

# Package ‘nettskjemar’

March 18, 2025

**Type** Package

**Title** Connect to the 'nettskjema.no' API of the University of Oslo

**Version** 1.0.1

**Description** Enables users to retrieve data, meta-data, and codebooks from [<https://nettskjema.no/>](https://nettskjema.no/). The data from the API is richer than from the online data portal. Mowinckel (2021)  [<doi:10.5281/zenodo.4745481>](https://doi.org/10.5281/zenodo.4745481).

**License** MIT + file LICENSE

**URL** <https://github.com/CAPRO-UiO/nettskjemar>,  
<https://zenodo.org/badge/latestdoi/206264675>,  
<https://github.com/capro-uo/nettskjemar>

**BugReports** <https://github.com/capro-uo/nettskjemar/issues>

**Depends** R (>= 4.1.0)

**Imports** cli, httr2, jsonlite, tools

**Suggests** covr, fs, haven, labelled, knitr, rmarkdown, spelling,  
testthat (>= 2.1.0), vcr (>= 0.6.0)

**VignetteBuilder** knitr

**Encoding** UTF-8

**Language** en-US

**RoxygenNote** 7.3.2

**NeedsCompilation** no

**Author** Athanasia Mo Mowinckel [aut, cre]  
( [<https://orcid.org/0000-0002-5756-0223>](https://orcid.org/0000-0002-5756-0223)),  
Trym Nohr Fjørtoft [ctb] ( [<https://orcid.org/0000-0002-7654-4587>](https://orcid.org/0000-0002-7654-4587))

**Maintainer** Athanasia Mo Mowinckel  [<a.m.mowinckel@psykologi.uio.no>](mailto:a.m.mowinckel@psykologi.uio.no)

**Repository** CRAN

**Date/Publication** 2025-03-18 15:10:02 UTC

## Contents

codebook . . . . .	2
get_raw_codebook . . . . .	3
has_codebook . . . . .	4
ns_add_labels . . . . .	4
ns_alter_checkbox . . . . .	5
ns_api_docs . . . . .	6
ns_auth_token . . . . .	6
ns_create_client . . . . .	7
ns_get_attachment . . . . .	7
ns_get_codebook . . . . .	8
ns_get_data . . . . .	9
ns_get_forms . . . . .	10
ns_get_form_attachments . . . . .	10
ns_get_form_reports . . . . .	11
ns_get_me . . . . .	12
ns_get_meta . . . . .	13
ns_get_submission . . . . .	13
ns_get_submission_attachments . . . . .	14
ns_list_form_attachments . . . . .	15
ns_list_submission_attachments . . . . .	16
ns_req . . . . .	16
ns_write_codebook . . . . .	17
ns_write_meta . . . . .	18
<b>Index</b>	<b>19</b>

---

codebook	<i>Constructor for nettskjema codebook</i>
----------	--

---

### Description

Function works on meta-data downloaded from a form. It will return a data.frame in long format with meta-data information from a specific nettskjema form.

### Usage

```
codebook(meta)
```

### Arguments

meta            form meta-data from [ns\\_get\\_meta](#)

### Value

long format tibble with meta-data for a specific nettskjema

## Examples

```
## Not run:  
meta_110000 <- ns_get_meta(110000)  
codebook(meta_110000)  
  
## End(Not run)
```

---

get_raw_codebook	<i>Get the raw codebook</i>
------------------	-----------------------------

---

## Description

The raw codebook is a list of answers and questions with unique ids for both, and the text associated with them. This information can be retrieved from the raw nettskjema data. This can be useful if the nettskjema submission data has been returned with `asis = TRUE`.

## Usage

```
get_raw_codebook(form_id)
```

## Arguments

`form_id` integer. Number of the form to retrieve

## Value

long format tibble with meta-data for a specific nettskjema

## Examples

```
## Not run:  
form_id <- 100000  
get_raw_codebook(form_id)  
  
Or if the token is saved in a non-standard name  
get_raw_codebook(form_id)  
  
## End(Not run)
```

---

has_codebook	<i>Check if form has codebook</i>
--------------	-----------------------------------

---

**Description**

Codebook is by default turned off in Nettskjema, but best practices in handling data is to have it on. This function checks if the codebook of a form has been activated or not

**Usage**

```
has_codebook(form_id)
```

**Arguments**

form\_id            integer. Number of the form to retrieve

**Value**

logical is codebook is turned on

**Examples**

```
## Not run:  
has_codebook(110000)  
  
## End(Not run)
```

---

ns_add_labels	<i>Add Labels to Dataset</i>
---------------	------------------------------

---

**Description**

The 'ns\_add\_labels' function applies labels to variables in a dataset based on a provided codebook. It assigns variable labels, value labels, and metadata required for the "ns-data" class.

**Usage**

```
ns_add_labels(data, codebook)
```

**Arguments**

data                data from ns\_get\_data  
codebook            object from ns\_get\_codebook

**Value**

A data frame with the same structure as 'data', but with variable and value labels applied. The resulting object is assigned the class "ns-data".

**Examples**

```
## Not run:
data <- ns_get_data(11000)
cb <- ns_get_codebook(11000)
labeled_data <- ns_add_labels(data, cb)

## End(Not run)
```

---

ns\_alter\_checkbox      *Alter checkbox matrix variables in a dataset*

---

**Description**

This function processes checkbox matrix variables in a dataset, transforming their representations into one of the specified formats.

**Usage**

```
ns_alter_checkbox(data, to = c("character", "list"), sep = ",", cb = NULL)
```

**Arguments**

data	A 'data.frame' containing the dataset to modify.
to	A character string specifying the output format for checkbox matrix variables. Must be one of "character" or "list".
sep	A character string used as a separator when 'to = "character"'. Defaults to ",".
cb	An optional codebook ('data.frame') to identify checkbox matrix variables. If 'NULL', the codebook is generated using 'ns_get_codebook()'.

**Value**

A modified 'data.frame' with processed checkbox matrix variables.

**Examples**

```
## Not run:
dt <- ns_get_data(11000)
ns_alter_checkbox(data = dt, to = "list")
ns_alter_checkbox(data = dt, to = "character", sep = ";")

## End(Not run)
```

---

ns_api_docs	<i>Access Nettskjema API Documentation</i>
-------------	--

---

**Description**

Opens the browser to display the API documentation for the Nettskjema PI.

**Usage**

```
ns_api_docs()
```

**Value**

None. Opens a browser window.

**Examples**

```
## Not run:
ns_api_docs()

## End(Not run)
```

---

ns_auth_token	<i>Retrieve access token</i>
---------------	------------------------------

---

**Description**

After creating a client in Nettskjema, this function will retrieve the access token needed for the remaining processes in the package. Automatically caches the token for more efficient API usage.

**Usage**

```
ns_auth_token(
  client_id = Sys.getenv("NETTSKJEMA_CLIENT_ID"),
  client_secret = Sys.getenv("NETTSKJEMA_CLIENT_SECRET"),
  cache = TRUE,
  cache_path = NULL
)
```

**Arguments**

client_id	Character. Default assumes this is stored in .Renviron as "ns_CLIENT_ID"
client_secret	Character. Default assumes this is stored in .Renviron as "ns_CLIENT_SECRET"
cache	Logical. Should the token be cached?
cache_path	Character. File path to where the token should be stored. Defaults to system cache directory.

---

ns_create_client	<i>Create a Nettskjema Client</i>
------------------	-----------------------------------

---

### Description

Opens the Nettskjema client creation URL in the default web browser.

### Usage

```
ns_create_client()
```

### Details

Use this function to create a new client for accessing Nettskjema APIs.

### Value

NULL. Opens the client creation page in a browser.

### Examples

```
## Not run:  
ns_create_client()  
  
## End(Not run)
```

---

ns_get_attachment	<i>Save Nettskjema attachments to file</i>
-------------------	--

---

### Description

The Nettskjema forms has an option to upload attachments with forms. These can be retrieved by this function. Recommended workflow is to first call [ns\\_list\\_submission\\_attachments](#), and use the output of this to pass along to the function. This function is called by [ns\\_get\\_form\\_attachments](#), but you can use it to define your own output file names.

### Usage

```
ns_get_attachment(attachment_id, path = NULL)
```

### Arguments

attachment_id	integer. Unique ID of attachment, as found in the submission data.
path	Character. Location to save the file.

**Value**

httr2-response. saves attachment to file.

**Examples**

```
## Not run:  
#submission_id <- c(22222, 1232, 21555)  
#attach_dt <- ns_list_submission_attachments(submission_id)  
#ns_save_attachment(path = attach_dt$path, output = attach_dt$standardized)  
  
## End(Not run)
```

---

ns_get_codebook	<i>Get codebook for a form</i>
-----------------	--------------------------------

---

**Description**

With the given API token, will retrieve a list of all the forms you have access to

**Usage**

```
ns_get_codebook(form_id, asis = FALSE)
```

**Arguments**

form_id	integer. Number of the form to retrieve
asis	logical. Whether to return the raw content from nettskjema. Default is FALSE.

**Value**

list of class ns\_meta\_data

**Examples**

```
## Not run:  
codebook_110000 <- ns_get_codebook(110000)  
  
## End(Not run)
```



---

ns_get_data	<i>Get data from a form</i>
-------------	-----------------------------

---

## Description

This function connects to a specific form and fetches all answers. The responses are placed in a data.frame for easy use.

## Usage

```
ns_get_data(form_id, type = NULL)
```

```
ns_get_submissions(form_id, type = NULL)
```

## Arguments

form_id	integer. Number of the form to retrieve
type	Character or NULL. One of either "original" or "long".

## Details

**"original"** - Returns data in the same tabular format as delivered by Nettskjema.

**"long"** - Returns the data in tall format, where there are multiple rows per participant (one per question) and each choice is timestamped.

## Value

data.frame

## Functions

- ns\_get\_submissions(): Alternative function name, does the same.

## Examples

```
## Not run:  
  
# Retrieve all data  
data_110000 <- ns_get_data(110000)  
  
## End(Not run)
```

---

ns_get_forms	<i>Get all forms you have access to</i>
--------------	---

---

### Description

All users have access to specific forms, and this function uses the token provided to check what the current user has access to.

### Usage

```
ns_get_forms(asis = FALSE)
```

### Arguments

asis                   logical. Whether to return the raw content from nettskjema. Default is FALSE.

### Value

data.frame, if asis = TRUE returns a list

---

ns_get_form_attachments	<i>Retrieve all attachments associated with a Nettskjema</i>
-------------------------	--

---

### Description

Some Nettskjema have fields for attachments. These can be downloaded through this function. There are two naming convention that may be used for the saved files' names. See output of [ns\\_list\\_submission\\_attachments](#) for idea of the names used.

### Usage

```
ns_get_form_attachments(
  form_id,
  filenames = c("standardized", "original"),
  output_dir = "."
)
```

### Arguments

form\_id               integer. Number of the form to retrieve

filenames            Character of either 'standardized' (default) or 'original' indicating which file names to use.

output\_dir           directory to output the files to

## Details

filenames types

**"original"** - uses file names as they were uploaded to Nettskjema

**"standardized"** - creates file names based on the submission id and a counter number to uniquely create file names for each submission (in case there are more than one attachment)

## Value

invisible named logical if the file was saved successfully

## Examples

```
## Not run:
form_id <- 1100000
ns_get_form_attachments(form_id)

# save files to specific folder
ns_get_form_attachments(form_id, output_dir = "~/Desktop")

# save using original file names
ns_get_form_attachments(form_id, filenames = "original")

## End(Not run)
```

---

ns\_get\_form\_reports     *Download files associated with a form*

---

## Description

Each Nettskjema form has several auxiliary forms that can be retrieved.

## Usage

```
ns_get_form_reports(
  form_id,
  type = c("csv", "excel", "spss"),
  path = report_path(form_id, type)
)
```

## Arguments

form_id	integer. Number of the form to retrieve
type	Character. Either "csv", "excel" or "spss".
path	Character. Full path to where the downloaded file should be saved. Defaults to current directory with correct extension.

**Value**

httr2-response

**Examples**

```
## Not run:
ns_get_form_reports(10009, type="csv")
ns_get_form_reports(10009, type="excel")
ns_get_form_reports(10009, type="spss")
ns_get_form_reports(
  10009,
  type="spss",
  path = "~/Desktop/10009/form.sav"
)

## End(Not run)
```

---

ns\_get\_me

*Information about the current user*

---

**Description**

Based on login credential, returns information on the user accessing the Nettskjema PI

**Usage**

```
ns_get_me()
```

**Value**

list of information

**Examples**

```
## Not run:
ns_get_me()

## End(Not run)
```

---

ns_get_meta	<i>Get metadata for a form</i>
-------------	--------------------------------

---

**Description**

With the given API token, will retrieve the meta-data connected to a form.

**Usage**

```
ns_get_meta(form_id)
```

**Arguments**

form\_id            integer. Number of the form to retrieve

**Value**

list of class ns\_meta\_data

**Examples**

```
## Not run:  
meta_110000 <- ns_get_meta(110000)  
  
## End(Not run)
```

---

ns_get_submission	<i>Get an individual submission answer</i>
-------------------	--

---

**Description**

Get an individual submission answer

**Usage**

```
ns_get_submission(submission_id)
```

**Arguments**

submission\_id    integer. Unique identifier of a specific Nettskjema submission.

**Value**

list of all answers

**Examples**

```
## Not run:  
ns_get_submission(1100)  
  
## End(Not run)
```

---

```
ns_get_submission_attachments  
Get Nettskjema attachments for a submission
```

---

**Description**

Each unique submission ID in Nettskjema may have some attachments. This function saves all attachments associated with a submission ID.

**Usage**

```
ns_get_submission_attachments(  
  submission_id,  
  filenames = "standardized",  
  output_dir = paste0("./", submission_id)  
)
```

**Arguments**

submission_id	integer. Unique identifier of a specific Nettskjema submission.
filenames	Character of either 'standardized' (default) or 'original' indicating which file names to use.
output_dir	directory to output the files to

**Value**

a data.frame with information on attachments available, files written to path

**Examples**

```
## Not run:  
submission_id <- 22222  
ns_get_submission_attachments(submission_id)  
ns_get_submission_attachments(submission_id, "original")  
  
## End(Not run)
```

---

`ns_list_form_attachments`*Retrieve all attachments associated with a Nettskjema*

---

### Description

Some Nettskjema have fields for attachments. These can be downloaded through this function. There are two naming convention that may be used for the saved files' names. See output of [ns\\_list\\_submission\\_attachments](#) for idea of the names used.

### Usage

```
ns_list_form_attachments(form_id)
```

### Arguments

`form_id` integer. Number of the form to retrieve

### Details

filenames types

**"original"** - uses file names as they were uploaded to Nettskjema

**"standardized"** - creates file names based on the submission id and a counter number to uniquely create file names for each submission (in case there are more than one attachment)

### Value

invisible named logical if the file was saved successfully

### Examples

```
## Not run:  
form_id <- 1100000  
ns_list_form_attachments(form_id)
```

```
## End(Not run)
```

---

`ns_list_submission_attachments`*List Nettskjema attachments for a submission*

---

**Description**

Each unique submission ID in Nettskjema may have some attachments. These can be listed using this function, which will provide the information on where in the Nettskjema API the files are located, their original file names, and a suggested standardized file names for tidier data output.

**Usage**

```
ns_list_submission_attachments(submission_id)
```

**Arguments**

`submission_id` integer. Unique identifier of a specific Nettskjema submission.

**Value**

a data.frame with information on attachments available.

**Examples**

```
## Not run:
submission_id <- 22222
ns_list_submission_attachments(submission_id)

## End(Not run)
```

---

`ns_req`*Basic Nettskjema htr2 request*

---

**Description**

Sets the API url and access token. This is used for almost all the functionality in this package. If you want to use endpoints not yet covered by functions in the package, this is the basis function you can build from.

**Usage**

```
ns_req(...)
```

**Arguments**

`...` arguments passed to `ns_auth_token`



**Value**

httr2 request

**Examples**

```
## Not run:
ns_req() |>
  httr2::req_url_path_append("form", "me") |>
  httr2::req_perform() |>
  httr2::resp_body_json()

## End(Not run)
```

---

ns_write_codebook	<i>Write codebook to file</i>
-------------------	-------------------------------

---

**Description**

Save codebook information to a file for safe keeping. Depending on the type of codebook (raw or tidy) the file will either be stored as a json-file through [write\\_json](#) or a text table through [write.table](#).

**Usage**

```
ns_write_codebook(codebook, path, ...)

## Default S3 method:
ns_write_codebook(codebook, path, ...)

## S3 method for class 'ns_codebook_raw'
ns_write_codebook(codebook, path, ...)

## S3 method for class 'ns_codebook'
ns_write_codebook(codebook, path, sep = "\t", ...)
```

**Arguments**

codebook	object of class ns_codebook
path	filename or path
...	arguments to <a href="#">write_json</a> or <a href="#">write.table</a>
sep	character. If writing text table, column delimiter.

**Details**

Given the two types of codebooks, writes different types of files.

**codebook** - writes a tab-separated table

**codebook\_raw** - writes a json-file

**Value**

no return value. Writes a file to path.

**Examples**

```
## Not run:
form_id <- 1100000
my_cb <- ns_get_codebook(form_id)
ns_write_codebook(my_cb, "my/path/codebook_110000.txt")

## End(Not run)
```

---

ns_write_meta	<i>Write meta-data to file</i>
---------------	--------------------------------

---

**Description**

Save meta-data information to a file for safe keeping. The function saves the data in json-format, which best preserves the information.

**Usage**

```
ns_write_meta(meta, path, ...)

## Default S3 method:
ns_write_meta(meta, path, ...)

## S3 method for class 'ns_meta'
ns_write_meta(meta, path, ...)
```

**Arguments**

meta	object of class ns_meta class
path	file name or path
...	other arguments to <a href="#">write_json</a>

**Value**

no return value. Writes a file to path.

**Examples**

```
## Not run:
form_id <- 1100000
my_meta <- ns_get_meta(form_id)
ns_write_meta(my_meta, "my/path/meta_110000.json")

## End(Not run)
```

# Index

codebook, [2](#)

get\_raw\_codebook, [3](#)

has\_codebook, [4](#)

ns\_add\_labels, [4](#)

ns\_alter\_checkbox, [5](#)

ns\_api\_docs, [6](#)

ns\_auth\_token, [6](#)

ns\_create\_client, [7](#)

ns\_get\_attachment, [7](#)

ns\_get\_codebook, [8](#)

ns\_get\_data, [9](#)

ns\_get\_form\_attachments, [7](#), [10](#)

ns\_get\_form\_reports, [11](#)

ns\_get\_forms, [10](#)

ns\_get\_me, [12](#)

ns\_get\_meta, [2](#), [13](#)

ns\_get\_submission, [13](#)

ns\_get\_submission\_attachments, [14](#)

ns\_get\_submissions (ns\_get\_data), [9](#)

ns\_list\_form\_attachments, [15](#)

ns\_list\_submission\_attachments, [7](#), [10](#),  
[15](#), [16](#)

ns\_req, [16](#)

ns\_write\_codebook, [17](#)

ns\_write\_meta, [18](#)

write.table, [17](#)

write\_json, [17](#), [18](#)