's first name
lastName	string	
required		The person's last name
department	string	
required		The person's department
birthDate	string <date>	
		The person's birthDate
deathDate	string <date>	
		The person's deathDate
id	string	
		Unique identifier for the object. It could not be send in POST
identifiers >		Array of objects (Identifier)
extensions >		Array of objects (Extension)

— 401 Unauthorized

— 404 Author not found

PATCH /illuminated-scenes/{illuminatedSceneId}/authors/{authorId}

Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

{

 "firstName": "string",
 "lastName": "string",

```
"department": "string",
"birthDate": "2019-08-24",
"deathDate": "2019-08-24",
"id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "firstName": "string",
  "lastName": "string",
  "department": "string",
  "birthDate": "2019-08-24",
  "deathDate": "2019-08-24",
  "id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Delete an author of an illuminated scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId string
required

authorId string
required

Responses

— **204** Author deleted successfully

— **401** Unauthorized

— **404** Author not found

DELETE /illuminated-scenes/{illuminatedSceneId}/authors/{authorId}

Editors of an Illuminated Scene

Editors related to Illuminated Scene

Get editors of an illuminated scene

returns a list of editors with support for pagination, sorting and multiple formats.

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

illuminatedSceneId string
required

QUERY PARAMETERS

page integer **>= 0**
Default: 0
Numero della pagina (facoltativo, predefinito: 0). L'indice parte da 0.

size integer **>= 1**
Default: 20
Numero di elementi per pagina (facoltativo, predefinito: 20). Deve essere maggiore di 0.

sort string
Example: `sort=createdDate,desc`
Campo per l'ordinamento (facoltativo). Formato: `campo,direzione` (es. `createdDate,desc`). Direzioni supportate: `asc`, `desc`.

HEADER PARAMETERS

Accept string
Enum: `"application/json"` `"text/turtle"`
Formato di risposta desiderato. Valori supportati: `application/json`, `text/turtle`.

Responses

✓ **200** Lista di editori restituita con successo.

RESPONSE SCHEMA: `application/json` ▾

Array [

<code>firstName</code> required	string The person's first name
<code>lastName</code> required	string The person's last name
<code>department</code> required	string The person's department
<code>birthDate</code>	string <date> The person's birthDate
<code>deathDate</code>	string <date> The person's deathDate
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers</code> >	Array of objects (Identifier)
<code>extensions</code> >	Array of objects (Extension)

]

✗ **406** Formato non supportato

RESPONSE SCHEMA: `application/json`

error string

`supported_formats` Array of strings

GET /illuminated-scenes/{illuminatedSceneId}/editors

Response samples

200

406

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
- {  
  "firstName": "string",  
  "lastName": "string",  
  "department": "string",  
  "birthDate": "2019-08-24",  
  "deathDate": "2019-08-24",  
  "id": "string",  
  + "identifiers": [ ... ],  
  + "extensions": [ ... ]  
}  
]
```

Add a new editor to an

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

`illuminatedSceneId` string
required

REQUEST BODY SCHEMA: application/json

`firstName` string
required
The person's first name

`lastName` string
required
The person's last name

department	string
required	The person's department
birthDate	string <date>
	The person's birthDate
deathDate	string <date>
	The person's deathDate
id	string
	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ **201** Editor created successfully

RESPONSE SCHEMA: application/json

firstName	string
required	The person's first name
lastName	string
required	The person's last name
department	string
required	The person's department
birthDate	string <date>
	The person's birthDate
deathDate	string <date>
	The person's deathDate
id	string
	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— **401** Unauthorized

POST /illuminated-scenes/{illuminatedSceneId}/editors

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Response samples

201

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

}

Get an editor of an illuminated scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId string
required

editorId string
required

Responses

✓ 200 Editor retrieved successfully

RESPONSE SCHEMA: application/json

firstName string
required
The person's first name

lastName string
required
The person's last name

department string
required
The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— 401 Unauthorized

GET /illuminated-scenes/{illuminatedSceneId}/editors/{editorId}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Replace an editor of an illuminated scene

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

illuminatedSceneId string
required

editorId string
required

REQUEST BODY SCHEMA: application/json

firstName string
required
The person's first name

lastName	string
required	The person's last name
department	string
required	The person's department
birthDate	string <date>
	The person's birthDate
deathDate	string <date>
	The person's deathDate
id	string
	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ 200 Editor replaced successfully

RESPONSE SCHEMA: application/json

firstName	string
required	The person's first name
lastName	string
required	The person's last name
department	string
required	The person's department
birthDate	string <date>
	The person's birthDate
deathDate	string <date>
	The person's deathDate
id	string
	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— 401 Unauthorized

```
PUT /illuminated-scenes/{illuminatedSceneId}/editors/{editorId}
```

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "firstName": "string",
  "lastName": "string",
  "department": "string",
  "birthDate": "2019-08-24",
  "deathDate": "2019-08-24",
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "firstName": "string",
  "lastName": "string",
  "department": "string",
  "birthDate": "2019-08-24",
  "deathDate": "2019-08-24",
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

```
- "extensions": [
  + { ... }
]
}
```

Update an editor of an illuminated scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId string
required

editorId string
required

REQUEST BODY SCHEMA: application/json

firstName string
required
The person's first name

lastName string
required
The person's last name

department string
required
The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ 200 Editor updated successfully

RESPONSE SCHEMA: application/json

<code>firstName</code>	string
required	The person's first name
<code>lastName</code>	string
required	The person's last name
<code>department</code>	string
required	The person's department
<code>birthDate</code>	string <date>
	The person's birthDate
<code>deathDate</code>	string <date>
	The person's deathDate
<code>id</code>	string
	Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

— **401** Unauthorized

— **404** Editor not found

PATCH /illuminated-scenes/{illuminatedSceneId}/editors/{editorId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "firstName": "string",
  "lastName": "string",
  "department": "string",
  "birthDate": "2019-08-24",
  "deathDate": "2019-08-24",
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
}
```

```
- "extensions": [
  + { ... }
]
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "firstName": "string",
  "lastName": "string",
  "department": "string",
  "birthDate": "2019-08-24",
  "deathDate": "2019-08-24",
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

Delete an editor of an illuminated scene

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

illuminatedSceneId <small>required</small>	string
editorId <small>required</small>	string

Responses

— 204 Editor deleted successfully

— 401 Unauthorized

— 404 Editor not found

DELETE /illuminated-scenes/{illuminatedSceneId}/editors/{editorId}

Locations

All Locations related operations

Get Locations

Get all locations. This operation retrieves all the available locations.

AUTHORIZATIONS: >

bearerAuth

QUERY PARAMETERS

id	Array of strings Filtrare le location per ID, valori multipli separati da virgola ad es. 1,2,3
name	string Filtrare le location per nome
address	string Filtrare le location per indirizzo
coordinates	Array of numbers Filtrare le location per coordinate, valori separati da virgola ad es. 1.0,2.0
identifiers	Array of strings Filtrare i manoscritti per identificatore, valori multipli separati da virgola ad es. 1,2,3
page	integer >= 0 Default: 0 Numero della pagina (facoltativo, predefinito: 0). L'indice parte da 0.
size	integer >= 1 Default: 20

Numero di elementi per pagina (facoltativo, predefinito: 20). Deve essere maggiore di 0.

sort

string

Example: `sort=name,asc`

Campo per l'ordinamento (facoltativo). Formato: `campo,direzione` (es. `name,asc`). Direzioni supportate: `asc`, `desc`.

HEADER PARAMETERS

Accept

string

Enum: `"application/json"` `"text/turtle"`

Formato di risposta desiderato. Valori supportati: `application/json`,
`text/turtle`

Responses

✓ 200 Successful operation

RESPONSE SCHEMA: `application/json`

Array [

 name string

 description string

 address string

 coordinates Array of numbers `= 2 items`

 id string

 Unique identifier for the object. It could not be send in POST

 identifiers > Array of objects (Identifier)

 extensions > Array of objects (Extension)

]

— 401 Unauthorized

GET /locations

Response samples

200

Content type

application/json

Copy Expand all Collapse all

```
[  
- {  
    "id": "1",  
    "name": "Location 1",  
    "description": "Description 1",  
    "address": "Address 1",  
    + "coordinates": [ ... ]  
},  
- {  
    "id": "2",  
    "name": "Location 2",  
    "description": "Description 2",  
    "address": "Address 2",  
    + "coordinates": [ ... ]  
}  
]
```

Create Location

Create a new location with the specified details

AUTHORIZATIONS: >

bearerAuth

REQUEST BODY SCHEMA: application/json

name	string
description	string
address	string
coordinates	Array of numbers = 2 items
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

— **201** Location created successfully

— **401** Unauthorized

POST /locations

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "name": "string",
  "description": "string",
  "address": "string",
  - "coordinates": [
    0,
    0
  ],
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

Get Location

Get a specific location by its ID

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

id string
required

Responses

✓ 200 Successful operation

RESPONSE SCHEMA: application/json

name	string
description	string
address	string
coordinates	Array of numbers = 2 items
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— 401 Unauthorized

— 404 Location not found

GET /locations/{id}

Response samples

200

Content type
application/json

Copy Expand all Collapse all

```
{  
  "name": "string",  
  "description": "string",  
  "address": "string",
```

```
- "coordinates": [
  0,
  0
],
"id": "string",
- "identifiers": [
  + { ... }
],
- "extensions": [
  + { ... }
]
}
```

Update Location

Update a location with new information

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id
required string

REQUEST BODY SCHEMA: **application/json**

name string

description string

address string

coordinates Array of numbers **= 2 items**

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** Location updated successfully

name	string
description	string
address	string
coordinates	Array of numbers = 2 items
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— 401 Unauthorized**— 404 Location not found**

PUT /locations/{id}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "name": "string",
  "description": "string",
  "address": "string",
  - "coordinates": [
    0,
    0
  ],
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
}
```

```
- "extensions": [
    + { ... }
]
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
    "name": "string",
    "description": "string",
    "address": "string",
    - "coordinates": [
        0,
        0
    ],
    "id": "string",
    - "identifiers": [
        + { ... }
    ],
    - "extensions": [
        + { ... }
    ]
}
```

Patch Location

partial update a location with new information

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id required	string
------------------------------	--------

REQUEST BODY SCHEMA: application/json

name	string
-------------	--------

description	string
--------------------	--------

address	string
coordinates	Array of numbers = 2 items
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ **200** Location patched successfully

RESPONSE SCHEMA: application/json

name	string
description	string
address	string
coordinates	Array of numbers = 2 items
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— **401** Unauthorized

— **404** Location not found

PATCH /locations/{id}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "name": "string",  
  "description": "string",  
  "address": "string",  
  - "coordinates": [  
    0,  
    0  
  ],  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "name": "string",  
  "description": "string",  
  "address": "string",  
  - "coordinates": [  
    0,  
    0  
  ],  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Delete Location

Delete a location. This operation deletes a specific location by its ID.

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

id
required string

Responses

— **204** Location deleted successfully

— **401** Unauthorized

— **404** Location not found

DELETE /locations/{id}

Manuscripts

All Manuscripts related operations

Get all manuscripts

This operation retrieves all manuscripts.

AUTHORIZATIONS: >

bearerAuth

QUERY PARAMETERS

id Array of strings

title	string Filtrati i manoscritti per titolo
description	string Filtrati i manoscritti per descrizione
status	string Enum: <code>"draft"</code> <code>"in_review"</code> <code>"revisions_required"</code> <code>"accepted"</code> <code>"in_editing"</code> <code>"published"</code> <code>"rejected"</code> <code>"withdrawn"</code> Filtrati i manoscritti in base al loro stato: <ul style="list-style-type: none"> • <code>draft</code> : Bozza • <code>in_review</code> : In revisione • <code>revisions_required</code> : Revisione necessaria • <code>accepted</code> : Accettato • <code>in_editing</code> : In fase di editing • <code>published</code> : Pubblicato • <code>rejected</code> : Rifiutato • <code>withdrawn</code> : Ritirato
locations	string Filtrati i manoscritti per nome location
locationId	Array of strings Filtrati i manoscritti per location ID, valori multipli separati da virgola ad es. 1,2,3
authors	Array of strings Filtrati i manoscritti per nome autore
authorId	Array of strings
editorId	Array of strings Filtrati i manoscritti per ID dell'editore, valori multipli separati da virgola ad es. 1,2,3
identifiers	Array of strings Filtrati i manoscritti per identificatore, valori multipli separati da virgola ad es. 1,2,3
page	integer <code>>= 0</code> Default: <code>0</code> Numero della pagina (facoltativo, predefinito: 0). L'indice parte da 0.
size	integer <code>>= 1</code> Default: <code>20</code> Numero di elementi per pagina (facoltativo, predefinito: 20). Deve essere maggiore di 0.
sort	string Example: <code>sort=name,asc</code> Campo per l'ordinamento (facoltativo). Formato: <code>campo,direzione</code> (es. <code>name,asc</code>). Direzioni supportate: <code>asc</code> , <code>desc</code> .

Accept string
required Enum: "application/json" "text/turtle"
 Formato di risposta desiderato. Valori supportati: **application/json**, **text/turtle**

Responses

✓ 200 A list of manuscripts

RESPONSE SCHEMA: application/json

Array [

id	string
title	string
description	string
previewImage	string <uri>
events >	Array of objects (Reference)
projects >	Array of objects (Reference)
locations >	Array of objects (Reference)
scenes >	Array of objects (Reference)
authors >	Array of objects (Reference)
editors >	Array of objects (Reference)
status	string Enum: "draft" "in_review" "revisions_required" "accepted" "in_editing" "published" "rejected" "withdrawn" Stato attuale del manoscritto:
	<ul style="list-style-type: none"> • draft : Bozza • in_review : In revisione • revisions_required : Revisione necessaria • accepted : Accettato • in_editing : In fase di editing • published : Pubblicato • rejected : Rifiutato • withdrawn : Ritirato

— **401** Unauthorized

— **406** Formato non supportato

GET /manuscripts

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
- {  
  "id": "string",  
  "title": "string",  
  "description": "string",  
  "previewImage": "http://example.com",  
+ "events": [ ... ],  
+ "projects": [ ... ],  
+ "locations": [ ... ],  
+ "scenes": [ ... ],  
+ "authors": [ ... ],  
+ "editors": [ ... ],  
  "status": "draft"  
}  
]
```

Create a new manuscript

This operation creates a new manuscript.

AUTHORIZATIONS: > *bearerAuth*

REQUEST BODY SCHEMA: application/json

id	string
----	--------

| title | string |

description	string
previewImage	string <uri>
events >	Array of objects (Reference)
projects >	Array of objects (Reference)
locations >	Array of objects (Reference)
scenes >	Array of objects (Reference)
authors >	Array of objects (Reference)
editors >	Array of objects (Reference)
status	string Enum: <code>"draft"</code> <code>"in_review"</code> <code>"revisions_required"</code> <code>"accepted"</code> <code>"in_editing"</code> <code>"published"</code> <code>"rejected"</code> <code>"withdrawn"</code>

Stato attuale del manoscritto:

- `draft` : Bozza
- `in_review` : In revisione
- `revisions_required` : Revisione necessaria
- `accepted` : Accettato
- `in_editing` : In fase di editing
- `published` : Pubblicato
- `rejected` : Rifiutato
- `withdrawn` : Ritirato

Responses

✓ 201 Manuscript created

RESPONSE SCHEMA: application/json

id	string
title	string
description	string
previewImage	string <uri>
events >	Array of objects (Reference)
projects >	Array of objects (Reference)

<code>locations ></code>	Array of objects (Reference)
<code>scenes ></code>	Array of objects (Reference)
<code>authors ></code>	Array of objects (Reference)
<code>editors ></code>	Array of objects (Reference)
<code>status</code>	<p>string</p> <p>Enum: <code>"draft"</code> <code>"in_review"</code> <code>"revisions_required"</code> <code>"accepted"</code> <code>"in_editing"</code> <code>"published"</code> <code>"rejected"</code> <code>"withdrawn"</code></p> <p>Stato attuale del manoscritto:</p> <ul style="list-style-type: none"> • <code>draft</code> : Bozza • <code>in_review</code> : In revisione • <code>revisions_required</code> : Revisione necessaria • <code>accepted</code> : Accettato • <code>in_editing</code> : In fase di editing • <code>published</code> : Pubblicato • <code>rejected</code> : Rifiutato • <code>withdrawn</code> : Ritirato

— 401 Unauthorized

POST /manuscripts

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "id": "string",
  "title": "string",
  "description": "string",
  "previewImage": "http://example.com",
  - "events": [
    + { ... }
  ],
  - "projects": [
    + { ... }
  ],
}
```

```
- "locations": [
    + { ... }
],
- "scenes": [
    + { ... }
],
- "authors": [
    + { ... }
],
- "editors": [
    + { ... }
],
"status": "draft"
}
```

Response samples

201

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "id": "string",  
    "title": "string",  
    "description": "string",  
    "previewImage": "http://example.com",  
    - "events": [  
        + { ... }  
    ],  
    - "projects": [  
        + { ... }  
    ],  
    - "locations": [  
        + { ... }  
    ],  
    - "scenes": [  
        + { ... }  
    ],  
    - "authors": [  
        + { ... }  
    ],  
    - "editors": [  
        + { ... }  
    ],  
    "status": "draft"  
}
```

Get a manuscript

This operation retrieves a manuscript based on its ID.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id required	string
------------------------------	--------

Responses

✓ 200 A manuscript

RESPONSE SCHEMA: application/json ✓

<code>title</code>	string
<code>description</code>	string
<code>creationDate</code>	string <date-time>
<code>publicationDate</code>	string <date-time>
<code>previewImage</code>	string <uri>
<code>events ></code>	Array of objects (Reference)
<code>projects ></code>	Array of objects (Reference)
<code>locations ></code>	Array of objects (Location)
<code>contents ></code>	Array of objects (LighContent)
<code>authors ></code>	Array of objects (Reference)
<code>editors ></code>	Array of objects (Reference)
<code>illuminated-scenes ></code>	Array of objects (LightIlluminatedScene)
<code>status</code>	<p>string</p> <p>Enum: <code>"draft"</code> <code>"in_review"</code> <code>"revisions_required"</code> <code>"accepted"</code> <code>"in_editing"</code> <code>"published"</code> <code>"rejected"</code> <code>"withdrawn"</code></p> <p>Stato attuale del manoscritto:</p> <ul style="list-style-type: none"> • <code>draft</code> : Bozza • <code>in_review</code> : In revisione • <code>revisions_required</code> : Revisione necessaria • <code>accepted</code> : Accettato • <code>in_editing</code> : In fase di editing • <code>published</code> : Pubblicato • <code>rejected</code> : Rifiutato • <code>withdrawn</code> : Ritirato
<code>id</code>	<p>string</p> <p>Unique identifier for the object. It could not be send in POST</p>
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

— **401** Unauthorized

— **404** Manuscript not found

— **406** Formato non supportato

```
GET /manuscripts/{id}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "title": "string",
  "description": "string",
  "creationDate": "2019-08-24T14:15:22Z",
  "publicationDate": "2019-08-24T14:15:22Z",
  "previewImage": "http://example.com",
  - "events": [
    + { ... }
  ],
  - "projects": [
    + { ... }
  ],
  - "locations": [
    + { ... }
  ],
  - "contents": [
    + { ... }
  ],
  - "authors": [
    + { ... }
  ],
  - "editors": [
    + { ... }
  ],
  - "illuminated-scenes": [
    + { ... }
  ],
  "status": "draft",
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

}

Update a manuscript

This operation updates a manuscript based on its ID.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id
required string

REQUEST BODY SCHEMA: application/json

id string

title string

description string

previewImage string <uri>

events > Array of objects (Reference)

projects > Array of objects (Reference)

locations > Array of objects (Reference)

scenes > Array of objects (Reference)

authors > Array of objects (Reference)

editors > Array of objects (Reference)

status string

Enum: `"draft"` `"in_review"` `"revisions_required"` `"accepted"`
`"in_editing"` `"published"` `"rejected"` `"withdrawn"`

Stato attuale del manoscritto:

- `draft` : Bozza
- `in_review` : In revisione
- `revisions_required` : Revisione necessaria
- `accepted` : Accettato
- `in_editing` : In fase di editing
- `published` : Pubblicato
- `rejected` : Rifiutato
- `withdrawn` : Ritirato

Responses

✓ 200 Manuscript updated

RESPONSE SCHEMA: application/json

id	string
title	string
description	string
previewImage	string <uri>
events >	Array of objects (Reference)
projects >	Array of objects (Reference)
locations >	Array of objects (Reference)
scenes >	Array of objects (Reference)
authors >	Array of objects (Reference)
editors >	Array of objects (Reference)
status	string Enum: <code>"draft"</code> <code>"in_review"</code> <code>"revisions_required"</code> <code>"accepted"</code> <code>"in_editing"</code> <code>"published"</code> <code>"rejected"</code> <code>"withdrawn"</code> Stato attuale del manoscritto: <ul style="list-style-type: none">• <code>draft</code> : Bozza• <code>in_review</code> : In revisione• <code>revisions_required</code> : Revisione necessaria• <code>accepted</code> : Accettato• <code>in_editing</code> : In fase di editing• <code>published</code> : Pubblicato• <code>rejected</code> : Rifiutato• <code>withdrawn</code> : Ritirato

— 401 Unauthorized

— 404 Manuscript not found

PATCH /manuscripts/{id}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "id": "string",  
    "title": "string",  
    "description": "string",  
    "previewImage": "http://example.com",  
    - "events": [  
        + { ... }  
    ],  
    - "projects": [  
        + { ... }  
    ],  
    - "locations": [  
        + { ... }  
    ],  
    - "scenes": [  
        + { ... }  
    ],  
    - "authors": [  
        + { ... }  
    ],  
    - "editors": [  
        + { ... }  
    ],  
    "status": "draft"  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "id": "string",  
    "title": "string",  
    "description": "string",  
    "previewImage": "http://example.com",  
    - "events": [  
        + { ... }  
    ],  
    - "projects": [  
        + { ... }  
    ],  
    - "locations": [  
        + { ... }  
    ],  
    - "scenes": [  
        + { ... }  
    ],  
    - "authors": [  
        + { ... }  
    ],  
    - "editors": [  
        + { ... }  
    ],  
    "status": "draft"  
}
```

```
"description": "string",
"previewImage": "http://example.com",
- "events": [
    + { ... }
],
- "projects": [
    + { ... }
],
- "locations": [
    + { ... }
],
- "scenes": [
    + { ... }
],
- "authors": [
    + { ... }
],
- "editors": [
    + { ... }
],
"status": "draft"
}
```

Delete a manuscript

This operation deletes a manuscript based on its ID.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id required	string
------------------------------	--------

Responses

— **204** Manuscript deleted

— **401** Unauthorized

— **404** Manuscript not found

DELETE /manuscripts/{id}

Replace a manuscript

This operation replaces a manuscript based on its ID.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id
required string

REQUEST BODY SCHEMA: application/json

id string

title string

description string

previewImage string <uri>

events > Array of objects (Reference)

projects > Array of objects (Reference)

locations > Array of objects (Reference)

scenes > Array of objects (Reference)

authors > Array of objects (Reference)

editors > Array of objects (Reference)

status string

Enum: `"draft"` `"in_review"` `"revisions_required"` `"accepted"`
`"in_editing"` `"published"` `"rejected"` `"withdrawn"`

Stato attuale del manoscritto:

- `draft` : Bozza
- `in_review` : In revisione
- `revisions_required` : Revisione necessaria
- `accepted` : Accettato
- `in_editing` : In fase di editing
- `published` : Pubblicato

- `rejected` : Rifiutato
 - `withdrawn` : Ritirato
-

Responses

✓ 200 Manuscript replaced

RESPONSE SCHEMA: application/json

<code>id</code>	string
<code>title</code>	string
<code>description</code>	string
<code>previewImage</code>	string <uri>
<code>events ></code>	Array of objects (Reference)
<code>projects ></code>	Array of objects (Reference)
<code>locations ></code>	Array of objects (Reference)
<code>scenes ></code>	Array of objects (Reference)
<code>authors ></code>	Array of objects (Reference)
<code>editors ></code>	Array of objects (Reference)
<code>status</code>	string Enum: <code>"draft"</code> <code>"in_review"</code> <code>"revisions_required"</code> <code>"accepted"</code> <code>"in_editing"</code> <code>"published"</code> <code>"rejected"</code> <code>"withdrawn"</code> Stato attuale del manoscritto: <ul style="list-style-type: none">• <code>draft</code> : Bozza• <code>in_review</code> : In revisione• <code>revisions_required</code> : Revisione necessaria• <code>accepted</code> : Accettato• <code>in_editing</code> : In fase di editing• <code>published</code> : Pubblicato• <code>rejected</code> : Rifiutato• <code>withdrawn</code> : Ritirato

— 401 Unauthorized

— 404 Manuscript not found

PUT /manuscripts/{id}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "id": "string",  
    "title": "string",  
    "description": "string",  
    "previewImage": "http://example.com",  
    - "events": [  
        + { ... }  
    ],  
    - "projects": [  
        + { ... }  
    ],  
    - "locations": [  
        + { ... }  
    ],  
    - "scenes": [  
        + { ... }  
    ],  
    - "authors": [  
        + { ... }  
    ],  
    - "editors": [  
        + { ... }  
    ],  
    "status": "draft"  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "id": "string",  
    "title": "string",  
    "description": "string",  
    "previewImage": "http://example.com",  
    - "events": [  
        + { ... }  
    ],  
    - "projects": [  
        + { ... }  
    ],  
    - "locations": [  
        + { ... }  
    ],  
    - "scenes": [  
        + { ... }  
    ],  
    - "authors": [  
        + { ... }  
    ],  
    - "editors": [  
        + { ... }  
    ],  
    "status": "published"  
}
```

```
"description": "string",
"previewImage": "http://example.com",
- "events": [
    + { ... }
],
- "projects": [
    + { ... }
],
- "locations": [
    + { ... }
],
- "scenes": [
    + { ... }
],
- "authors": [
    + { ... }
],
- "editors": [
    + { ... }
],
"status": "draft"
}
```

Contents of a Manuscript

All Manuscript Contents related operations

Ottiene tutti i contenuti di un manoscritto

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id required	integer ID del manoscritto
------------------------------	-------------------------------

HEADER PARAMETERS

Accept

string

Enum: "application/json" "text/turtle"

Formato di risposta desiderato. Valori supportati: [application/json](#),

[text/turtle](#)

Responses

✓ 200 Elenco dei contenuti del manoscritto

RESPONSE SCHEMA: [application/json](#)

Array [

id	integer
manuscript_id	integer
intro	string
category	string Enum: "Storia e contesto" "Struttura e scrittura" "Materiali e tecniche esecutive" "Iconografia e iconologia" "Censure e modifiche" "Conservazione e restauro" "Bibliografia"
content	string
last_updated	string <date-time>

]

— 401 Unauthorized

— 404 Manoscritto non trovato

— 406 Formato non supportato

GET /manuscripts/{id}/contents

Response samples

200

Content type

application/json

```
[
  - {
    "id": 1,
    "manuscript_id": 1,
    "category": "Storia e contesto",
    "content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",
    "last_updated": "2021-09-01T12:00:00Z"
  }
]
```

Aggiunge un nuovo contenuto a un manoscritto

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id	integer
required	ID del manoscritto

REQUEST BODY SCHEMA: application/json

intro	string
category	string
required	Enum: "Storia e contesto" "Struttura e scrittura" "Materiali e tecniche esecutive" "Iconografia e iconologia" "Censure e modifiche" "Conservazione e restauro" "Bibliografia"
content	string

Responses

✓ 201 Contenuto creato con successo

RESPONSE SCHEMA: application/json

id	integer
manuscript_id	integer
intro	string

category	string Enum: "Storia e contesto" "Struttura e scrittura" "Materiali e tecniche esecutive" "Iconografia e iconologia" "Censure e modifiche" "Conservazione e restauro" "Bibliografia"
content	string
last_updated	string <date-time>

— 401 Unauthorized

POST /manuscripts/{id}/contents

Request samples

Payload

Content type

application/json

Copy

```
{  
  "category": "Storia e contesto",  
  "content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit."  
}
```

Response samples

201

Content type

application/json

Copy

```
{  
  "id": 1,  
  "manuscript_id": 1,  
  "category": "Storia e contesto",  
  "content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",  
  "last_updated": "2021-09-01T12:00:00Z"  
}
```

Aggiorna un contenuto specifico di un manoscritto

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

manuscriptId integer
required
ID del manoscritto

contentId integer
required
ID del contenuto

REQUEST BODY SCHEMA: application/json

intro string

category string
required
Enum: "Storia e contesto" | "Struttura e scrittura" | "Materiali e tecniche esecutive" | "Iconografia e iconologia" | "Censure e modifiche" | "Conservazione e restauro" | "Bibliografia"

content string
required

Responses

✓ 200 Contenuto aggiornato con successo

RESPONSE SCHEMA: application/json

id integer

manuscript_id integer

intro string

category string
Enum: "Storia e contesto" | "Struttura e scrittura" | "Materiali e tecniche esecutive" | "Iconografia e iconologia" | "Censure e modifiche" | "Conservazione e restauro" | "Bibliografia"

content string

last_updated string <date-time>

— 401 Unauthorized

PUT /manuscripts/{manuscriptId}/contents/{contentId}

Request samples

Payload

Content type

application/json

Copy

```
{  
  "category": "Storia e contesto",  
  "content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit."  
}
```

Response samples

200

Content type

application/json

Copy

```
{  
  "id": 1,  
  "manuscript_id": 1,  
  "category": "Storia e contesto",  
  "content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",  
  "last_updated": "2021-09-01T12:00:00Z"  
}
```

Elimina un contenuto di un manoscritto

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

manuscriptId
required

integer
ID del manoscritto

contentId	integer
required	ID del contenuto

Responses

— **204** Contenuto eliminato con successo

— **401** Unauthorized

— **404** Contenuto non trovato

DELETE /manuscripts/{manuscriptId}/contents/{contentId}

Locations of a Manuscript

Locations related to Manuscript

Get locations of a manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId	string
required	

Responses

✓ **200** Successful operation

RESPONSE SCHEMA: *application/json*

Array [

<code>name</code>	string
<code>description</code>	string
<code>address</code>	string
<code>coordinates</code>	Array of numbers <small>= 2 items</small>
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

]

— **401 Unauthorized**

— **404 Manuscript not found**

GET /manuscripts/{manuscriptId}/locations

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
- {  
  "name": "string",  
  "description": "string",  
  "address": "string",  
  + "coordinates": [ ... ],  
  "id": "string",  
  + "identifiers": [ ... ],  
  + "extensions": [ ... ]  
}  
]
```

Add location to manuscript

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

manuscriptId required string

REQUEST BODY SCHEMA: application/json

name string

description string

address string

coordinates Array of numbers = 2 items

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

— 201 Location added successfully

— 401

POST /manuscripts/{manuscriptId}/locations

Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

```
{  
    "name": "string",  
    "description": "string",  
    "address": "string",  
    - "coordinates": [  
        0,  
        0  
    ],  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Get location of a manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId string
required

locationId string
required

Responses

✓ **200** Successful operation

RESPONSE SCHEMA: **application/json**

name string

description string

address string

coordinates Array of numbers = 2 items

id string

Unique identifier for the object. It could not be send in POST

`identifiers` > Array of objects (Identifier)

`extensions` > Array of objects (Extension)

— **401** Unauthorized

— **404** Location not found

GET /manuscripts/{manuscriptId}/locations/{locationId}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "name": "string",  
  "description": "string",  
  "address": "string",  
  - "coordinates": [  
    0,  
    0  
  ],  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Update location of manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId
required

string

locationId
required

string

REQUEST BODY SCHEMA: application/json

name string

description string

address string

coordinates Array of numbers **= 2 items**

id string

Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** Location updated successfully

RESPONSE SCHEMA: application/json

name string

description string

address string

coordinates Array of numbers **= 2 items**

id string

Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— **401** Unauthorized

— 404 Location not found

PUT /manuscripts/{manuscriptId}/locations/{locationId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "name": "string",  
    "description": "string",  
    "address": "string",  
    - "coordinates": [  
        0,  
        0  
    ],  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "name": "string",  
    "description": "string",  
    "address": "string",  
    - "coordinates": [  
        0,  
        0  
    ],
```

```
        "id": "string",
      - "identifiers": [
          + { ... }
        ],
      - "extensions": [
          + { ... }
        ]
    }
```

Patch location of manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId required string
locationId required string

REQUEST BODY SCHEMA: application/json

name string
description string
address string
coordinates Array of numbers = 2 items
id string
Unique identifier for the object. It could not be send in POST
identifiers > Array of objects (Identifier)
extensions > Array of objects (Extension)

Responses

✓ 200 Location patched successfully

RESPONSE SCHEMA: application/json

name string

description	string
address	string
coordinates	Array of numbers = 2 items
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— **401** Unauthorized

— **404** Location not found

PATCH /manuscripts/{manuscriptId}/locations/{locationId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "name": "string",  
  "description": "string",  
  "address": "string",  
  - "coordinates": [  
    0,  
    0  
  ],  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "name": "string",  
    "description": "string",  
    "address": "string",  
    - "coordinates": [  
        0,  
        0  
    ],  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Delete location of manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId
required string

locationId
required string

Responses

— **204** Location deleted successfully

— **401** Unauthorized

— **404** Location not found

```
DELETE /manuscripts/{manuscriptId}/locations/{locationId}
```

Authors of a Manuscript

Authors related to Manuscript

Get authors of a manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId string
required

Responses

✓ **200** Successful operation

RESPONSE SCHEMA: application/json

Array [

firstName string
required
The person's first name

lastName string
required
The person's last name

department string
required
The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string

Unique identifier for the object. It could not be send in POST

identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)
]	

— **401** Unauthorized

— **404** Manuscript not found

GET /manuscripts/{manuscriptId}/authors

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
- {  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    + "identifiers": [ ... ],  
    + "extensions": [ ... ]  
}  
]
```

Get author of a manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId <small>required</small>	string
---	--------

authorId	string
required	

Responses

✓ 200 Successful operation

RESPONSE SCHEMA: application/json

firstName string
required
The person's first name

lastName string
required
The person's last name

department string
required
The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— 401 Unauthorized

— 404 Author not found

GET /manuscripts/{manuscriptId}/authors/{authorId}

Response samples

200

Content type
application/json

```
{
  "firstName": "string",
  "lastName": "string",
  "department": "string",
  "birthDate": "2019-08-24",
  "deathDate": "2019-08-24",
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

Update author of manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId string
required

authorId string
required

REQUEST BODY SCHEMA: application/json

firstName string
required
The person's first name

lastName string
required
The person's last name

department string
required
The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

`identifiers >` Array of objects (Identifier)

`extensions >` Array of objects (Extension)

Responses

✓ 200 Author updated successfully

RESPONSE SCHEMA: `application/json`

`firstName` string
required

The person's first name

`lastName` string
required

The person's last name

`department` string
required

The person's department

`birthDate` string `<date>`
The person's birthDate

`deathDate` string `<date>`
The person's deathDate

`id` string
Unique identifier for the object. It could not be send in POST

`identifiers >` Array of objects (Identifier)

`extensions >` Array of objects (Extension)

— 401 Unauthorized

— 404 Author not found

PUT /manuscripts/{manuscriptId}/authors/{authorId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Patch author of manuscript

AUTHORIZATIONS: >

bearerAuth

manuscriptId
required

string

authorId
required

string

REQUEST BODY SCHEMA: application/json

firstName
required

string

The person's first name

lastName
required

string

The person's last name

department
required

string

The person's department

birthDate

string <date>

The person's birthDate

deathDate

string <date>

The person's deathDate

id

string

Unique identifier for the object. It could not be send in POST

identifiers >

Array of objects (Identifier)

extensions >

Array of objects (Extension)

Responses

✓ **200** Author patched successfully

RESPONSE SCHEMA: application/json

firstName
required

string

The person's first name

lastName
required

string

The person's last name

department
required

string

The person's department

birthDate

string <date>

The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— 401 Unauthorized

— 404 Author not found

PATCH /manuscripts/{manuscriptId}/authors/{authorId}

Request samples

Payload

Content type
application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Response samples

200

Content type
application/json

```
{
  "firstName": "string",
  "lastName": "string",
  "department": "string",
  "birthDate": "2019-08-24",
  "deathDate": "2019-08-24",
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

Remove author from manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId required string

authorId required string

Responses

— **204** Author removed successfully

— **401** Unauthorized

— **404** Author not found

DELETE /manuscripts/{manuscriptId}/authors/{authorId}

Editors of a Manuscript

Editors related to Manuscript

Get editors of a manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId string
required

Responses

✓ 200 Successful operation

RESPONSE SCHEMA: application/json

Array [

firstName string
required
The person's first name

lastName string
required
The person's last name

department string
required
The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

]

— **401** Unauthorized

— **404** Manuscript not found

GET /manuscripts/{manuscriptId}/editors

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
  - {  
      "firstName": "string",  
      "lastName": "string",  
      "department": "string",  
      "birthDate": "2019-08-24",  
      "deathDate": "2019-08-24",  
      "id": "string",  
      + "identifiers": [ ... ],  
      + "extensions": [ ... ]  
    }  
]
```

Add editor to manuscript

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

manuscriptId string
required

REQUEST BODY SCHEMA: application/json

firstName string
required The person's first name

<code>lastName</code>	string
required	The person's last name
<code>department</code>	string
required	The person's department
<code>birthDate</code>	string <date>
	The person's birthDate
<code>deathDate</code>	string <date>
	The person's deathDate
<code>id</code>	string
	Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

Responses

— **201** Editor added successfully

— **401** Unauthorized

POST /manuscripts/{manuscriptId}/editors

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "firstName": "string",
  "lastName": "string",
  "department": "string",
  "birthDate": "2019-08-24",
  "deathDate": "2019-08-24",
  "id": "string",
```

```
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Get editor of a manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId
required string

editorId
required string

Responses

✓ **200** Successful operation

RESPONSE SCHEMA: application/json

firstName string
required The person's first name

lastName string
required The person's last name

department string
required The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

`extensions` >

Array of objects (Extension)

— **401** Unauthorized

— **404** Editor not found

GET /manuscripts/{manuscriptId}/editors/{editorId}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Update editor of manuscript

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

`manuscriptId`
required

string

`editorId`
required

string

firstName <small>required</small>	string The person's first name
lastName <small>required</small>	string The person's last name
department <small>required</small>	string The person's department
birthDate	string <date> The person's birthDate
deathDate	string <date> The person's deathDate
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ 200 Editor updated successfully

firstName <small>required</small>	string The person's first name
lastName <small>required</small>	string The person's last name
department <small>required</small>	string The person's department
birthDate	string <date> The person's birthDate
deathDate	string <date> The person's deathDate
id	string Unique identifier for the object. It could not be send in POST

`identifiers` > Array of objects (Identifier)

`extensions` > Array of objects (Extension)

— 401 Unauthorized

— 404 Editor not found

PUT /manuscripts/{manuscriptId}/editors/{editorId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",
```

```
"birthDate": "2019-08-24",
"deathDate": "2019-08-24",
"id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Patch editor of manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId
required string

editorId
required string

REQUEST BODY SCHEMA: application/json

firstName
required string
The person's first name

lastName
required string
The person's last name

department
required string
The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ 200 Editor patched successfully

RESPONSE SCHEMA: application/json

firstName <small>required</small>	string	The person's first name
lastName <small>required</small>	string	The person's last name
department <small>required</small>	string	The person's department
birthDate	string <date>	The person's birthDate
deathDate	string <date>	The person's deathDate
id	string	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)	
extensions >	Array of objects (Extension)	

— 401 Unauthorized

— 404 Editor not found

PATCH /manuscripts/{manuscriptId}/editors/{editorId}

Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

{

 "firstName": "string",
 "lastName": "string",

```
"department": "string",
"birthDate": "2019-08-24",
"deathDate": "2019-08-24",
"id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "firstName": "string",
  "lastName": "string",
  "department": "string",
  "birthDate": "2019-08-24",
  "deathDate": "2019-08-24",
  "id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Remove editor from manuscript

[AUTHORIZATIONS](#) >

bearerAuth

PATH PARAMETERS

manuscriptId
required string

editorId
required string

Responses

— **204** Editor removed successfully

— **401** Unauthorized

— **404** Editor not found

DELETE /manuscripts/{manuscriptId}/editors/{editorId}

Illuminated Scenes of a Manuscript

Illuminated Scenes related to Manuscript

Get illuminated scenes of a manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId string
required

Responses

✓ **200** Successful operation

RESPONSE SCHEMA: application/json

Array [

title	string
Title of the scene	

description	string Description of the scene
panorama	string <uri> Panoramic view of the scene
previewImage	string <uri> Preview image of the scene
authors >	Array of objects (Reference) List of authors of the scene
editors >	Array of objects (Reference) List of editors
visibility	string Enum: <code>"public"</code> <code>"private"</code> <code>"restricted"</code> Visibility status of the scene
project >	object (Reference) Una reference da una risorsa a un'altra
manuscript >	object (Reference) Una reference da una risorsa a un'altra
poses >	Array of objects (Pose) List of poses
annotations3D >	Array of objects (3DAnnotation) List of 3D annotations
]	

— **401** Unauthorized

— **404** Manuscript not found

GET /manuscripts/{manuscriptId}/illuminated-scenes

Response samples

200

Content type
application/json

Copy Expand all Collapse all

```
[  
  - {  
    "title": "string",  
    "description": "string",  
    "panorama": "http://example.com",  
    "previewImage": "http://example.com",  
    + "authors": [ ... ],  
    + "editors": [ ... ],  
    "visibility": "public",  
    + "project": { ... },  
    + "manuscript": { ... },  
    + "poses": [ ... ],  
    + "annotations3D": [ ... ]  
  }  
]
```

Add illuminated scene to manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId string
required

REQUEST BODY SCHEMA: `application/json`

title string
Title of the scene

description string
Description of the scene

panorama string <uri>
Panoramic view of the scene

previewImage string <uri>
Preview image of the scene

authors > Array of objects (Reference)
List of authors of the scene

editors > Array of objects (Reference)
List of editors

<code>visibility</code>	string Enum: "public" "private" "restricted" Visibility status of the scene
<code>project ></code>	object (Reference) Una reference da una risorsa a un'altra
<code>manuscript ></code>	object (Reference) Una reference da una risorsa a un'altra
<code>poses ></code>	Array of objects (Pose) List of poses
<code>annotations3D ></code>	Array of objects (3DAnnotation) List of 3D annotations

Responses

— **201** Illuminated scene added successfully

— **401** Unauthorized

POST /manuscripts/{manuscriptId}/illuminated-scenes

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "title": "string",
  "description": "string",
  "panorama": "http://example.com",
  "previewImage": "http://example.com",
  - "authors": [
    + { ... }
  ],
  - "editors": [
    + { ... }
  ],
  "visibility": "public",
```

```
- "project": {
    "reference": "Risorsa/1234",
    "type": "Risorsa",
    + "identifier": { ... },
    "display": "descrizione della risorsa"
},
- "manuscript": {
    "reference": "Risorsa/1234",
    "type": "Risorsa",
    + "identifier": { ... },
    "display": "descrizione della risorsa"
},
- "poses": [
    + { ... }
],
- "annotations3D": [
    + { ... }
]
}
```

Get illuminated scene of a manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId required	string
illuminatedSceneId required	string

Responses

✓ **200** Successful operation

RESPONSE SCHEMA: application/json

title	string Title of the scene
description	string Description of the scene

panorama	string <uri> Panoramic view of the scene
previewImage	string <uri> Preview image of the scene
authors >	Array of objects (Reference) List of authors of the scene
editors >	Array of objects (Reference) List of editors
visibility	string Enum: <code>"public"</code> <code>"private"</code> <code>"restricted"</code> Visibility status of the scene
project >	object (Reference) Una reference da una risorsa a un'altra
manuscript >	object (Reference) Una reference da una risorsa a un'altra
poses >	Array of objects (Reference) List of poses
annotations3D >	Array of objects (Reference) List of 3D annotations

— **401** Unauthorized

— **404** Illuminated scene not found

GET /manuscripts/{manuscriptId}/illuminated-scenes/{illuminatedSceneId}

Response samples

200

Content type
application/json

Copy Expand all Collapse all

```
{  
  "title": "string",  
  "description": "string",  
  "panorama": "http://example.com",  
  "previewImage": "http://example.com",
```

```
- "authors": [
    + { ... }
],
- "editors": [
    + { ... }
],
"visibility": "public",
- "project": {
    "reference": "Risorsa/1234",
    "type": "Risorsa",
    + "identifier": { ... },
    "display": "descrizione della risorsa"
},
- "manuscript": {
    "reference": "Risorsa/1234",
    "type": "Risorsa",
    + "identifier": { ... },
    "display": "descrizione della risorsa"
},
- "poses": [
    + { ... }
],
- "annotations3D": [
    + { ... }
]
}
```

Update illuminated scene of manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId string
required

illuminatedSceneId string
required

REQUEST BODY SCHEMA: **application/json**

title string
Title of the scene

description string
Description of the scene

panorama	string <uri> Panoramic view of the scene
previewImage	string <uri> Preview image of the scene
visibility	string Enum: <code>"public"</code> <code>"private"</code> <code>"restricted"</code> Visibility status of the scene
project >	object (Reference) Una reference da una risorsa a un'altra
manuscript >	object (Reference) Una reference da una risorsa a un'altra
poses >	Array of objects (Reference) List of poses
annotations3D >	Array of objects (Reference) List of 3D annotations

Responses

✓ 200 Illuminated scene updated successfully

RESPONSE SCHEMA: `application/json`

title	string Title of the scene
description	string Description of the scene
panorama	string <uri> Panoramic view of the scene
previewImage	string <uri> Preview image of the scene
authors >	Array of objects (Reference) List of authors of the scene
editors >	Array of objects (Reference) List of editors

<code>visibility</code>	string Enum: "public" "private" "restricted" Visibility status of the scene
<code>project ></code>	object (Reference) Una reference da una risorsa a un'altra
<code>manuscript ></code>	object (Reference) Una reference da una risorsa a un'altra
<code>poses ></code>	Array of objects (Pose) List of poses
<code>annotations3D ></code>	Array of objects (3DAnnotation) List of 3D annotations

— **401** Unauthorized

— **404** Illuminated scene not found

PUT /manuscripts/{manuscriptId}/illuminated-scenes/{illuminatedSceneId}

Request samples

Payload

Content type
application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "title": "string",
  "description": "string",
  "panorama": "http://example.com",
  "previewImage": "http://example.com",
  "visibility": "public",
  - "project": {
      "reference": "Risorsa/1234",
      "type": "Risorsa",
      + "identifier": { ... },
      "display": "descrizione della risorsa"
    },
}
```

```
- "manuscript": {
    "reference": "Risorsa/1234",
    "type": "Risorsa",
    + "identifier": { ... },
    "display": "descrizione della risorsa"
},
- "poses": [
    + { ... }
],
- "annotations3D": [
    + { ... }
]
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
    "title": "string",
    "description": "string",
    "panorama": "http://example.com",
    "previewImage": "http://example.com",
- "authors": [
    + { ... }
],
- "editors": [
    + { ... }
],
    "visibility": "public",
- "project": {
        "reference": "Risorsa/1234",
        "type": "Risorsa",
        + "identifier": { ... },
        "display": "descrizione della risorsa"
},
- "manuscript": {
        "reference": "Risorsa/1234",
        "type": "Risorsa",
        + "identifier": { ... },
        "display": "descrizione della risorsa"
},
- "poses": [
    + { ... }
],
```

```
- "annotations3D": [
  + { ... }
]
}
```

Patch illuminated scene of manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId string
required

illuminatedSceneId string
required

REQUEST BODY SCHEMA: application/json

title	string Title of the scene
description	string Description of the scene
panorama	string <uri> Panoramic view of the scene
previewImage	string <uri> Preview image of the scene
visibility	string Enum: "public" "private" "restricted" Visibility status of the scene
project >	object (Reference) Una reference da una risorsa a un'altra
manuscript >	object (Reference) Una reference da una risorsa a un'altra
poses >	Array of objects (Reference) List of poses
annotations3D >	Array of objects (Reference) List of 3D annotations

Responses

✓ 200 Illuminated scene patched successfully

RESPONSE SCHEMA: application/json

title	string	
		Title of the scene
description	string	
		Description of the scene
panorama	string <uri>	
		Panoramic view of the scene
previewImage	string <uri>	
		Preview image of the scene
authors >	Array of objects (Reference)	
		List of authors of the scene
editors >	Array of objects (Reference)	
		List of editors
visibility	string	
	Enum:	<input type="button" value="public"/> <input type="button" value="private"/> <input type="button" value="restricted"/>
		Visibility status of the scene
project >	object (Reference)	
		Una reference da una risorsa a un'altra
manuscript >	object (Reference)	
		Una reference da una risorsa a un'altra
poses >	Array of objects (Pose)	
		List of poses
annotations3D >	Array of objects (3DAnnotation)	
		List of 3D annotations

— 401 Unauthorized

— 404 Illuminated scene not found

PATCH /manuscripts/{manuscriptId}/illuminated-scenes/{illuminatedSceneId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "title": "string",  
    "description": "string",  
    "panorama": "http://example.com",  
    "previewImage": "http://example.com",  
    "visibility": "public",  
    - "project": {  
        "reference": "Risorsa/1234",  
        "type": "Risorsa",  
        + "identifier": { ... },  
        "display": "descrizione della risorsa"  
    },  
    - "manuscript": {  
        "reference": "Risorsa/1234",  
        "type": "Risorsa",  
        + "identifier": { ... },  
        "display": "descrizione della risorsa"  
    },  
    - "poses": [  
        + { ... }  
    ],  
    - "annotations3D": [  
        + { ... }  
    ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "title": "string",  
    "description": "string",  
    "panorama": "http://example.com",  
    "previewImage": "http://example.com",
```

```
- "authors": [
  + { ... }
],
- "editors": [
  + { ... }
],
"visibility": "public",
- "project": {
  "reference": "Risorsa/1234",
  "type": "Risorsa",
  + "identifier": { ... },
  "display": "descrizione della risorsa"
},
- "manuscript": {
  "reference": "Risorsa/1234",
  "type": "Risorsa",
  + "identifier": { ... },
  "display": "descrizione della risorsa"
},
- "poses": [
  + { ... }
],
- "annotations3D": [
  + { ... }
]
}
```

Delete illuminated scene of manuscript

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

manuscriptId string
required

illuminatedSceneId string
required

Responses

— **204** Illuminated scene deleted successfully

— **401** Unauthorized

DELETE /manuscripts/{manuscriptId}/illuminated-scenes/{illuminatedSceneId}

Persons

All Persons related operations

Create Person

Create a new person with the given information

AUTHORIZATIONS: > *bearerAuth*

REQUEST BODY SCHEMA: application/json

firstName required	string The person's first name
lastName required	string The person's last name
department required	string The person's department
birthDate	string <date> The person's birthDate
deathDate	string <date> The person's deathDate
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ 201 Person created

RESPONSE SCHEMA: application/json

firstName <small>required</small>	string	The person's first name
lastName <small>required</small>	string	The person's last name
department <small>required</small>	string	The person's department
birthDate	string <date>	The person's birthDate
deathDate	string <date>	The person's deathDate
id	string	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)	
extensions >	Array of objects (Extension)	

— 401 Unauthorized

POST /persons

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

{

```
"firstName": "string",
"lastName": "string",
"department": "string",
"birthDate": "2019-08-24",
```

```
"deathDate": "2019-08-24",
"id": "string",
- "identifiers": [
  + { ... }
],
- "extensions": [
  + { ... }
]
}
```

Response samples

201

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "firstName": "string",
  "lastName": "string",
  "department": "string",
  "birthDate": "2019-08-24",
  "deathDate": "2019-08-24",
  "id": "string",
- "identifiers": [
  + { ... }
],
- "extensions": [
  + { ... }
]
}
```

Get persons

Get all persons

AUTHORIZATIONS: > [bearerAuth](#)

Responses

✓ 200 A list of persons

Array [

firstName string
required The person's first name

lastName string
required The person's last name

department string
required The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

]

— 401 Unauthorized

GET /persons

Response samples

200

Content type
application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
  - {  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    + "identifiers": [ ... ],  
    + "extensions": [ ... ]  
  }  
]
```

Get a specific person

Get a person . This operation retrieves a specific person by their ID.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id
required string

Responses

✓ 200 A person

RESPONSE SCHEMA: application/json

firstName string
required
The person's first name

lastName string
required
The person's last name

department string
required
The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

<code>id</code>	string
	Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

— **401 Unauthorized**

— **404 Person not found**

GET /persons/{id}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Replace a person

Replace a person with new information

AUTHORIZATIONS: >

bearerAuth

id string
required

REQUEST BODY SCHEMA: application/json

firstName string
required
The person's first name

lastName string
required
The person's last name

department string
required
The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ 200 Person replaced

RESPONSE SCHEMA: application/json

firstName string
required
The person's first name

lastName string
required
The person's last name

department string
required
The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

<code>id</code>	string
	Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)

<code>extensions ></code>	Array of objects (Extension)
------------------------------	------------------------------

— 401 Unauthorized

— 404 Person not found

PUT /persons/{id}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Patch a person

Update a person . This operation updates a person's information based on their ID.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id string
required

REQUEST BODY SCHEMA: *application/json*

firstName string
required The person's first name

lastName string
required The person's last name

department string
required The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions >

Array of objects (Extension)

Responses

✓ 200 Person updated

RESPONSE SCHEMA: application/json

firstName string
required
The person's first name

lastName string
required
The person's last name

department string
required
The person's department

birthDate string <date>
The person's birthDate

deathDate string <date>
The person's deathDate

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— 401 Unauthorized

— 404 Person not found

PATCH /persons/{id}

Request samples

Payload

Content type

application/json

```
{
  "firstName": "string",
  "lastName": "string",
  "department": "string",
  "birthDate": "2019-08-24",
  "deathDate": "2019-08-24",
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

Response samples

200

Content type

application/json

```
{
  "firstName": "string",
  "lastName": "string",
  "department": "string",
  "birthDate": "2019-08-24",
  "deathDate": "2019-08-24",
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

Delete a person

Delete a person . This operation deletes a person based on their ID.

AUTHORIZATIONS: >

bearerAuth

id string
required

Responses

— **204** Person deleted

— **401** Unauthorized

— **404** Person not found

DELETE /persons/{id}

Poses

All Pose related operations

Retrieve all poses

AUTHORIZATIONS: > *bearerAuth*

QUERY PARAMETERS

page	integer <input type="text" value=">= 0"/>	
	Default:	<input type="text" value="0"/>
	Numero della pagina (facoltativo, predefinito: 0). L'indice parte da 0.	
size	integer <input type="text" value=">= 1"/>	
	Default:	<input type="text" value="20"/>
	Numero di elementi per pagina (facoltativo, predefinito: 20). Deve essere maggiore di 0.	
sort	string	
	Example: <input type="text" value="sort=name,asc"/>	

Campo per l'ordinamento (facoltativo). Formato: `campo,direzione` (es. `name,asc`). Direzioni supportate: `asc`, `desc`.

HEADER PARAMETERS

Accept	string Enum: <code>"application/json"</code> <code>"text/turtle"</code> Formato di risposta desiderato. Valori supportati: <code>application/json</code> , <code>text/turtle</code>
--------	--

Responses

✓ 200 List of all available poses

RESPONSE SCHEMA: `application/json` ▼

Array [

name	string Name of the pose
models >	Array of objects (Reference) List of 3D models associated with the pose
contents >	Array of objects (Reference)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

]

— 401 Unauthorized

GET /poses

Response samples

200

Content type

`application/json`

```
[  
  - {  
      "name": "string",  
      + "models": [ ... ],  
      + "contents": [ ... ],  
      "id": "string",  
      + "identifiers": [ ... ],  
      + "extensions": [ ... ]  
    }  
]
```

Create a new pose

AUTHORIZATIONS: > *bearerAuth*

REQUEST BODY SCHEMA: application/json

<code>name</code>	string Name of the pose
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers</code> >	Array of objects (Identifier)
<code>extensions</code> >	Array of objects (Extension)

Responses

— **201** Pose successfully created

— **401** Unauthorized

POST /poses

Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

```
{  
    "name": "string",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Get a pose

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

id
required string

Responses

✓ **200** Pose retrieved successfully

RESPONSE SCHEMA: application/json

name	string Name of the pose
3d-models >	Array of objects (3DModel) List of 3D models associated with the pose
contents >	Array of objects (LighContent)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)

`extensions` >

Array of objects (Extension)

— **401** Unauthorized

— **404** Pose not found

GET /poses/{id}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "name": "string",  
    - "3d-models": [  
        + { ... }  
    ],  
    - "contents": [  
        + { ... }  
    ],  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Replace a pose

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

id
required

string

<code>name</code>	string Name of the pose
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

Responses

✓ **200** Pose replaced successfully

<code>name</code>	string Name of the pose
<code>3d-models ></code>	Array of objects (3DModel) List of 3D models associated with the pose
<code>contents ></code>	Array of objects (LighContent)
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

— **401** Unauthorized

— **404** Pose not found

PUT /poses/{id}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "name": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "name": "string",  
  - "3d-models": [  
    + { ... }  
  ],  
  - "contents": [  
    + { ... }  
  ],  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Update a pose

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

id string
required

REQUEST BODY SCHEMA: application/json

name string
Name of the pose

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** Pose updated successfully

RESPONSE SCHEMA: application/json

name string
Name of the pose

3d-models > Array of objects (3DModel)
List of 3D models associated with the pose

contents > Array of objects (LighContent)

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— **401** Unauthorized

— **404** Pose not found

PATCH /poses/{id}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "name": "string",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "name": "string",  
  - "3d-models": [  
    + { ... }  
  ],  
  - "contents": [  
    + { ... }  
  ],  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Delete a pose

AUTHORIZATIONS: >

bearerAuth

id string
required

Responses

— **204** Pose deleted successfully

— **401** Unauthorized

— **404** Pose not found

DELETE /poses/{id}

tagPoses/{id}/3d-models

Get 3D models of a pose

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

poseId string
required

Responses

✓ **200** Successful operation

RESPONSE SCHEMA: application/json

Array [

<code>name</code>	string Name of the 3D model
<code>textures ></code>	Array of objects (Texture) List of textures associated with the 3D model
<code>annotations ></code>	Array of objects (3DAnnotation) List of 3D annotations linked to the model
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

]

— **401** Unauthorized

— **404** Pose not found

GET /poses/{poseId}/3d-models

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
- {  
    "name": "string",  
    + "textures": [ ... ],  
    + "annotations": [ ... ],  
    "id": "string",  
    + "identifiers": [ ... ],  
    + "extensions": [ ... ]  
}  
]
```

Add 3D model to pose

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

poseId
required

string

REQUEST BODY SCHEMA: application/json

name

string

Name of the 3D model

textures >

Array of objects (Texture)

List of textures associated with the 3D model

annotations >

Array of objects (3DAnnotation)

List of 3D annotations linked to the model

id

string

Unique identifier for the object. It could not be send in POST

identifiers >

Array of objects (Identifier)

extensions >

Array of objects (Extension)

Responses

— **201** 3D model added successfully

— **401** Unauthorized

POST /poses/{poseId}/3d-models

Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

```
{  
    "name": "string",  
    - "textures": [  
        + { ... }  
    ],  
    - "annotations": [  
        + { ... }  
    ],  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Get 3D model of a pose

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

poseId
required string

3dModelId
required string

Responses

✓ 200 Successful operation

RESPONSE SCHEMA: application/json

name string
Name of the 3D model

textures > Array of objects (Texture)
List of textures associated with the 3D model

annotations > Array of objects (3DAnnotation)
List of 3D annotations linked to the model

id string

Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— **401** Unauthorized

— **404** 3D model not found

GET /poses/{poseId}/3d-models/{3dModelId}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "name": "string",  
    - "textures": [  
        + { ... }  
    ],  
    - "annotations": [  
        + { ... }  
    ],  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Update 3D model of pose

AUTHORIZATIONS >

bearerAuth

PATH PARAMETERS

poseId string
required

3dModelId string
required

REQUEST BODY SCHEMA: application/json

name string
Name of the 3D model

textures > Array of objects (Texture)
List of textures associated with the 3D model

annotations > Array of objects (3DAnnotation)
List of 3D annotations linked to the model

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** 3D model updated successfully

RESPONSE SCHEMA: application/json

name string
Name of the 3D model

textures > Array of objects (Texture)
List of textures associated with the 3D model

annotations > Array of objects (3DAnnotation)
List of 3D annotations linked to the model

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— **401** Unauthorized

— 404 3D model not found

PUT /poses/{poseId}/3d-models/{3dModelId}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "name": "string",  
    - "textures": [  
        + { ... }  
    ],  
    - "annotations": [  
        + { ... }  
    ],  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "name": "string",  
    - "textures": [  
        + { ... }  
    ],  
    - "annotations": [  
        + { ... }  
    ],  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

```
        "id": "string",
      - "identifiers": [
          + { ... }
        ],
      - "extensions": [
          + { ... }
        ]
    ]
}
```

Patch 3D model of pose

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

poseId
required string

3dModelId
required string

REQUEST BODY SCHEMA: application/json

name string
Name of the 3D model

textures > Array of objects (Texture)
List of textures associated with the 3D model

annotations > Array of objects (3DAnnotation)
List of 3D annotations linked to the model

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** 3D model patched successfully

RESPONSE SCHEMA: application/json

<code>name</code>	string Name of the 3D model
<code>textures ></code>	Array of objects (Texture) List of textures associated with the 3D model
<code>annotations ></code>	Array of objects (3DAnnotation) List of 3D annotations linked to the model
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

— **401** Unauthorized

— **404** 3D model not found

PATCH /poses/{poseId}/3d-models/{3dModelId}

Request samples

Payload

Content type
application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "name": "string",  
  - "textures": [  
    + { ... }  
  ],  
  - "annotations": [  
    + { ... }  
  ],  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "name": "string",  
  - "textures": [  
    + { ... }  
  ],  
  - "annotations": [  
    + { ... }  
  ],  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Delete 3D model of pose

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

poseId string
required

3dModelId string
required

Responses

— **204** 3D model deleted successfully

— **401** Unauthorized

— **404** 3D model not found

DELETE /poses/{poseId}/3d-models/{3dModelId}

tagPoses/{id}/contents

Get contents of a pose

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

poseId string
required

Responses

✓ **200** Successful operation

RESPONSE SCHEMA: application/json

```
Array [
    id          integer
    manuscript_id integer
    intro       string
    category    string
        Enum: "Storia e contesto" "Struttura e scrittura" "Materiali e
              tecniche esecutive" "Iconografia e iconologia" "Censure e
              modifiche" "Conservazione e restauro" "Bibliografia"
    content     string
    last_updated string <date-time>
]
```

— **401** Unauthorized

— **404** Pose not found

```
GET /poses/{poseId}/contents
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
- {  
    "id": 1,  
    "manuscript_id": 1,  
    "category": "Storia e contesto",  
    "content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",  
    "last_updated": "2021-09-01T12:00:00Z"  
}  
]
```

Add content to pose

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

poseId
required

string

REQUEST BODY SCHEMA: application/json

id

integer

manuscript_id

integer

intro

string

category

string

Enum: "Storia e contesto" | "Struttura e scrittura" | "Materiali e tecniche esecutive" | "Iconografia e iconologia" | "Censure e modifiche" | "Conservazione e restauro" | "Bibliografia"

content

string

last_updated

string <date-time>

Responses

— **201** Content added successfully

— **401** Unauthorized

POST /poses/{poseId}/contents

Request samples

Payload

Content type

application/json

Copy

```
{  
  "id": 1,  
  "manuscript_id": 1,  
  "category": "Storia e contesto",  
  "content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",
```

```
"last_updated": "2021-09-01T12:00:00Z"  
}
```

Get content of a pose

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

poseId
required string

contentId
required string

Responses

✓ 200 Successful operation

RESPONSE SCHEMA: application/json

id integer

manuscript_id integer

intro string

category string

Enum: "Storia e contesto" "Struttura e scrittura" "Materiali e tecniche esecutive" "Iconografia e iconologia" "Censure e modifiche" "Conservazione e restauro" "Bibliografia"

content string

last_updated string <date-time>

— 401 Unauthorized

— 404 Content not found

GET /poses/{poseId}/contents/{contentId}

Response samples

200

Content type

application/json

Copy

```
{  
    "id": 1,  
    "manuscript_id": 1,  
    "category": "Storia e contesto",  
    "content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",  
    "last_updated": "2021-09-01T12:00:00Z"  
}
```

Update content of pose

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

poseId
required string

contentId
required string

REQUEST BODY SCHEMA: application/json

id integer

manuscript_id integer

intro string

category string
Enum: `"Storia e contesto"` `"Struttura e scrittura"` `"Materiali e tecniche esecutive"` `"Iconografia e iconologia"` `"Censure e modifiche"` `"Conservazione e restauro"` `"Bibliografia"`

content string

last_updated string <date-time>

Responses

✓ 200 Content updated successfully

RESPONSE SCHEMA: application/json

id	integer
manuscript_id	integer
intro	string
category	string Enum: "Storia e contesto" "Struttura e scrittura" "Materiali e tecniche esecutive" "Iconografia e iconologia" "Censure e modifiche" "Conservazione e restauro" "Bibliografia"
content	string
last_updated	string <date-time>

— 401 Unauthorized

— 404 Content not found

PUT /poses/{poseId}/contents/{contentId}

Request samples

Payload

Content type

application/json

Copy

```
{  
    "id": 1,  
    "manuscript_id": 1,  
    "category": "Storia e contesto",  
    "content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",  
    "last_updated": "2021-09-01T12:00:00Z"  
}
```

Response samples

200

Content type

application/json

Copy

```
{  
    "id": 1,  
    "manuscript_id": 1,  
    "category": "Storia e contesto",  
    "content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",  
    "last_updated": "2021-09-01T12:00:00Z"  
}
```

Patch content of pose

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

poseId
required string

contentId
required string

REQUEST BODY SCHEMA: application/json

id integer

manuscript_id integer

intro string

category string
Enum: "Storia e contesto" | "Struttura e scrittura" | "Materiali e tecniche esecutive" | "Iconografia e iconologia" | "Censure e modifiche" | "Conservazione e restauro" | "Bibliografia"

content string

last_updated string <date-time>

Responses

✓ 200 Content patched successfully

RESPONSE SCHEMA: application/json

id	integer
manuscript_id	integer
intro	string
category	string Enum: "Storia e contesto" "Struttura e scrittura" "Materiali e tecniche esecutive" "Iconografia e iconologia" "Censure e modifiche" "Conservazione e restauro" "Bibliografia"
content	string
last_updated	string <date-time>

— 401 Unauthorized

— 404 Content not found

PATCH /poses/{poseId}/contents/{contentId}

Request samples

Payload

Content type

application/json

Copy

```
{  
  "id": 1,  
  "manuscript_id": 1,  
  "category": "Storia e contesto",  
  "content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",  
  "last_updated": "2021-09-01T12:00:00Z"  
}
```

Response samples

200

Content type

application/json

Copy

```
{  
  "id": 1,  
  "manuscript_id": 1,  
  "category": "Storia e contesto",  
  "content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",  
  "last_updated": "2021-09-01T12:00:00Z"  
}
```

Delete content of pose

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

poseId
required string

contentId
required string

Responses

— **204** Content deleted successfully

— **401** Unauthorized

— **404** Content not found

DELETE /poses/{poseId}/contents/{contentId}

Projects

All Projects related operations

Create a new project

This operation creates a new project.

AUTHORIZATIONS: > *bearerAuth*

REQUEST BODY SCHEMA: application/json

periods > object (Period)

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ 201 Project created

RESPONSE SCHEMA: application/json

periods > object (Period)

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

— 401 Unauthorized

POST /projects

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  - "periods": {
      "start": "2019-08-24T14:15:22Z",
      "end": "2019-08-24T14:15:22Z"
    },
    "id": "string",
  - "identifiers": [
      + { ... }
    ],
  - "extensions": [
      + { ... }
    ]
}
```

Response samples

201

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  - "periods": {  
    "start": "2019-08-24T14:15:22Z",  
    "end": "2019-08-24T14:15:22Z"  
  },  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Get all projects

This operation retrieves all projects.

AUTHORIZATIONS: > *bearerAuth*

Responses

✓ **200** A list of projects

RESPONSE SCHEMA: `application/json`

Array [

periods > object (Period)

id string

 Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

]

— **401** Unauthorized

GET /projects

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
  - {  
    + "periods": { ... },  
    "id": "string",  
    + "identifiers": [ ... ],  
    + "extensions": [ ... ]  
  }  
]
```

Get a project

This operation retrieves a project based on its ID.

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

id
required string

Responses

✓ 200 A project

RESPONSE SCHEMA: application/json

periods > object (Period)

id string

Unique identifier for the object. It could not be send in POST

`identifiers` > Array of objects (Identifier)

`extensions` > Array of objects (Extension)

— 401 Unauthorized

— 404 Project not found

GET /projects/{id}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  - "periods": {  
      "start": "2019-08-24T14:15:22Z",  
      "end": "2019-08-24T14:15:22Z"  
    },  
    "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Update a project

This operation updates a project based on its ID.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

`id`
required string

periods >	object (Period)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ 200 Project updated

periods >	object (Period)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— 401 Unauthorized

— 404 Project not found

PATCH /projects/{id}

Request samples

Payload

Content type
application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  - "periods": {  
      "start": "2019-08-24T14:15:22Z",  
      "end": "2019-08-24T14:15:22Z"  
    },  
    "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  - "periods": {  
      "start": "2019-08-24T14:15:22Z",  
      "end": "2019-08-24T14:15:22Z"  
    },  
    "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Delete a project

This operation deletes a project based on its ID.

[AUTHORIZATIONS](#) >

bearerAuth

[PATH PARAMETERS](#)

id required	string
------------------------------	--------

Responses

— **204** Project deleted

— **401** Unauthorized

— **404** Project not found

DELETE /projects/{id}

Replace a project

This operation replaces a project based on its ID.

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

id
required string

REQUEST BODY SCHEMA: application/json

periods > object (Period)

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

Responses

✓ **200** Project replaced

RESPONSE SCHEMA: application/json

periods >	object (Period)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— **401 Unauthorized**

— **404 Project not found**

PUT /projects/{id}

Request samples

Payload

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  - "periods": {  
      "start": "2019-08-24T14:15:22Z",  
      "end": "2019-08-24T14:15:22Z"  
    },  
    "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  - "periods": {  
    "start": "2019-08-24T14:15:22Z",  
    "end": "2019-08-24T14:15:22Z"  
  },  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Texture

All Texture related operations

Retrieve all textures

AUTHORIZATIONS: > *bearerAuth*

QUERY PARAMETERS

page	integer <code>>= 0</code> Default: <code>0</code>	Numero della pagina (facoltativo, predefinito: 0). L'indice parte da 0.
size	integer <code>>= 1</code> Default: <code>20</code>	Numero di elementi per pagina (facoltativo, predefinito: 20). Deve essere maggiore di 0.
sort	string Example: <code>sort=name,asc</code>	Campo per l'ordinamento (facoltativo). Formato: <code>campo,direzione</code> (es. <code>name,asc</code>). Direzioni supportate: <code>asc</code> , <code>desc</code> .

HEADER PARAMETERS

Accept	string Enum: <code>"application/json"</code> <code>"text/turtle"</code>
--------	--

Responses

✓ 200 List of all available textures

RESPONSE SCHEMA: [application/json](#)

Array [

type	string
	Enum: "RGB" "IR1" "IR2" "UV"
	Texture type (RGB, IR1, IR2, UV)
fileUrl	string <uri>
	URL of the texture image
id	string
	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

]

— **401** Unauthorized

— **406** Formato non supportato

GET /texture

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
  - {  
    "type": "RGB",  
    "fileUrl": "http://example.com",  
    "id": "string",  
    + "identifiers": [ ... ],  
    + "extensions": [ ... ]  
  }  
]
```

Add a texture to a 3D model

AUTHORIZATIONS: > *bearerAuth*

REQUEST BODY SCHEMA: application/json

<code>type</code>	string Enum: <code>"RGB"</code> <code>"IR1"</code> <code>"IR2"</code> <code>"UV"</code> Texture type (RGB, IR1, IR2, UV)
<code>fileUrl</code>	string <uri> URL of the texture image
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers</code> >	Array of objects (Identifier)
<code>extensions</code> >	Array of objects (Extension)

Responses

— **201** Texture successfully added

— **401** Unauthorized

POST /texture

Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

```
{  
  "type": "RGB",  
  "fileUrl": "http://example.com",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Get a texture

AUTHORIZATIONS:

bearerAuth

PATH PARAMETERS

id
required string

Responses

✓ **200** Texture retrieved successfully

RESPONSE SCHEMA: application/json

type string
Enum: `"RGB"` `"IR1"` `"IR2"` `"UV"`
Texture type (RGB, IR1, IR2, UV)

fileUrl string <uri>
URL of the texture image

id string
Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

`extensions` >

Array of objects (Extension)

— **401** Unauthorized

— **404** Texture not found

GET /texture/{id}

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "type": "RGB",  
    "fileUrl": "http://example.com",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Replace a texture

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

id string
required

REQUEST BODY SCHEMA: application/json

type string
Enum: `"RGB"` `"IR1"` `"IR2"` `"UV"`
Texture type (RGB, IR1, IR2, UV)

fileUrl	string <uri> URL of the texture image
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ **200** Texture replaced successfully

RESPONSE SCHEMA: application/json

type	string Enum: "RGB" "IR1" "IR2" "UV" Texture type (RGB, IR1, IR2, UV)
fileUrl	string <uri> URL of the texture image
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— **401** Unauthorized

— **404** Texture not found

PUT /texture/{id}

Request samples

Payload

Content type
application/json

Copy Expand all Collapse all

```
{  
  "type": "RGB",  
  "fileUrl": "http://example.com",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

200

Content type

application/json

Copy Expand all Collapse all

```
{  
  "type": "RGB",  
  "fileUrl": "http://example.com",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Update a texture

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id
required string

REQUEST BODY SCHEMA: application/json

type string
Enum: `"RGB"` `"IR1"` `"IR2"` `"UV"`

Texture type (RGB, IR1, IR2, UV)

fileUrl	string <uri> URL of the texture image
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

Responses

✓ 200 Texture updated successfully

RESPONSE SCHEMA: application/json

type	string Enum: <code>"RGB"</code> <code>"IR1"</code> <code>"IR2"</code> <code>"UV"</code> Texture type (RGB, IR1, IR2, UV)
fileUrl	string <uri> URL of the texture image
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

— 401 Unauthorized

— 404 Texture not found

PATCH /texture/{id}

Request samples

Payload

Content type
application/json

Copy Expand all Collapse all

```
{  
  "type": "RGB",  
  "fileUrl": "http://example.com",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Response samples

200

Content type

application/json

Copy Expand all Collapse all

```
{  
  "type": "RGB",  
  "fileUrl": "http://example.com",  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Delete a texture

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id required	string
------------------------------	--------

Responses

— **204** Texture deleted successfully

— **401** Unauthorized

— **404** Texture not found

DELETE /texture/{id}

Authors

All Author related operations

Get all manuscripts of an author

This operation retrieves all manuscripts for an author.

AUTHORIZATIONS: > *bearerAuth*

PATH PARAMETERS

id
required string

HEADER PARAMETERS

Accept string
Enum: `"application/json"` `"text/turtle"`
Formato di risposta desiderato. Valori supportati: `application/json`,
`text/turtle`

Responses

✓ **200** Successful operation

RESPONSE SCHEMA: `application/json`

id	string
title	string
description	string
previewImage	string <uri>
events >	Array of objects (Reference)
projects >	Array of objects (Reference)
locations >	Array of objects (Reference)
scenes >	Array of objects (Reference)
authors >	Array of objects (Reference)
editors >	Array of objects (Reference)
status	<p>string</p> <p>Enum: <code>"draft"</code> <code>"in_review"</code> <code>"revisions_required"</code> <code>"accepted"</code> <code>"in_editing"</code> <code>"published"</code> <code>"rejected"</code> <code>"withdrawn"</code></p> <p>Stato attuale del manoscritto:</p> <ul style="list-style-type: none"> • <code>draft</code>: Bozza • <code>in_review</code>: In revisione • <code>revisions_required</code>: Revisione necessaria • <code>accepted</code>: Accettato • <code>in_editing</code>: In fase di editing • <code>published</code>: Pubblicato • <code>rejected</code>: Rifiutato • <code>withdrawn</code>: Ritirato
]	

— **401** Unauthorized

— **404** Author not found

— **406** Formato non supportato

GET /authors/{id}/manuscripts

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
- {  
    "id": "string",  
    "title": "string",  
    "description": "string",  
    "previewImage": "http://example.com",  
+ "events": [ ... ],  
+ "projects": [ ... ],  
+ "locations": [ ... ],  
+ "scenes": [ ... ],  
+ "authors": [ ... ],  
+ "editors": [ ... ],  
    "status": "draft"  
}  
]
```

Editors

All editors related operations

Get all manuscripts of an editor

This operation retrieves all manuscripts of an editor.

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

id
required

string

HEADER PARAMETERS

Accept

string

Enum: ["application/json"](#) ["text/turtle"](#)

Responses

✓ 200 Successful operation

RESPONSE SCHEMA: `application/json`

Array [

id	string
title	string
description	string
previewImage	string <uri>
events >	Array of objects (Reference)
projects >	Array of objects (Reference)
locations >	Array of objects (Reference)
scenes >	Array of objects (Reference)
authors >	Array of objects (Reference)
editors >	Array of objects (Reference)
status	string Enum: <code>"draft"</code> <code>"in_review"</code> <code>"revisions_required"</code> <code>"accepted"</code> <code>"in_editing"</code> <code>"published"</code> <code>"rejected"</code> <code>"withdrawn"</code> Stato attuale del manoscritto: <ul style="list-style-type: none">• <code>draft</code>: Bozza• <code>in_review</code>: In revisione• <code>revisions_required</code>: Revisione necessaria• <code>accepted</code>: Accettato• <code>in_editing</code>: In fase di editing• <code>published</code>: Pubblicato• <code>rejected</code>: Rifiutato• <code>withdrawn</code>: Ritirato

]

— 401 Unauthorized

— 404 Editor not found

GET /editors/{id}/manuscripts

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
- {  
    "id": "string",  
    "title": "string",  
    "description": "string",  
    "previewImage": "http://example.com",  
    + "events": [ ... ],  
    + "projects": [ ... ],  
    + "locations": [ ... ],  
    + "scenes": [ ... ],  
    + "authors": [ ... ],  
    + "editors": [ ... ],  
    "status": "draft"  
}  
]
```

Get all scenes of an editor

This operation retrieves all scenes for an editor.

AUTHORIZATIONS: >

bearerAuth

PATH PARAMETERS

id
required string

HEADER PARAMETERS

Accept string

Responses

✓ 200 Successful operation

RESPONSE SCHEMA: application/json

Array [

title	string Title of the scene
description	string Description of the scene
panorama	string <uri> Panoramic view of the scene
previewImage	string <uri> Preview image of the scene
authors >	Array of objects (Reference) List of authors of the scene
editors >	Array of objects (Reference) List of editors
visibility	string Enum: "public" "private" "restricted" Visibility status of the scene
project >	object (Reference) Una reference da una risorsa a un'altra
manuscript >	object (Reference) Una reference da una risorsa a un'altra
poses >	Array of objects (Reference) List of poses
annotations3D >	Array of objects (Reference) List of 3D annotations

]

— 401 Unauthorized

— **404** Editor not found

— **406** Formato non supportato

GET /editors/{id}/scenes

Response samples

200

Content type

application/json

[Copy](#) [Expand all](#) [Collapse all](#)

```
[  
  - {  
    "title": "string",  
    "description": "string",  
    "panorama": "http://example.com",  
    "previewImage": "http://example.com",  
    + "authors": [ ... ],  
    + "editors": [ ... ],  
    "visibility": "public",  
    + "project": { ... },  
    + "manuscript": { ... },  
    + "poses": [ ... ],  
    + "annotations3D": [ ... ]  
  }  
]
```

API Reference

Reference

reference	string
required	Literal reference, URL relativa, interna o assoluta
type	string <uri>
	Tipo della risorsa a cui fa riferimento (es. "Patient")
identifier >	object
	Riferimento logico quando non si conosce il riferimento letterale
display	string
	Testo alternativo per la risorsa

Copy Expand all Collapse all

```
{
  "reference": "Risorsa/1234",
  "type": "Risorsa",
  - "identifier": {
    "system": "http://digilab.com/risorsa-ids",
    "value": "1234"
  },
  "display": "descrizione della risorsa"
}
```

Base Entity

id	string
	Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "id": "string",  
  - "identifiers": [  
    + { ... }  
  ],  
  - "extensions": [  
    + { ... }  
  ]  
}
```

Location

A location object extending the [Base Entity](#)

name	string
description	string
address	string
coordinates	Array of numbers = 2 items
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "name": "string",  
  "description": "string",  
  "address": "string",
```

```
- "coordinates": [
  0,
  0
],
"id": "string",
- "identifiers": [
  + { ... }
],
- "extensions": [
  + { ... }
]
}
```

Person

A person object extending the [Base Entity](#)

firstName required	string The person's first name
lastName required	string The person's last name
department required	string The person's department
birthDate	string <date> The person's birthDate
deathDate	string <date> The person's deathDate
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    "firstName": "string",  
    "lastName": "string",  
    "department": "string",  
    "birthDate": "2019-08-24",  
    "deathDate": "2019-08-24",  
    "id": "string",  
    - "identifiers": [  
        + { ... }  
    ],  
    - "extensions": [  
        + { ... }  
    ]  
}
```

Project

A project object extending the [Base Entity](#)

periods > object (Period)

id string

Unique identifier for the object. It could not be send in POST

identifiers > Array of objects (Identifier)

extensions > Array of objects (Extension)

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
    - "periods": {  
        "start": "2019-08-24T14:15:22Z",  
        "end": "2019-08-24T14:15:22Z"  
    },  
    "id": "string",
```

```
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Event

An event object extending the [Base Entity](#)

<code>type</code>	string
<code>Period ></code>	object (Period)
<code>Location ></code>	object (Location)
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "type": "string",
- "Period": {
    "start": "2019-08-24T14:15:22Z",
    "end": "2019-08-24T14:15:22Z"
},
```

```

- "Location": {
    "name": "string",
    "description": "string",
    "address": "string",
    + "coordinates": [ ... ],
    "id": "string",
    + "identifiers": [ ... ],
    + "extensions": [ ... ]
},
"id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}

```

Manuscript

A manuscript object extending the [Base Entity](#)

<code>title</code>	string
<code>description</code>	string
<code>creationDate</code>	string <date-time>
<code>publicationDate</code>	string <date-time>
<code>previewImage</code>	string <uri>
<code>events ></code>	Array of objects (Reference)
<code>projects ></code>	Array of objects (Reference)
<code>locations ></code>	Array of objects (Location)
<code>contents ></code>	Array of objects (LighContent)
<code>authors ></code>	Array of objects (Reference)
<code>editors ></code>	Array of objects (Reference)

`illuminated-scenes` > Array of objects (LightIlluminatedScene)

`status`

string

Enum: "draft" "in_review" "revisions_required" "accepted"
"in_editing" "published" "rejected" "withdrawn"

Stato attuale del manoscritto:

- `draft`: Bozza
- `in_review`: In revisione
- `revisions_required`: Revisione necessaria
- `accepted`: Accettato
- `in_editing`: In fase di editing
- `published`: Pubblicato
- `rejected`: Rifiutato
- `withdrawn`: Ritirato

`id`

string

Unique identifier for the object. It could not be send in POST

`identifiers` >

Array of objects (Identifier)

`extensions` >

Array of objects (Extension)

[Copy](#) [Expand all](#) [Collapse all](#)

{

 "title": "string",
 "description": "string",
 "creationDate": "2019-08-24T14:15:22Z",
 "publicationDate": "2019-08-24T14:15:22Z",
 "previewImage": "<http://example.com>",

- "events": [

 + { ... }

],

- "projects": [

 + { ... }

],

- "locations": [

 + { ... }

],

- "contents": [

 + { ... }

],

- "authors": [

 + { ... }

],

```
- "editors": [
  + { ... }
],
- "illuminated-scenes": [
  + { ... }
],
  "status": "draft",
  "id": "string",
- "identifiers": [
  + { ... }
],
- "extensions": [
  + { ... }
]
}
```

Light Manuscript

A Light manuscript object

id	string
title	string
description	string
previewImage	string <uri>
events >	Array of objects (Reference)
projects >	Array of objects (Reference)
locations >	Array of objects (Reference)
scenes >	Array of objects (Reference)
authors >	Array of objects (Reference)
editors >	Array of objects (Reference)
status	string Enum: <code>"draft"</code> <code>"in_review"</code> <code>"revisions_required"</code> <code>"accepted"</code> <code>"in_editing"</code> <code>"published"</code> <code>"rejected"</code> <code>"withdrawn"</code>

Stato attuale del manoscritto:

- `draft`: Bozza
 - `in_review`: In revisione
 - `revisions_required`: Revisione necessaria
 - `accepted`: Accettato
 - `in_editing`: In fase di editing
 - `published`: Pubblicato
 - `rejected`: Rifiutato
 - `withdrawn`: Ritirato
-

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "id": "string",  
  "title": "string",  
  "description": "string",  
  "previewImage": "http://example.com",  
  - "events": [  
    + { ... }  
  ],  
  - "projects": [  
    + { ... }  
  ],  
  - "locations": [  
    + { ... }  
  ],  
  - "scenes": [  
    + { ... }  
  ],  
  - "authors": [  
    + { ... }  
  ],  
  - "editors": [  
    + { ... }  
  ],  
  "status": "draft"  
}
```

title	string Title of the scene
description	string Description of the scene
panorama	string <uri> Panoramic view of the scene
previewImage	string <uri> Preview image of the scene
authors >	Array of objects (Reference) List of authors of the scene
editors >	Array of objects (Reference) List of editors
visibility	string Enum: <code>"public"</code> <code>"private"</code> <code>"restricted"</code> Visibility status of the scene
project >	object (Reference) Una reference da una risorsa a un'altra
manuscript >	object (Reference) Una reference da una risorsa a un'altra
poses >	Array of objects (Pose) List of poses
annotations3D >	Array of objects (3DAnnotation) List of 3D annotations

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "title": "string",
  "description": "string",
  "panorama": "http://example.com",
  "previewImage": "http://example.com",
  - "authors": [
    + { ... }
  ],
}
```

```

- "editors": [
  + { ... }
],
"visibility": "public",
- "project": {
  "reference": "Risorsa/1234",
  "type": "Risorsa",
  + "identifier": { ... },
  "display": "descrizione della risorsa"
},
- "manuscript": {
  "reference": "Risorsa/1234",
  "type": "Risorsa",
  + "identifier": { ... },
  "display": "descrizione della risorsa"
},
- "poses": [
  + { ... }
],
- "annotations3D": [
  + { ... }
]
}

```

Light Illuminated Scene

A light illuminated manuscript scene object

title	string Title of the scene
description	string Description of the scene
panorama	string <uri> Panoramic view of the scene
previewImage	string <uri> Preview image of the scene
authors >	Array of objects (Reference) List of authors of the scene

`editors` > Array of objects (Reference)

List of editors

`visibility` string

Enum: `"public"` `"private"` `"restricted"`

Visibility status of the scene

`project` > object (Reference)

Una reference da una risorsa a un'altra

`manuscript` > object (Reference)

Una reference da una risorsa a un'altra

`poses` > Array of objects (Reference)

List of poses

`annotations3D` > Array of objects (Reference)

List of 3D annotations

[Copy](#) [Expand all](#) [Collapse all](#)

```
{  
  "title": "string",  
  "description": "string",  
  "panorama": "http://example.com",  
  "previewImage": "http://example.com",  
  - "authors": [  
    + { ... }  
  ],  
  - "editors": [  
    + { ... }  
  ],  
  "visibility": "public",  
  - "project": {  
    "reference": "Risorsa/1234",  
    "type": "Risorsa",  
    + "identifier": { ... },  
    "display": "descrizione della risorsa"  
  },  
  - "manuscript": {  
    "reference": "Risorsa/1234",  
    "type": "Risorsa",  
    + "identifier": { ... },  
    "display": "descrizione della risorsa"  
  },  
}
```

```
- "poses": [
  + { ... }
],
- "annotations3D": [
  + { ... }
]
}
```

Pose

A pose object extending the [Base Entity](#)

name	string Name of the pose
3d-models >	Array of objects (3DModel) List of 3D models associated with the pose
contents >	Array of objects (LighContent)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "name": "string",
- "3d-models": [
  + { ... }
],
- "contents": [
  + { ... }
],
  "id": "string",
```

```
- "identifiers": [
  + { ... }
],
- "extensions": [
  + { ... }
]
}
```

Texture

A texture object extending the [Base Entity](#)

type	string Enum: "RGB" "IR1" "IR2" "UV" Texture type (RGB, IR1, IR2, UV)
fileUrl	string <uri> URL of the texture image
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "type": "RGB",
  "fileUrl": "http://example.com",
  "id": "string",
  - "identifiers": [
    + { ... }
  ],
  - "extensions": [
    + { ... }
  ]
}
```

3D Model

A 3D model object extending the [Base Entity](#)

<code>name</code>	string Name of the 3D model
<code>textures ></code>	Array of objects (Texture) List of textures associated with the 3D model
<code>annotations ></code>	Array of objects (3DAnnotation) List of 3D annotations linked to the model
<code>id</code>	string Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)
<code>extensions ></code>	Array of objects (Extension)

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "name": "string",
  - "textures": [
    + { ... }
  ],
}
```

```
- "annotations": [
  + { ... }
],
"id": "string",
- "identifiers": [
  + { ... }
],
- "extensions": [
  + { ... }
]
}
```

Annotation

An annotation object extending the [Base Entity](#)

type	string Enum: "text" "image" "video" "audio" Annotation type (text, image, video, audio)
content	string Content of the annotation (text, image URL, video URL, audio URL)
id	string Unique identifier for the object. It could not be send in POST
identifiers >	Array of objects (Identifier)
extensions >	Array of objects (Extension)

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "type": "text",
  "content": "string",
  "id": "string",
- "identifiers": [
  + { ... }
],
```

```
- "extensions": [
  + { ... }
]
}
```

3d Annotation

A 3D annotation object extending the [Base Entity](#)

<code>region</code>	string	3D region where the annotation applies
<code>annotation ></code>	object (Annotation)	
<code>model ></code>	object (3DModel)	
<code>id</code>	string	Unique identifier for the object. It could not be send in POST
<code>identifiers ></code>	Array of objects (Identifier)	
<code>extensions ></code>	Array of objects (Extension)	

[Copy](#) [Expand all](#) [Collapse all](#)

```
{
  "region": "string",
  - "annotation": {
    "type": "text",
    "content": "string",
    "id": "string",
    + "identifiers": [ ... ],
    + "extensions": [ ... ]
  },
}
```

```
- "model": {
    "name": "string",
    + "textures": [ ... ],
    + "annotations": [ ... ],
    "id": "string",
    + "identifiers": [ ... ],
    + "extensions": [ ... ]
},
"id": "string",
- "identifiers": [
    + { ... }
],
- "extensions": [
    + { ... }
]
}
```

Content

A content object extending the [Base Entity](#)

<code>id</code>	integer
<code>manuscript_id</code>	integer
<code>intro</code>	string
<code>category</code>	string Enum: <code>"Storia e contesto"</code> <code>"Struttura e scrittura"</code> <code>"Materiali e tecniche esecutive"</code> <code>"Iconografia e iconologia"</code> <code>"Censure e modifiche"</code> <code>"Conservazione e restauro"</code> <code>"Bibliografia"</code>
<code>content</code>	string
<code>last_updated</code>	string <date-time>

Copy

```
{
  "id": 1,
```

```
"manuscript_id": 1,  
"category": "Storia e contesto",  
"content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit.",  
"last_updated": "2021-09-01T12:00:00Z"  
}
```

LighContent

A content input object

intro	string
category required	string Enum: <input type="button" value="Storia e contesto"/> <input type="button" value="Struttura e scrittura"/> <input type="button" value="Materiali e tecniche esecutive"/> <input type="button" value="Iconografia e iconologia"/> <input type="button" value="Censure e modifiche"/> <input type="button" value="Conservazione e restauro"/> <input type="button" value="Bibliografia"/>
content required	string

Copy

```
{  
  "category": "Storia e contesto",  
  "content": "Lorem ipsum dolor sit amet, consectetur adipiscing elit."  
}
```

