fyyd api

Documentations version: 0.2

This is the initial documentation for the API to fyyd, the podcast search engine, directory and slow cooker.

Take my apologies for this simple markdown, I tried to use something more fancy, but I took an arrow to my knee.

Please be aware: This version is most likely to change in the very near future. It's my first kind of that API and I'm still learning how to do things the more or less correct way. All that works and all that is fine, but it may not be optimal. I want it to be at least near to optimal.

Endpoint for version 0.2

https://api.fyyd.de/0.2/

always prefix this for every request or else you'll end up in version 0.1 which is mostly different from 0.2.

Table of contents

Preface

Podcasts and their metadata always has been a fascinating subject for me. So much data, so few approaches to collect and bring it to good use.

Now it's 2017 and there are a few (I know of three, maybe four) more or less experimental projects to work with this gigantic loads of data. One of that projects is fyyd, my attempt to create a

- podcast search engine
- podcast directory
- podcast recommendation system

fyyd is a freetime project, that get's some love now and then. The progress is as fast as I can afford some free time beside family, job and whatever one has to do.

As I started fyyd, I knew that an API would be crucial for this project to escape the scope of just a webpage and to find more stunning ways to do whatever one can do with that data. As always: The crowd will find a way to use the data you would never have dreamed of.

Version 0.1 of that API found it's way to a small group of developers, leading into the development of Skoon, an iOS-App to curate podcast's episodes directly from some podcatchers and the integration of fyyd into podcat, a lovely designed podcatcher that pleasantly differs from most machine like user interfaces.

Version 0.1 had its problems and was a rush job. Thanks to Jeanette and Stefan here and now you see the documentation for version 0.2, that at the moment has no additional features, but recieved a rewrite to be more logical and safe.

Thanks to all who helped, will help and contribute to this small project.

Authorization

Authorization for 3rd party apps and services is provided with an oauth2 interface. At the moment this is a little bit rudimentary, but working.

App registration

First step is to register your app at fyyd.de. At the time of writing this documentation (2017-04-17), you have to be a registered developer at fyyd. This is not as a big deal as it might sound, but I'd like to know, who is using auth with the API and what happens. Just register an account at fyyd.de and give me a short note that you'd like to use the API via mail.

When this is done, head to https://fyyd.de/dev/app/ to create the app's registration at fyyd.

You then create the credentials and maybe an accesstoken (which only works on app context, which is non existent at this time).

Auth workflow

user context

Head your client to **https://fyyd.de/oauth/authorize?client_id=XYZ** where client_id is your client's id (wow... ;)

The user logs in (if neccessary) and authorizes the app.

fyyd now redirects to your registered app's callback-URL, appending the token via #token=ACCESSTOKEN.

In all future requests, add an authorization header:

```
Authorization: Bearer ACCESSTOKEN
```

app context

The authorization for application only level is working, but there is no use-case for that at the moment.

The most simple way would to create the accesstoken directly at the developers page at fyyd. You may also send a POST request to https://fyyd.de/oauth/token, containing

```
grant_type=client_credentials
```

inside the body.

Additionally you need to either set

an Authorization Header like

```
Authorization: Basic base64encode($client_id:$client_secret)
```

(please: base64encode has to be calculated and not included in the headers...)

or added as

```
&client_id=$client_id&client_secret=$client_secret
```

in the requests body.

Account

Requests for /account always refer to the user identified by the accesstoken. Thus, you of course need to authenticate to use this requests. All other requests are free until now.

[GET /account/info]

Description

Returns general information about the account

Parameters

Authorization

required

Response

```
{
        "status": 1,
        "msg": "ok",
        "meta": {
                "API_INFO": {
                        "API_VERSION": 0.2
                }
        },
        "data": {
                "nick": "your-nick",
                "id": 42,
                "fullname": "Jon Doe",
                "bio": "This is me!",
                "url": "https:\/\/fyyd.de\/",
                "layoutImageURL": "https:\/\/img.fyyd.de\/user\/layout\/42.jpg",
                "thumbImageURL": "https:\/\/img.fyyd.de\/user\/thumbs\/42.png",
                "microImageURL": "https:\/\/img.fyyd.de\/user\/micro\/42.png"
        }
}
```

[GET /account/curations]

Description

Returns this account's curations. Includes non-public curations as well.

Parameters

Authorization

required

Response

```
{
    "status": 1,
    "msg": "ok",
    "meta": {
        "API_INFO": {
            "API_VERSION": 0.2
        }
    },
    "data": [
        {
            "title": "Retro Computing",
```

```
            "id": 1099,
            "description": "Alles \u00fcber Rechner, f\u00fcr die es garantiert
   keine Garantie mehr gibt.",
            "layoutImageURL": "https:\/\/img.fyyd.de\/curation\/layout\/1099.jpg",
            "thumbImageURL": "https:\/\/img.fyyd.de\/curation\/thumbs\/1099.png",
            "microImageURL": "https:\/\/img.fyyd.de\/curation\/micro\/1099.png",
            "public": 0,
            "slug": "retro-computing",
            "url": "https:\/\/fyyd.de\/user\/eazy\/curation\/retro-computing",
            "xmlURL": "https:\/\/feeds.fyyd.de\/eazy\/retro-computing"
        },
        {
            "title": "Netzpolitik",
            "id": 623,
            "description": "Vorsicht, k\u00f6nnte Spuren von Netzpolitik
   enthalten!",
            "layoutImageURL": "https:\/\/img.fyyd.de\/curation\/layout\/623.jpg",
            "thumbImageURL": "https:\/\/img.fyyd.de\/curation\/thumbs\/623.png",
            "microImageURL": "https:\/\/img.fyyd.de\/curation\/micro\/623.png",
            "public": 1,
            "slug": "netzpolitik",
            "url": "https:\/\/fyyd.de\/user\/eazy\/curation\/netzpolitik",
            "xmlURL": "https:\/\/feeds.fyyd.de\/eazy\/netzpolitik"
        },
        {
            "title": "Podcasting",
            "id": 586,
            "description": "Alles \u00fcber das Podcasting",
            "layoutImageURL": "https:\/\/img.fyyd.de\/curation\/layout\/586.jpg",
            "thumbImageURL": "https:\/\/img.fyyd.de\/curation\/thumbs\/586.png",
            "microImageURL": "https:\/\/img.fyyd.de\/curation\/micro\/586.png",
            "public": 1,
            "slug": "podcasting",
            "url": "https:\/\/fyyd.de\/user\/eazy\/curation\/podcasting",
            "xmlURL": "https:\/\/feeds.fyyd.de\/eazy\/podcasting"
        }

    ]
}
```

[GET /account/collections]

Description

Returns this account's collections.

Parameters

Authorization

required

Response

```json
{
    "status": 1,
    "msg": "ok",
    "meta": {
        "API_INFO": {
            "API_VERSION": 0.2
        }
    },
    "data": [
        {
            "title": "Netzpolitik",
            "id": 1,
            "description": "(Netz)politische Podcasts",
            "layoutImageURL": "https:\/\/img.fyyd.de\/collection\/layout\/1.jpg",
            "thumbImageURL": "https:\/\/img.fyyd.de\/collection\/thumbs\/1.png",
            "microImageURL": "https:\/\/img.fyyd.de\/collection\/micro\/1.png",
            "slug": "netzpolitik",
            "url": "https:\/\/fyyd.de\/user\/eazy\/collection\/netzpolitik"
        },
        {
            "title": "Gamechanger",
            "id": 8,
            "description": "",
            "layoutImageURL": "https:\/\/img.fyyd.de\/collection\/layout\/8.jpg",
            "thumbImageURL": "https:\/\/img.fyyd.de\/collection\/thumbs\/8.png",
            "microImageURL": "https:\/\/img.fyyd.de\/collection\/micro\/8.png",
            "slug": "gamechanger",
            "url": "https:\/\/fyyd.de\/user\/eazy\/collection\/gamechanger"
        },
        {
            "title": "Lieblingspodcasts",
            "id": 11,
            "description": "Was ich so mit auf die Insel nehmen w\u00fcrde..",
            "layoutImageURL": "https:\/\/img.fyyd.de\/collection\/layout\/11.jpg",
            "thumbImageURL": "https:\/\/img.fyyd.de\/collection\/thumbs\/11.png",
            "microImageURL": "https:\/\/img.fyyd.de\/collection\/micro\/11.png",
            "slug": "lieblingspodcasts",
            "url": "https:\/\/fyyd.de\/user\/eazy\/collection\/lieblingspodcasts"
        }
    ]
}
```

User

Gets public available information about a registered account at fyyd

[GET /user]

Description

Retrieves the accounts profile information

Parameters

- **user_id (required, int)** the users account id
- **or nick (optional, string)** identifies the user by nick

Response

```
{
  "status": 1,
  "msg": "ok",
  "meta": {
      "API_INFO": {
          "API_VERSION": 0.2
      }
  },
  "data": {
      "nick": "eazy",
      "id": 42,
      "fullname": "John Doe",
      "bio": "This is me!",
      "url": "https:\/\/fyyd.de\/",
      "layoutImageURL": "https:\/\/img.fyyd.de\/user\/layout\/42.jpg",
      "thumbImageURL": "https:\/\/img.fyyd.de\/user\/thumbs\/42.png",
      "microImageURL": "https:\/\/img.fyyd.de\/user\/micro\/42.png"
  }
}
```

[GET /user/curations[/episodes]]

Description

Retrieves the curations maintained by the given user. You may append /episodes to also get the content of this curations. If you're authenticated and you request your own curations, non-public curations will be included also.

Parameters

- **user_id (required, int)** the users account id
- **or nick (optional, string)** identifies the user by nick

Response

```
{
  "status": 1,
  "msg": "ok",
  "meta": {
      "API_INFO": {
          "API_VERSION": 0.2
      }
  },
  "data": [
      {
          "title": "Netzpolitik",
          "id": 623,
          "description": "Vorsicht, k\u00f6nnte Spuren von Netzpolitik enthalten!",
          "layoutImageURL": "https:\/\/img.fyyd.de\/curation\/layout\/623.jpg",
          "thumbImageURL": "https:\/\/img.fyyd.de\/curation\/thumbs\/623.png",
          "microImageURL": "https:\/\/img.fyyd.de\/curation\/micro\/623.png",
          "public": 1,
          "slug": "netzpolitik",
          "url": "https:\/\/fyyd.de\/user\/eazy\/curation\/netzpolitik",
          "xmlURL": "https:\/\/feeds.fyyd.de\/eazy\/netzpolitik"
      },
      {
          "title": "Podcasting",
          "id": 586,
          "description": "Alles \u00fcber das Podcasting",
          "layoutImageURL": "https:\/\/img.fyyd.de\/curation\/layout\/586.jpg",
          "thumbImageURL": "https:\/\/img.fyyd.de\/curation\/thumbs\/586.png",
          "microImageURL": "https:\/\/img.fyyd.de\/curation\/micro\/586.png",
          "public": 1,
          "slug": "podcasting",
          "url": "https:\/\/fyyd.de\/user\/eazy\/curation\/podcasting",
          "xmlURL": "https:\/\/feeds.fyyd.de\/eazy\/podcasting"
      }
  ]
}
```

[GET /user/collections[/podcasts]]

Description

Retrieves the collections maintained by the given user. You may append /podcasts to also get the content of this collections.

Parameters

- **user_id (required, int)** the users account id
- **or nick (optional, string)** identifies the user by nick

Response

```
{
  "status": 1,
  "msg": "ok",
  "meta": {
      "API_INFO": {
          "API_VERSION": 0.2
      }
  },
  "data": [
      {
          "title": "Netzpolitik",
          "id": 1,
          "description": "(Netz)politische Podcasts",
          "layoutImageURL": "https:\/\/img.fyyd.de\/collection\/layout\/1.jpg",
          "thumbImageURL": "https:\/\/img.fyyd.de\/collection\/thumbs\/1.png",
          "microImageURL": "https:\/\/img.fyyd.de\/collection\/micro\/1.png",
          "slug": "netzpolitik",
          "url": "https:\/\/fyyd.de\/user\/eazy\/collection\/netzpolitik"
      },
      {
          "title": "Gamechanger",
          "id": 8,
          "description": "",
          "layoutImageURL": "https:\/\/img.fyyd.de\/collection\/layout\/8.jpg",
          "thumbImageURL": "https:\/\/img.fyyd.de\/collection\/thumbs\/8.png",
          "microImageURL": "https:\/\/img.fyyd.de\/collection\/micro\/8.png",
          "slug": "gamechanger",
          "url": "https:\/\/fyyd.de\/user\/eazy\/collection\/gamechanger"
      }
  ]
}
```

Action

This is the actions API to fyyd. It's a more or less formfree key-value-store.

Requests for /action always refer to the user identified by the accesstoken and the used app.

[POST /action]

Description

Saves an action for the authorized user.

The action always referes to an object inside fyyd's database, which may be:

- an episode
- a podcast
- a curation
- a colleciton
- a user

The action itself is represented by a string you may choose freely. Some of these actions will be used in future features of fyyd. This is not implemented at the time this documentation is written.

Parameters

- **object_id (required, int)** the id of the object
- **object_type (required, string)** the type of the object. episode, podcast, curation, collection, user
- **action (required, string)** a free to choose action
- **metadata (optional, string)** further data you'd like to associate to this action

Response

Returns 204 - No Content

[GET /action]

Description

Gets all actions for the authorized user, triggered by the authorized app and filtered by the given criteria.

Parameters

- **object_id (optional, int)** the id of the object
- **object_type (optional, string)** the type of the object. episode, podcast, curation, collection, user
- **action (optional, string)** a free to choose action
- **metadata (optional, string)** further data you'd like to associate to this action
- **date_start (optional, date string)** optional start date for this query
- **date_end (optional, date string)** optional end date for this query

Choose one or more of these parameters. Providing none of them returns all actions issued by the logged in user with the authorized app.

Response

```
{
    "status": 1,
    "msg": "ok",
    "meta": {
        "API_INFO": {
            "API_VERSION": 0.2
        }
    },
    "data": [
        {
            "id": 1,
            "object_id": 85,
            "object_type": "podcast",
            "action": "test",
            "metadata": "More data",
            "date": "2017-04-17 20:04:22",
            "user_id": 23,
            "app_id": 42
        },
        {
            "id": 2,
            "object_id": 85,
            "object_type": "podcast",
            "action": "test",
            "metadata": "More data, really!",
            "date": "2017-04-17 20:09:01",
            "user_id": 23,
            "app_id": 42
        }
    ]
}
```

Podcast

Retrieves information about a podcast

[GET /podcast/[/episodes]]

Description

Gets information about the podcast with id {id}. You may retrieve the episodes by appending /episodes.

Additionally you can address the resultset with {page} and {count}.

Parameters

- **podcast_id (required, int)** the podcast's id
- **page (optional, int)** the page you want to address, default: 0
- **count (optional,int)** the page's size, default: 50

Response

```
{
  "status": 1,
  "msg": "ok",
  "meta": {
      "API_INFO": {
          "API_VERSION": 0.2
      }
  },
  "data": {
      "title": "Schall und Rauch",
      "id": 64,
      "xmlURL": "http:\/\/schallrauch.hoersuppe.de\/sur\/m4a",
      "htmlURL": "http:\/\/schallrauch.hoersuppe.de\/sur\/show",
      "imgURL": "http:\/\/schallrauch.hoersuppe.de\/feedimages\/itunes1400.png",
      "status": 399,
      "slug": "schall-und-rauch",
      "layoutImageURL": "https:\/\/img.fyyd.de\/pd\/layout\/64.jpg",
      "thumbImageURL": "https:\/\/img.fyyd.de\/pd\/thumbs\/64.png",
      "microImageURL": "https:\/\/img.fyyd.de\/pd\/micro\/64.png",
      "language": "de",
      "lastpoll": "2017-04-17 14:42:41",
      "generator": "firtz podcast publisher v2.0",
      "categories": [
            52
        ],
      "lastpub": "2016-01-08 21:15:22",
      "rank": 22,
      "url_fyyd": "https:\/\/fyyd.de\/podcast\/64",
      "description": "Alexander und Christian, zwei mittelreife Fr\u00fcchte der
70er, sozialisiert in den 80ern ver\u00f6ffentlichen ihre monatlichen
Schw\u00e4tzchen auf der Parkbank.",
      "subtitle": "Zwei Opas erz\u00e4hlen vom Krieg",
      "episodes": [
          {
              "title": "Stasi, Sherlock und die ARD",
              "id": 1742,
              "guid": "sur-sur022",
              "url": "http:\/\/schallrauch.hoersuppe.de\/sur\/show\/sur022",
              "enclosure": "http:\/\/cdn.podseed.org\/schallundrauch\/sur022.m4a",
              "podcast_id": 64,
              "imgURL":
"http:\/\/schallrauch.hoersuppe.de\/feedimages\/itunes1400.png",
              "pubdate": "2016-01-08 21:15:22",
```

```
            "duration": 5579,
            "url_fyyd": "https:\/\/fyyd.de\/episode\/1742",
            "description": "<p>Alexander erz\u00e4hlt von seinen Eindr\u00fccken
    beim Besuch der BStU, war dar\u00fcber hinaus in London und macht mir die Nase lang,
    weil er dort die aktuelle Folge Sherlock schon geschaut hat. Im Hotel. So ein Hund!
    <\/p>\n\n<p>Daf\u00fcr aber hat Deutschland durchaus etwas zu bieten im Bereich der
    Serien, man sollte nur nicht unbedingt beim Tatort suchen. Alexander rezitiert
    Schweiger, was dann schnell das Ende der Folge einl\u00e4utet…<\/p>\n"
          }
      ]
    }
}
```

[POST /podcast/action]

Description

Performes an action to a podcast owned by you (see claim-button on every podcast page).

At the moment, 'check' is the only action to perform. This triggers an update within the next poll cycle.

Parameters

- **podcast_id (required, int)** the podcast's id
- **action (required, string)** the action to perform:
  - **check**

Response

returns a 204: No Content

[GET /podcasts]

Description

Gets information about all podcasts within fyyd.

Additionally you can address the resultset with {page} and {count}.

Please note that this request is the plural form of podcast, podcasts ;-)

Parameters

- **page (optional, int)** the page you want to address, default: 0
- **count (optional,int)** the page's size, default: 50

Response

The response includes data about the pagination.

```json
{
  "status": 1,
  "msg": "ok",
  "meta": {
      "paging": {
          "count": 2,
          "page": 0,
          "first_page": 0,
          "last_page": 7094,
          "next_page": 1,
          "prev_page": null
      },
      "API_INFO": {
          "API_VERSION": 0.2
      }
  },
  "data": [
      {
          "title": "Freakonomics Radio",
          "id": 1,
          "xmlURL": "http:\/\/feeds.feedburner.com\/freakonomicsradio",
          "htmlURL": "http:\/\/www.wnyc.org\/articles\/freakonomics-podcast",
          "imgURL":
"https:\/\/media2.wnyc.org\/i\/1400\/1400\/l\/80\/1\/wn16_wnycstudios_freakonomics-
rev3.png",
          "status": 390,
          "slug": "freakonomics-radio",
          "layoutImageURL": "https:\/\/img.fyyd.de\/pd\/layout\/1.jpg",
          "thumbImageURL": "https:\/\/img.fyyd.de\/pd\/thumbs\/1.png",
          "microImageURL": "https:\/\/img.fyyd.de\/pd\/micro\/1.png",
          "language": "en",
          "lastpoll": "2017-04-16 00:34:49",
          "generator": null,
          "categories": [
            52
          ],
          "lastpub": "2017-03-16 04:00:00",
          "rank": 1,
          "url_fyyd": "https:\/\/fyyd.de\/podcast\/1",
          "description": "Have fun discovering the hidden side of everything with
host Stephen J. Dubner, co-author of the best-selling \"Freakonomics\u201d books.
Each week, hear surprising conversations that explore the riddles of everyday life
and the weird wrinkles of human nature\u2014from cheating and crime to parenting and
sports. Dubner talks with Nobel laureates and provocateurs, social scientists and
entrepreneurs \u2014 and his \u201cFreakonomics\u201d co-author Steve Levitt. After
just a few episodes, this podcast will have you too thinking like a Freak. Produced
by WNYC Studios, home of other great podcasts such as \u201cRadiolab,\" \"Death, Sex
& Money,\" and \"On the Media.\" ",
          "subtitle": "Have fun discovering the hidden side of everything with host
```

```
      Stephen J. Dubner, co-author of the best-selling \"Freakonomics\u201d books. Each
 week, hear surprising conversations that explore the riddles of everyday life and
 the weird wrinkles of human nature\u2014f"
        },
        {
            "title": "Skeptoid",
            "id": 3,
            "xmlURL": "http:\/\/skeptoid.com\/podcast.xml",
            "htmlURL": "https:\/\/skeptoid.com",
            "imgURL": "http:\/\/skeptoid.com\/images\/1500b.jpg",
            "status": 304,
            "slug": "skeptoid",
            "layoutImageURL": "https:\/\/img.fyyd.de\/pd\/layout\/3.jpg",
            "thumbImageURL": "https:\/\/img.fyyd.de\/pd\/thumbs\/3.png",
            "microImageURL": "https:\/\/img.fyyd.de\/pd\/micro\/3.png",
            "language": "en",
            "lastpoll": "2017-04-17 02:39:44",
            "generator": null,
            "categories": [
                    52
             ],
            "lastpub": "2017-04-11 02:00:00",
            "rank": 1,
            "url_fyyd": "https:\/\/fyyd.de\/podcast\/3",
            "description": "Since 2006, the weekly Skeptoid podcast has been taking on
 all the most popular myths and revealing the true science, true history, and true
 lessons we can learn from each. Free subscribers get the most recent 50 episodes,
 premium subscribers (skeptoid.com) can access the full archive, all ad-free.",
            "subtitle": "A weekly look at the true facts behind urban legends and
 stories you've probably heard."
        }
  ]
}
```

Additionally links inside the header are provided:

```
Link: <https://api.fyyd.de/0.2/podcasts/1/2>; rel="next",
<https://api.fyyd.de/0.2/podcasts/0/2>; rel="first",
<https://api.fyyd.de/0.2/podcasts/7094/2>; rel="last"
```

[GET /categories]

Description

Gets the complete categories tree

Parameters

Response

```json
{
  "status": 1,
  "msg": "ok",
  "meta": {
      "API_INFO": {
          "API_VERSION": 0.2
      }
  },
  "data": [
      {
          "id": 1,
          "slug": "arts",
          "name": "Arts",
          "name_de": "Kunst",
          "subcategories": [
              {
                  "id": 2,
                  "slug": "design",
                  "name": "Design",
                  "name_de": "Design"
              },
              {
                  "id": 3,
                  "slug": "fashion-beauty",
                  "name": "Fashion & Beauty",
                  "name_de": "Mode & Sch\u00f6nheit"
              },
              {
                  "id": 4,
                  "slug": "food",
                  "name": "Food",
                  "name_de": "Essen"
              },
              {
                  "id": 5,
                  "slug": "literature",
                  "name": "Literature",
                  "name_de": "Literatur"
              },
              {
                  "id": 6,
                  "slug": "performing-arts",
                  "name": "Performing Arts",
                  "name_de": "Darstellende K\u00fcnste"
              },
              {
                  "id": 7,
```

```machine_data
                "slug": "visual-arts",
                "name": "Visual Arts",
                "name_de": "Bildende K\u00fcnste"
            }
        ]
    },
    {
        "id": 8,
        "slug": "business",
        "name": "Business",
        "name_de": "Wirtschaft",
        "subcategories": [
            {
                "id": 9,
                "slug": "business-news",
                "name": "Business News",
                "name_de": "Wirtschaftsnachrichten"
            },
            {
                "id": 10,
                "slug": "careers",
                "name": "Careers",
                "name_de": "Karriere"
            },
            {
                "id": 11,
                "slug": "investing",
                "name": "Investing",
                "name_de": "Geldanlage"
            },
            {
                "id": 12,
                "slug": "management-marketing",
                "name": "Management & Marketing",
                "name_de": "Management & Markteting"
            },
            {
                "id": 13,
                "slug": "shopping",
                "name": "Shopping",
                "name_de": "Shopping"
            }
        ]
    },
    {
        "id": 14,
        "slug": "comedy",
        "name": "Comedy",
        "name_de": "Kom\u00f6dien",
```

```
            "subcategories": []
        }
      ]
    ]
  }
```

[GET /category]

Description

Retrieves the podcasts inside the specified category. The categories system referres to Apple's iTunes Categories. Categories not in Apple's specification are ignored on import.

Parameters

- **category_id (required, int)** the category's id (see /categories)
- **page (optional, int)** the page you want to address, default: 0
- **count (optional,int)** the page's size, default: 50

Response

```
{
  "status": 1,
  "msg": "ok",
  "meta": {
      "paging": {
          "count": 2,
          "page": 0,
          "first_page": 0,
          "last_page": 117,
          "next_page": 1,
          "prev_page": null
      },
      "API_INFO": {
          "API_VERSION": 0.2
      }
  },
  "data": {
      "category": {
          "id": 61,
          "slug": "professional",
          "title": "Professional",
          "url": "https:\/\/fyyd.de\/discover\/cat\/professional\/0"
      },
      "parent": {
          "id": 57,
          "slug": "sports-recreation",
          "title": "Sports & Recreation",
          "url": "https:\/\/fyyd.de\/discover\/cat\/sports-recreation\/0"
```

```
        },
      "podcasts": [
          {
              "title": "vollraute - der gladbach podcast mit der raute im herzen -",
              "id": 170,
              "xmlURL": "http:\/\/feedpress.me\/vollrautepodcast",
              "htmlURL": "http:\/\/vollraute.de",
              "imgURL": "http:\/\/vollraute.de\/wp-
content\/cache\/podlove\/5a\/cc055669c606ec882f928be09be530\/vollraute-der-gladbach-
podcast-mit-der-raute-im-herzen_original.png",
              "status": 390,
              "slug": "vollraute-der-gladbach-podcast-mit-der-raute-im-herzen",
              "layoutImageURL": "https:\/\/img.fyyd.de\/pd\/layout\/170.jpg",
              "thumbImageURL": "https:\/\/img.fyyd.de\/pd\/thumbs\/170.png",
              "microImageURL": "https:\/\/img.fyyd.de\/pd\/micro\/170.png",
              "language": "de",
              "lastpoll": "2017-04-16 00:36:11",
              "generator": "Podlove Podcast Publisher v2.4.1",
              "categories": [
                  52
              ],
              "lastpub": "2017-04-11 21:14:53",
              "rank": 1,
              "url_fyyd": "https:\/\/fyyd.de\/podcast\/170",
              "description": "Ein Podcast \u00fcber die Borussia aus
M\u00f6nchengladbach.\nNie neutral, sondern immer durch die Vereinsbrille.",
              "subtitle": "Ein Podcast \u00fcber die Borussia aus
M\u00f6nchengladbach"
          },
          {
              "title": "Collinas Erben",
              "id": 296,
              "xmlURL": "http:\/\/fokus-fussball.de\/feed\/mp3\/",
              "htmlURL": "http:\/\/fokus-fussball.de",
              "imgURL": "http:\/\/fokus-fussball.de\/wp-
content\/cache\/podlove\/5f\/716d5e4a7909b9b87b75a456524d89\/collinas-
erben_original.jpg",
              "status": 304,
              "slug": "collinas-erben",
              "layoutImageURL": "https:\/\/img.fyyd.de\/pd\/layout\/296.jpg",
              "thumbImageURL": "https:\/\/img.fyyd.de\/pd\/thumbs\/296.png",
              "microImageURL": "https:\/\/img.fyyd.de\/pd\/micro\/296.png",
              "language": "de",
              "lastpoll": "2017-04-17 16:12:25",
              "generator": "Podlove Podcast Publisher v2.3.8",
              "categories": [
                  52
              ],
              "lastpub": "2016-11-15 08:53:35",
```

```
            "rank": 10,
            "url_fyyd": "https:\/\/fyyd.de\/podcast\/296",
            "description": "Fu\u00dfball ist ein einfaches Spiel \u2013 deshalb
ist es so erfolgreich: Der Ball ist rund. Das Spiel dauert um die 90
Minuten.\n\nProblematisch wird es nur, wenn Sonderf\u00e4lle auftreten. Wenn ein
Spieler pl\u00f6tzlich mit nacktem Oberk\u00f6rper auf den Zaun springt, wenn ein
Engl\u00e4nder doch mal mit einem Wembley-Tor trifft oder wenn auch nach der zigsten
Wiederholung nicht klar ist, ob es ein Foul in der 88. Minute war und ob es im oder
vor dem Strafraum passierte.\n\nDer Schiedsrichter-Podcast Collinas Erben
m\u00f6chte f\u00fcr mehr Fachwissen auf den Trib\u00fcnen und an den
Empfangsger\u00e4ten sorgen. Alex Feuerherdt und Klaas Reese sprechen \u00fcber das
Schiedsrichterwesen im Allgemeinen und die Spielregeln im Speziellen.",
            "subtitle": "Der Schiedsrichter-Podcast"
        }
    ]
  }
}
```

[GET /podcast/recommend]

Description

Gets information about what fyyd thinks, might be a good idea to listen to also. At the moment, this is based on collections information.

Parameters

- **podcast_id (required, int)** the Podcast's id
- **count (optional,int)** the number of recommended podcasts, default: 10

Response

```
{
    "status": 1,
    "msg": "ok",
    "meta": {
        "API_INFO": {
            "API_VERSION": 0.2
        }
    },
    "data": [
        {
            "title": "Explikator",
            "id": 158,
            "xmlURL": "http:\/\/explikator.de\/feed\/podcast\/",
            "htmlURL": "http:\/\/explikator.de\/podcast",
            "imgURL": "http:\/\/explikator.de\/wp-
content\/cache\/podlove\/ed\/92555a31e2f10f4922dd5b5b8b2b77\/explikator_original.png
",
            "status": 304,
            "slug": "explikator",
            "layoutImageURL": "https:\/\/img.fyyd.de\/pd\/layout\/158.jpg",
            "thumbImageURL": "https:\/\/img.fyyd.de\/pd\/thumbs\/158.png",
            "microImageURL": "https:\/\/img.fyyd.de\/pd\/micro\/158.png",
            "language": "de",
            "lastpoll": "2017-04-23 10:13:56",
            "generator": "Podlove Podcast Publisher v2.3.18",
            "user_id": null,
            "categories": [
                52
             ],
            "lastpub": "2017-04-20 00:01:21",
            "rank": 29,
            "url_fyyd": "https:\/\/fyyd.de\/podcast\/158",
            "description": "Jeden Werktag 10 Minuten Podcast und ein neuer
Musiktitel. Themen sind Wissenschaft, Bewusstsein, Geschichte, Film & TV und
Entertainment. Unbekanntes & Unbeachtetes, Hintergrund & Oberfl\u00e4che. Mal
ernster, mal satirischer aber immer Morgen-Kaffee-tauglich!",
            "subtitle": "Morgenradio 2.0: Jeden Werktag 10 Minuten Unbeachtetes &
Unbekanntes aus Gesellschaft und Kultur."
        }
    ]
}
```

Episode

Request to retrieve an episode's information

[GET /episode]

Description

Returns information about a single episode

Parameters

- **episode_id (required, int)** the episode's id

Response

```
{
  "status": 1,
  "msg": "ok",
  "meta": {
      "API_INFO": {
          "API_VERSION": 0.2
      }
  },
   "data": {
        "title": "Space Hulk (Stay Forever, Folge 33)",
        "id": 42,
        "guid": "796a53669cb29b0504fdd4f0a3f9a41d",
        "url":
"http:\/\/kaliban.podspot.de\/files\/Stay_Forever_Ep33_Space_Hulk.mp3",
        "enclosure":
"http:\/\/kaliban.podspot.de\/files\/Stay_Forever_Ep33_Space_Hulk.mp3",
        "podcast_id": 5,
        "imgURL": "",
        "pubdate": "2014-03-06 18:10:30",
        "duration": null,
        "url_fyyd": "https:\/\/fyyd.de\/episode\/42",
        "description": "Christian hasst Space Hulk, Gunnar liebt es. Fight!"
    }
}
```

Curations

All about Curations inside fyyd. Retrieve and set.

[GET /curation[/episodes]]

Description

Returns information about a single curation, given the curation's id. May append episodes, not paged at the moment. If you're authenticated and you request your own curation, non-public curations will be accessable too.

Parameters

- **curation_id (required, int)** the curation's id

Response

```
{
  "status": 1,
  "msg": "ok",
  "meta": {
      "API_INFO": {
          "API_VERSION": 0.2
      }
  },
  "data": {
      "title": "Testfeed",
      "id": 601,
      "description": "nur ein Testfeed",
      "layoutImageURL": "https:\/\/fyyd.de\/images\/curation300.png",
      "thumbImageURL": "https:\/\/fyyd.de\/images\/curation80.png",
      "microImageURL": "https:\/\/fyyd.de\/images\/curation20.png",
      "public": 1,
      "slug": "testfeed",
      "user_id": 1000,
      "url": "https:\/\/fyyd.de\/user\/eazy\/curation\/testfeed",
      "xmlURL": "https:\/\/feeds.fyyd.de\/eazy\/testfeed",
      "episodes": [
          {
              "title": "A!195 \u2013 Raussismus",
              "id": 1662345,
              "guid": "podlove-2017-04-14t12:27:42+00:00-56a19c3a4209a8d",
              "url": "https:\/\/aufwachen-podcast.de\/2017\/04\/14\/a195-
raussismus\/",
              "enclosure": "https:\/\/aufwachen-
podcast.de\/podlove\/file\/3244\/s\/feed\/c\/itunes\/195raussismus.m4a",
              "podcast_id": 95,
              "imgURL": "",
              "pubdate": "2017-04-14 15:05:11",
              "duration": 16018,
              "favedDate": "2017-04-17 16:46:39",
              "url_fyyd": "https:\/\/fyyd.de\/episode\/1662345",
              "description": "Freitag, 14. April 2017, 15:05 Uhr<p>Frankreich steht
seit Monaten Kopf."
          },
          {
```

```
            "title": "LdN051 BVB-Bus, Steuern, Syrien, US-Au\u00dfen-
\u201cPolitik\u201d, Altmaier",
            "id": 1661432,
            "guid": "podlove-2017-04-14t07:57:05+00:00-5ee52bc0e03190f",
            "url":
"https:\/\/www.kuechenstud.io\/lagedernation\/2017\/04\/14\/ldn051-bvb-bus-steuern-
syrien-us-aussen-politik-altmaier\/",
            "enclosure":
"https:\/\/www.kuechenstud.io\/lagedernation\/podlove\/file\/1795\/s\/feed\/c\/mp3\/
LdN051.mp3",
            "podcast_id": 44980,
            "imgURL": "",
            "pubdate": "2017-04-14 10:07:59",
            "duration": 3945,
            "favedDate": "2017-04-17 16:45:58",
            "url_fyyd": "https:\/\/fyyd.de\/episode\/1661432",
            "description": "<p>Liebe Lage-Gemeinde!<\/p>\n<p>P\u00fcnktlich zu
Ostertagen die neue Lage."
        }
    ]
  }
}
```

[POST /curation]

Description

Creates or modifies a curation.

Parameters

- **curation_id (opt, int)** when set, an existing curation is modified
- **title (req when created, string)** the title to set for this curation
- **description (opt, string)** the description for this curation
- **slug (opt, string)** the slug to set for this curation. If not set while creation, the title will be slugified. *When set, but empty while modifying a curation, also a slugified title will overwrite the existing slug*.
- **public (opt, def: 0)** defines, if this curation will be set to public and thus accessable for all.
- **image (opt, binary)** upload an optional image for the curation. Use multipart/form-data ([RFC 2388](#)) and provide a content-type.

Response

Creating and mofiying a curation returns the complete object of the curation as in GET /curation.

[POST /curation/delete]

Description

Deletes a curation. Please note: there's no coming back, no questions, no backup. It's deleted!

Parameters

- **curation_id (req, int)** when set, an existing curation is modified

Response

returns 204: No content

[POST /curate]

Description

Adds OR removes the episode identified by episode_id into the curation identified by curation_id. The curation must be owned by the user identified by the accesstoken. The state of the episode inside this curation toggles with each call, the state is returned inside response's data section.

If you want to enforce a state without knowing what state the episode had before, use **force_state**.

Parameters

- **curation_id (required, int)** the curation's id
- **episode_id (required, int)** the episode's id
- **why (optional, string)** a text to indicate, why you curate this episode
- **force_state (optional, bool)** enforce adding or removing the episode from the curation

Response

```
{
    "status": 1,
    "msg": "ok",
    "meta": {
        "API_INFO": {
            "API_VERSION": 0.2
        }
    },
    "data": {
        "state": 1,
        "text": "added AdH039: Singende Nonne to TestKuration"
    }
}
```

[GET /curate]

Description

Retuns the state of the curation of an episode in a curation.

Parameters

- **curation_id (required, int)** the curation's id
- **episode_id (required, int)** the episode's id

Response

```
{
    "status": 1,
    "msg": "ok",
    "meta": {
        "API_INFO": {
            "API_VERSION": 0.2
        }
    },
    "data": {
        "state": 1
    }
}
```

Collections

This part is all about collections at fyyd. The content part of collections are podcasts.

[GET /collection[/podcasts]]

Description

Returns information about a single collection, given the collection's id. May append podcasts, not paged at the moment.

Parameters

- **collection_id (required, int)** the collection's id

Response

```
{
  "status": 1,
  "msg": "ok",
  "meta": {
      "API_INFO": {
          "API_VERSION": 0.2
      }
  },
  "data": {
      "title": "Netzpolitik",
      "id": 1,
      "description": "(Netz)politische Podcasts",
```

          "layoutImageURL": "https:\/\/img.fyyd.de\/collection\/layout\/1.jpg",
          "thumbImageURL": "https:\/\/img.fyyd.de\/collection\/thumbs\/1.png",
          "microImageURL": "https:\/\/img.fyyd.de\/collection\/micro\/1.png",
          "slug": "netzpolitik",
          "user_id": 1000,
          "url": "https:\/\/fyyd.de\/user\/eazy\/collection\/netzpolitik",
          "podcasts": [
              {
                  "title": "#heiseshow (Audio)",
                  "id": 5037,
                  "xmlURL": "https:\/\/www.heise.de\/heiseshow.rss",
                  "htmlURL": "http:\/\/www.heise.de\/heiseshow.rss",
                  "imgURL":
"https:\/\/www.heise.de\/imgs\/02\/5\/1\/1\/7\/THUMB_heiseshow_quadrat-
b3e99ad376d2732e.jpg",
                  "status": 304,
                  "slug": "heiseshow-audio",
                  "layoutImageURL": "https:\/\/img.fyyd.de\/pd\/layout\/5037.jpg",
                  "thumbImageURL": "https:\/\/img.fyyd.de\/pd\/thumbs\/5037.png",
                  "microImageURL": "https:\/\/img.fyyd.de\/pd\/micro\/5037.png",
                  "language": "de",
                  "lastpoll": "2017-04-17 05:20:40",
                  "generator": null,
                  "categories": [
                      52
                  ],
                  "lastpub": "2017-04-13 02:00:00",
                  "rank": 1,
                  "url_fyyd": "https:\/\/fyyd.de\/podcast\/5037",
                  "description": "Immer donnerstags um 16 Uhr diskutieren wir in der
#heiseshow mit G\u00e4sten \u00fcber aktuelle Ereignisse aus der Hightech-Welt und
der Netzpolitik",
                  "subtitle": "Jede Woche Donnerstag live (Audio)"
              },
              {
                  "title": "Breitband - Medien und digitale Kultur (ganze Sendung) -
Deutschlandradio Kultur",
                  "id": 503,
                  "xmlURL": "http:\/\/www.deutschlandradiokultur.de\/podcast-breitband-
medien-und-digitale-kultur-ganze-sendung.1552.de.podcast.xml",
                  "htmlURL": "http:\/\/www.deutschlandradio.de\/weiterleitung-breitband-
de.233.de.html",
                  "imgURL":
"http:\/\/www.deutschlandradiokultur.de\/media\/files\/2\/258cfe6db750912b0bb36410d2
fdf775v2.jpg",
                  "status": 390,
                  "slug": "breitband-medien-und-digitale-kultur-ganze-sendung-
deutschlandradio-kultur",
                  "layoutImageURL": "https:\/\/img.fyyd.de\/pd\/layout\/503.jpg",

```
                "thumbImageURL": "https:\/\/img.fyyd.de\/pd\/thumbs\/503.png",
                "microImageURL": "https:\/\/img.fyyd.de\/pd\/micro\/503.png",
                "language": "de",
                "lastpoll": "2017-04-17 17:12:06",
                "generator": null,
                "categories": [
                    52
                ],
                "lastpub": "2017-04-15 13:05:01",
                "rank": 14,
                "url_fyyd": "https:\/\/fyyd.de\/podcast\/503",
                "description": "Feed provided by Deutschlandradio. Click to visit.",
                "subtitle": "Die Beitr\u00e4ge zur Sendung"
            }
        ]
      }
}
```

[POST /collection]

Description

Creates or modifies a collection.

Parameters

- **collection_id (opt, int)** when set, an existing collection is modified
- **title (req when created, string)** the title to set for this collection
- **description (opt, string)** the description for this collection
- **slug (opt, string)** the slug to set for this collection. If not set while creation, the title will be slugified. *When set, but empty while modifying a collection, also a slugified title will overwrite the existing slug.*
- **image (opt, binary)** upload an optional image for the collection. Use multipart/form-data ([RFC 2388](#)) and provide a content-type.

Response

Creating and mofiying a collection returns the complete object of the collection as in GET /collection.

[POST /collection/delete]

Description

Deletes a collection. Please note: there's no coming back, no questions, no backup. It's deleted!

Parameters

- **collection_id (req, int)**

Response

returns 204: No content

[POST /collect]

Description

Adds OR removes the podcast identified by podcast_id into the collection identified by collection_id. The collection must be owned by the user identified by the accesstoken. The status of the podcast inside this collection toggles.

If you want to enforce a state without knowing what state the podcast had before, use **force_state**.

Parameters

- **collection_id (required, int)** the collection's id
- **podcast_id (required, int)** the podcasts's id
- **force_state (optional, bool)** enforce adding or removing the podcast from the collection.

Response

```
{
    "status": 1,
    "msg": "ok",
    "meta": {
        "API_INFO": {
            "API_VERSION": 0.2
        }
    },
    "data": {
        "state": 1,
        "text": "added Auf dem Holzweg to Gamechanger"
    }
}
```

[GET /collect]

Description

Retuns the state of the collection of a podcast in a collection.

Parameters

- **collection_id (required, int)** the collection's id
- **podcast_id (required, int)** the podcast's id

Response

```
{
    "status": 1,
    "msg": "ok",
    "meta": {
        "API_INFO": {
            "API_VERSION": 0.2
        }
    },
    "data": {
        "state": 1
    }
}
```

Search

This request is the only one in this section for now. The search API will grow in the near future, drafts are already written on how to manage this important part of fyyd's API.

[GET /search/episode]

Description

This request tries to find an episode inside fyyd's database, matching any or some of a set of given criteria.

This reflects Skoon's need to find an episode's id to add to the user's curation.

Parameters

- **title (optional, string)** the episode's title. Search might use parts of the string to find the episode.
- **guid (optional, string)** the episode's GUID as stated inside the podcasts feed.
- **podcast_id (optional, int)** the podcast's id in fyyd's database.
- **podcast_title (optional, string)** the podcast's title. Search might use parts of the string to find the podcast.
- **pubdate (optional, string)** the pubDate as stated inside the podcasts feed.
- **duration (optional, int)** the duration of the episode in seconds.
- **url (optional, string)** the episode's url as stated inside the podcast's feed.
- **term (optional,string)** a search term to find inside the episodes.

Please note: title, guid, pubdate, duration, url and term add episodes together. Think of a logical OR.

In contrast to that, podcast_id and podcast_title restrict all episodes to podcasts matching to one of podcast_id or podcast_title.

- **count (optional, int, default: 10)**

Response

```
{
  "status": 1,
  "msg": "ok",
  "meta": {
      "API_INFO": {
          "API_VERSION": 0.2
      }
  },
  "data": [
      {
          "title": "FS195 Dein Butler ist ein Zombie",
          "id": 1597636,
          "guid": "podlove-2017-03-22t23:23:57+00:00-f1bfca935c876bc",
          "url": "http:\/\/freakshow.fm\/fs195-dein-butler-ist-ein-zombie",
          "enclosure":
"https:\/\/tracking.feedpress.it\/link\/13453\/5554460\/fs195-dein-butler-ist-ein-
zombie.m4a",
          "podcast_id": 85,
          "imgURL": "",
          "pubdate": "2017-03-23 02:16:34",
          "duration": 14632,
          "url_fyyd": "https:\/\/fyyd.de\/episode\/1597636",
          "description": "\n<p>Open Source Development \u2014 Ultraschall \u2014
Podlove Publisher \u2014 Podcast Empanzipation \u2014 Thunderbolt..."
      }
  ]
}
```

[GET /search/podcast]

Description

This request tries to find a podcast inside fyyd's database, matching any or some of a set of given criteria.

Parameters

- **title (optional, string)** the podcasts's title. Search might use parts of the string to find the podcast.
- **url (optional, string)** the podcast's url as stated inside the podcast's feed.
- **term (optional,string)** a search term to find inside the podcast.

Please note: title, url and term add episodes together. Think of a logical OR.

- **count (optional, int, default: 10)**

Response

```
{
  "status": 1,
  "msg": "ok",
  "meta": {
      "API_INFO": {
          "API_VERSION": 0.2
      }
  },
  "data": [
      {
          "title": "Freak Show",
          "id": 85,
          "xmlURL": "https:\/\/feeds.metaebene.me\/freakshow\/m4a",
          "htmlURL": "http:\/\/freakshow.fm",
          "imgURL": "http:\/\/freakshow.fm\/wp-
content\/cache\/podlove\/04\/662a9d4edcf77ea2abe3c74681f509\/freak-
show_original.jpg",
          "status": 390,
          "slug": "freak-show",
          "layoutImageURL": "https:\/\/img.fyyd.de\/pd\/layout\/85.jpg",
          "thumbImageURL": "https:\/\/img.fyyd.de\/pd\/thumbs\/85.png",
          "microImageURL": "https:\/\/img.fyyd.de\/pd\/micro\/85.png",
          "language": "de",
          "lastpoll": "2017-04-16 00:36:29",
          "generator": "Podlove Podcast Publisher v2.5.0.build315",
          "categories": [
                  52,
                  61
              ],
          "lastpub": "2017-04-13 02:03:27",
          "rank": 32,
          "url_fyyd": "https:\/\/fyyd.de\/podcast\/85",
          "description": "Die muntere Talk Show um Leben mit Technik, das Netz und
Technikkultur. Bisweilen Apple-lastig aber selten einseitig. Wir leben und lieben
Technologie und reden dar\u00fcber. Mit Tim, hukl, roddi, Clemens und Denis. Freak
Show hie\u00df irgendwann mal mobileMacs.",
          "subtitle": "Menschen! Technik! Sensationen!"
      },
      {
          "title": "Timmy Trumpet \u2013 Freak Show",
          "id": 36645,
          "xmlURL": "http:\/\/timmytrumpet.podtree.com\/feed\/podcast",
          "htmlURL": "http:\/\/timmytrumpet.podtree.com",
          "imgURL": "http:\/\/media2-
timmytrumpet.podtree.com\/media\/itunes_image\/FreakShow_Radio_1400x1400.jpg",
          "status": 304,
          "slug": "timmy-trumpet-freak-show",
          "layoutImageURL": "https:\/\/img.fyyd.de\/pd\/layout\/36645.jpg",
          "thumbImageURL": "https:\/\/img.fyyd.de\/pd\/thumbs\/36645.png",
```

```
        "microImageURL": "https:\/\/img.fyyd.de\/pd\/micro\/36645.png",
        "language": "en",
        "lastpoll": "2017-04-17 03:58:39",
        "generator": "https:\/\/wordpress.org\/?v=4.5.2",
        "categories": 38,
        "lastpub": "2016-06-27 00:30:17",
        "rank": 0,
        "url_fyyd": "https:\/\/fyyd.de\/podcast\/36645",
        "description": "Australia\u2019s #1 DJ \/ Multi-Platinum selling artist
Timmy Trumpet presents a weekly mix of the hottest tracks, bootlegs, mashups and
more! Put your headphones on and turn the world off. Subscribe
now.\ntimmytrumpet.com\nfb.com\/timmytrumpet\nTwitter.com\/timmytrumpet",
        "subtitle": "Timmy Trumpet - Freak Show\u2028"
    }
  ]
}
```