Contents

API Object: marketplace.profile

Definition

Use the `/profile` API to update or read a profile or list of profiles in the system.

Resource Information

The following properties appear in the Type JSON:

Request URL

`https://localhost:8443/marketplace/api/profile`

This placeholder URL will vary depending upon your deployment. Be mindful that `https://localhost:8443/marketplace` is an example "base/context/domain" where your WAR is deployed.

Request Methods

PUT,
GET,

Each method will be explained in the following sections:

PUT

Use this call to **update** a profile in the system.

Request

URL

`https://localhost:8443/marketplace/api/profile/{id}`

Heading

```
Accept = application/JSON
Content-Type = application/JSON
```

Body

```
{
"id":3,
"bio":"Bio",
"displayName":"Test Admin 1",
"uuid":"unknown",
"username":"testAdmin1",
"email":"testAdmin1@nowhere.com"
}
```

Response Code:

200

Response

```
{
    "id":3,
    "displayName":"Test User 1",
    "username":"testUser1",
    "createdDate":"2015-07-14T20:38:01.529+0000",
    "launchInWebtop":true,
    "organizations":[

    ],
    "stewardedOrganizations":[

    ],
    "highestRole":"USER",
    "bio":"Admin updating user bio",
    "lastLogin":"2015-07-30T17:22:15.710+0000",
    "email":"testUser1@nowhere.com",
    "_links":{
        "curies":{
            "href":"http://ozoneplatform.org/docs/rels/{rel}",
            "name":"ozp",
            "templated":true
        },
        "ozp:application-library":{
            "href":"https://localhost:8443/marketplace/api/profile/3/library"
        },
        "ozp:user-data":{
            "href":"https://localhost:8443/marketplace/api/profile/3/data"
        },
        "self":{
            "href":"https://localhost:8443/marketplace/api/profile/3"
        }
    }
}
```

Requirements

GET

Use this call to **read or view** a profile or all the profiles in the system.

Request

If you want to see a list of all the profiles in the system, enter:
`https://localhost:8443/marketplace/api/profile/`

However, to view metadata about only one profile, enter:
`https://localhost:8443/marketplace/api/profile/{id}`

Marketplace (i.e., Center) returns the representation of the profile that matches the{id}, as shown in the Response for one profile id.

Response Code:

200

Response for one profile id

```
{
    "id":3,
    "displayName":"Test User 1",
    "username":"testUser1",
    "createdDate":"2015-07-14T20:38:01.529+0000",
    "launchInWebtop":true,
    "organizations":[

    ],
    "stewardedOrganizations":[

    ],
    "highestRole":"USER",
    "bio":"Admin updating user bio",
    "lastLogin":"2015-07-30T13:22:35.163+0000",
    "email":"testUser1@nowhere.com",
    "_links":{
        "curies":{
            "href":"http://ozoneplatform.org/docs/rels/{rel}",
            "name":"ozp",
            "templated":true
        },
        "ozp:application-library":{
            "href":"https://localhost:8443/marketplace/api/profile/3/library"
        },
        "ozp:user-data":{
            "href":"https://localhost:8443/marketplace/api/profile/3/data"
        },
        "self":{
            "href":"https://localhost:8443/marketplace/api/profile/3"
        }
    }
}
```

Requirements

Possible Errors

This table lists common errors. Other errors may occur but these are the most likely:

Error Code
Error
Troubleshooting

404
Invalid Profile
The profile you're looking for does not exist. If the person's account was deleted you will not see any information about their profile.

400
Invalid bio
The bio section of the profile must be 256 characters or less.

400
Agency cannot be created or updated.
The agency must include all required fields.

API Object: marketplace.scorecard

Definition

Use the `/scorecard` API to create, update, read or delete scorecard data in the system.

Resource Information

The following properties appear in the scorecard JSON:

Request URL

`https://localhost:8443/marketplace/api/scorecard`

This placeholder URL will vary depending upon your deployment. Be mindful that `https://localhost:8443/marketplace` is an example "base/context/domain" where your WAR is deployed.

Request Methods

POST,
PUT,
GET,
DELETE

Each method will be explained in the following sections:

POST

Use this call to **create** a scorecard question in the system.

Request

URL

`https://localhost:8443/marketplace/api/scorecard`

Heading

```
Accept = application/JSON
Content-Type = application/JSON
```

Body

```
{
    "description":"DESCRIPTION",
    "image":null,
    "showOnListing":false,
    "question":"Description"
}
```

Response Code:

201

Response

```
{
    "question":"Description",
    "description":"DESCRIPTION",
    "image":null,
    "showOnListing":false,
    "_links":{
        "self":{
            "href":"https://localhost:8443/marketplace/api/scorecard/1"
        }
    }
}
```

Requirements

PUT

Use this call to **update** a scorecard question in the system.

Request

URL

`https://localhost:8443/marketplace/api/scorecard/{id}`

Heading

```
Accept = application/JSON
Content-Type = application/JSON
```

Body

```
{
    "id":27,
    "description":"DESCRIPTION",
    "image":null,
    "showOnListing":false,
    "question":"Description"
}
```

Response Code:

200

Response

```
{
    "question":"Description",
    "description":"DESCRIPTION",
    "image":null,
    "showOnListing":false,
    "_links":
    {
        "self":
        {
            "href":"https://localhost:8443/marketplace/api/scorecard/27"
        }
    }
}
```

Requirements

GET

Use this call to **read or view** a scorecard question in the system.

Request

`https://localhost:8443/marketplace/api/scorecard/{id}`

Response Code:

200

Response

```
{
    "question":"123456",
    "description":"DESCRIPTION",
    "image":null,
    "showOnListing":false,
    "_links":
    {
        "self":
        {
            "href":"https://localhost:8443/marketplace/api/scorecard/2"
        }
    }
}
```

Requirements

DELETE

Use this call to remove a scorecard question from the system.

Requirements

`https://localhost:8443/marketplace/api/scorecard/{id}`

Response Code:

204

Response

`no content`

Requirements

Possible Errors

This table lists common errors. Other errors may occur but these are the most likely:

Error Code
Error
Troubleshooting

400
System-generated Scorecard questions cannot be deleted.
If you try to change them, the system returns an error.

400
User cannot create a scorecard question without giving it a name that is less than 256 characters.
Give the question a name that is less than 256.

400
The description is too long.
The description must be less than 500 characters.

400
The question must include an image icon.
This icon is used to identify the question in the listing description. It must be a standard image format.

400
The question must include an image icon.
This icon is used to identify the question in the listing description.

API Object: marketplace.type

Definition

Use the `/type` API to create, update, read or delete a type or list of types in the system.

Resource Information

The following properties appear in the Type JSON:

Request URL

`https://localhost:8443/marketplace/api/type`

This placeholder URL will vary depending upon your deployment. Be mindful that `https://localhost:8443/marketplace` is an example "base/context/domain" where your WAR is deployed.

Request Methods

POST,
PUT,

Each method will be explained in the following sections:

POST

Use this call to **create** a type in the system.

Request

URL

`https://localhost:8443/marketplace/api/type`

Heading

```
Accept = application/JSON
Content-Type = application/JSON
```

Body

```
{
    "title":"Title",
    "description":"This is a description"
}
```

Response Code:

201

Response

```
{
    "id":65,
    "description":"This is a description",
    "title":"Title",
    "_links":{
        "self":{
            "href":"https://localhost:8443/marketplace/api/type/65"
        }
    }
}
```

Requirements

PUT

Use this call to **update** a type in the system.

Request

URL

```
https://localhost:8443/marketplace/api/type/{id}
```

Heading

```
Accept = application/JSON
Content-Type = application/JSON
```

Body

```
{
    "id":1,
    "title":"Widget",
    "description":"This is the widget description."
}
```

Response Code:

200

Response

```
{
    "id":1,
    "description":"This is the widget description.",
    "title":"Widget",
    "_links":{
        "self":{
            "href":"https://localhost:8443/marketplace/api/type/1"
        }
    }
}
```

Requirements

GET

Use this call to **read or view** a type or all the types in the system.

Request

If you want to see a list of all the types in the system, enter:
`https://localhost:8443/marketplace/api/type/`

However, to view metadata about only one type, enter:
`https://localhost:8443/marketplace/api/type/{id}`

Marketplace (i.e., Center) returns the representation of the type that matches the{id}, as shown in the Response for one type id.

Response Code:

200

Response for one type id

```json
{
    "id":2,
    "description":"A web app",
    "title":"Web Application",
    "_links":{
        "self":{
            "href":"https://localhost:8443/marketplace/api/type/2"
        }
    }
}
```

Requirements

Optional Parameters

If you want to limit the responses, for example, only return 5, use Optional Parameters which are included in the code as `@QueryParam`:

offset–an integer offset
Example: `https://localhost:8443/marketplace/api/type?offset=5`

max–maximum number of type ids your call will return
Example: `https://localhost:8443/marketplace/api/type?max=5`

DELETE

Use this call to remove a type from the system.

Requirements

`https://localhost:8443/marketplace/api/type/{id}`

Response Code:

204

Response

no content

Requirements

Possible Errors

This table lists common errors. Other errors may occur but these are the most likely:

Error Code
Error
Troubleshooting

400
Type cannot be deleted.
See if it is associated with a listing. If any listing is assigned to the type, you cannot delete that type.

403
User cannot create, edit, delete a type.
Only administrators can create, edit, delete types.

400
Type cannot be created or updated.
The type must include all required fields.

400
Type must have a unique name.
If the type name is not unique, a validation error occurs when you try to save.