

RESTful API Reference Documentation Sample

Note: This writing sample documents a POST and a GET request for a (fictional) software company, "SoundDate," where users match with each other for blind dates based on the sound of each other's voice.

The POST request documented below is how a user uploads an audio file to his or her profile on the app. The GET request is how a user retrieves an audio file that a different user uploaded to his or her profile.

Upload an audio file to your profile

This is a POST request for when a user uploads an audio file to his or her profile.

Note: The user's **id** is not specified, since in this case, the user's **id** will have been determined via authorization.

URL

POST `https://api.sounddate.com/profile/sound`

Headers

Header Name	Description	Required	Values
Bearer	The user's access token	Required	<i>See section on Authorization</i>
Content-Type	The audio file's format	Optional	Can be audio/mpeg or audio/x-wav Default value: audio/mpeg
Accept	The API response format	Optional	Can be application/xml or

application/json

Default value:

application/json

POST Body

Note: The POST body is simply the sound file to be uploaded. Sound files must be 5 minutes (300 seconds) or shorter.

Sample Request

POST https://api.sounddate.com/profile/sound/

Bearer: 234c234892e08f

Content-Type: audio/mpeg

Accept: application/json

{sound_file}

Response

Element	Description	Type	Notes
id	The id of the audio file to be uploaded	int	
length	The length of the audio file to be uploaded	float	Represents the length of the sound file in seconds

Sample Response

```
{  
  "id": 3543,  
  "length": 19.8
```

}

Retrieve an audio file from another user's profile

This is a GET request for when a user retrieves an audio file from another's user's profile.

URL

GET https://api.sounddate.com/user/{user_id}/profile/sound

Note: {user_id} represents the **id** of the user whose profile contains the audio file.

Query Parameters

Parameter	Description	Type	Required	Notes
sortOrder	Determines the order in which audio files are returned	string	Optional	Can be mostRecent , earliest , shortest , or longest Default: mostRecent

Note: **mostRecent** returns audio files in order of most recently uploaded to earliest upload. **earliest** returns audio files in order of earliest upload to most recent. **shortest** returns audio files in order of shortest to longest length. **longest** returns sound files in order of longest to shortest length.

Headers

Header Name	Description	Required	Values
Bearer	The user's access token	Required	<i>See section on Authorization</i>
Accept	The API response format	Optional	Can be application/xml or application/json Default value: application/json

Sample Request

```
GET
https://api.sounddate.com/user/4321/profile/sound?sortOrder=shortest
```

```
Bearer: 234c234892e08f
```

```
Accept: application/json
```

```
{
  "soundFiles": [
    {
      "id": 23456,
      "url": "https://api.sounddate.com/profile/sound/23456.mp3",
      "length": 11.2
    },
    {
      "id": 24559,
      "url": "https://api.sounddate.com/profile/sound/24559.mp3",
      "length": 19.8,
    }
  ]
}
```

```
]
}
```

Response

Element	Description	Type	Notes
soundFiles	List of audio file information	array	
id	The id of the audio file returned	int	
url	The location of the audio file returned	string	
length	The length of the audio file returned	float	Represents the length of the audio in seconds

Sample Response

```
{
  "soundFiles": [
    {
      "id": 23456,
      "url": "https://api.sounddate.com/profile/sound/23456.mp3",
      "length": 11.2
    },
    {
      "id": 24559,
      "url": "https://api.sounddate.com/profile/sound/24559.mp3",
      "length": 19.8,
    }
  ]
}
```

}

Status Codes and Errors

The following table lists returned HTTP status codes.

Code	Description	Notes
200	OK	Successful GET or POST request
401	Unauthorized	User's access token is invalid
413	Payload Too Large/Request Entity Too Large	User attempted to upload an audio file that exceeds the maximum length