AmCAT manual, concept version April 2013

Table of contents **page**

1. Getting started

1.1 AmCAT versions, reporting problems

At the moment of writing, there are two versions of AmCAT: http://amcat-dev.labs.vu.nl and http://amcat-production.labs.vu.nl. The production.labs version is the most recent of the two (April 2013), but this might change in the future. See the AmCAT mailing on https://groups.google.com/forum/?fromgroups#!forum/amcat-dev for updates on AmCAT, including which version is the most recent. When AmCAT works very slowly or does not function, it might help to try the other version. Trying another browser can also help in such cases. AmCAT is tested in the most recent versions of Mozilla Firefox, Google Chrome and Internet Explorer. Internet Explorer 6, 7 and 8 are not recommended.

You can report bugs, problems and suggestions for future updates regarding AmCAT by creating an issue on the website https://code.google.com/p/amcat/issues/list (select ' new issue'). The AmCAT development team will send you an email once your issue has been resolved.

1.2 Logging in

The screenshot below shows the AmCAT login screen. In case you already have an AmCAT username and password, you enter these here. When you do not have a username and password yet, you can create a new account through 'register'. After doing so, a link that you can follow to activate your account will be sent to your email address. You can then start using AmCAT immediately.

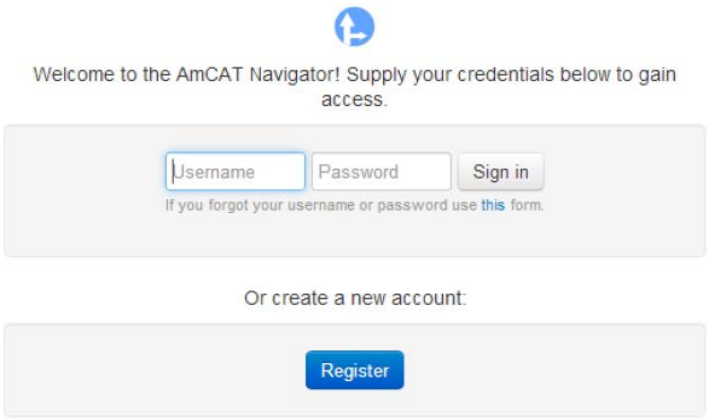


Figure 1. The AmCAT3 login screen.

1.3 The start screen

AmCAT is divided into two parts: the AmCAT Navigator, where you can create and manage projects or do automatic content analysis, and the AmCAT Annotator, where you can code articles manually. The start screen is part of the AmCAT Navigator. From the start screen, you can either go to the overview of your projects or go directly to the codingjobs assigned to you in the Annotator. Moreover, you can consult several lists that give an overview of all resources in the AmCAT database, and change the details of your account.

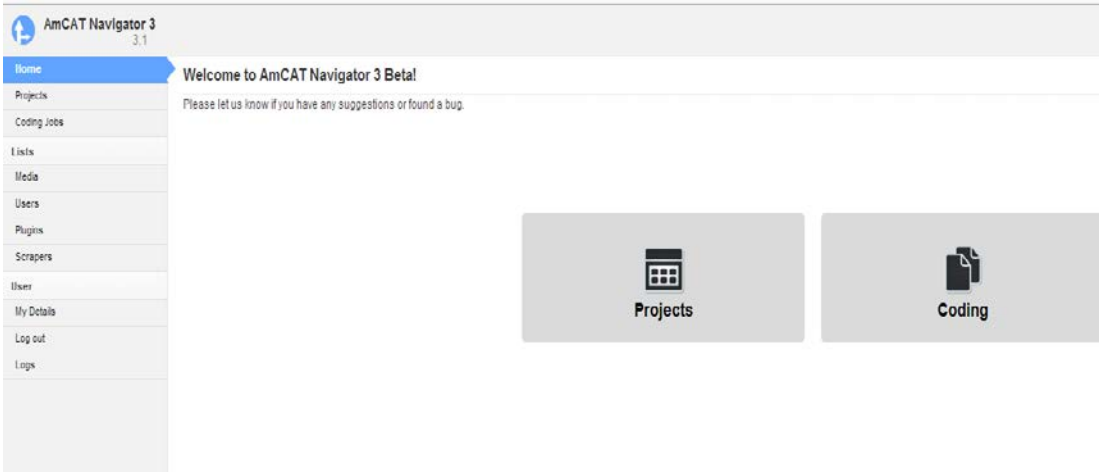


Figure 2. The AmCAT start screen.

Click ‘projects’ in menu on the left side of the screen or on the panel ‘Projects’ in the middle of the screen to go to an overview of your projects. Click ‘coding jobs’ on the left side of the screen or on the panel ‘Coding’ in the middle of the screen to go to the Annotator and start coding.

1.4 Changing your account details

Click ‘my details’ in the menu on the left side of the screen to see an overview of your AmCAT account details. Here you can change these details and save the changes by clicking ‘update user details’. The password in this overview is **not** your AmCAT password. If you want to change your password, you can do so by clicking ‘change password’ on the upper side of the screen.

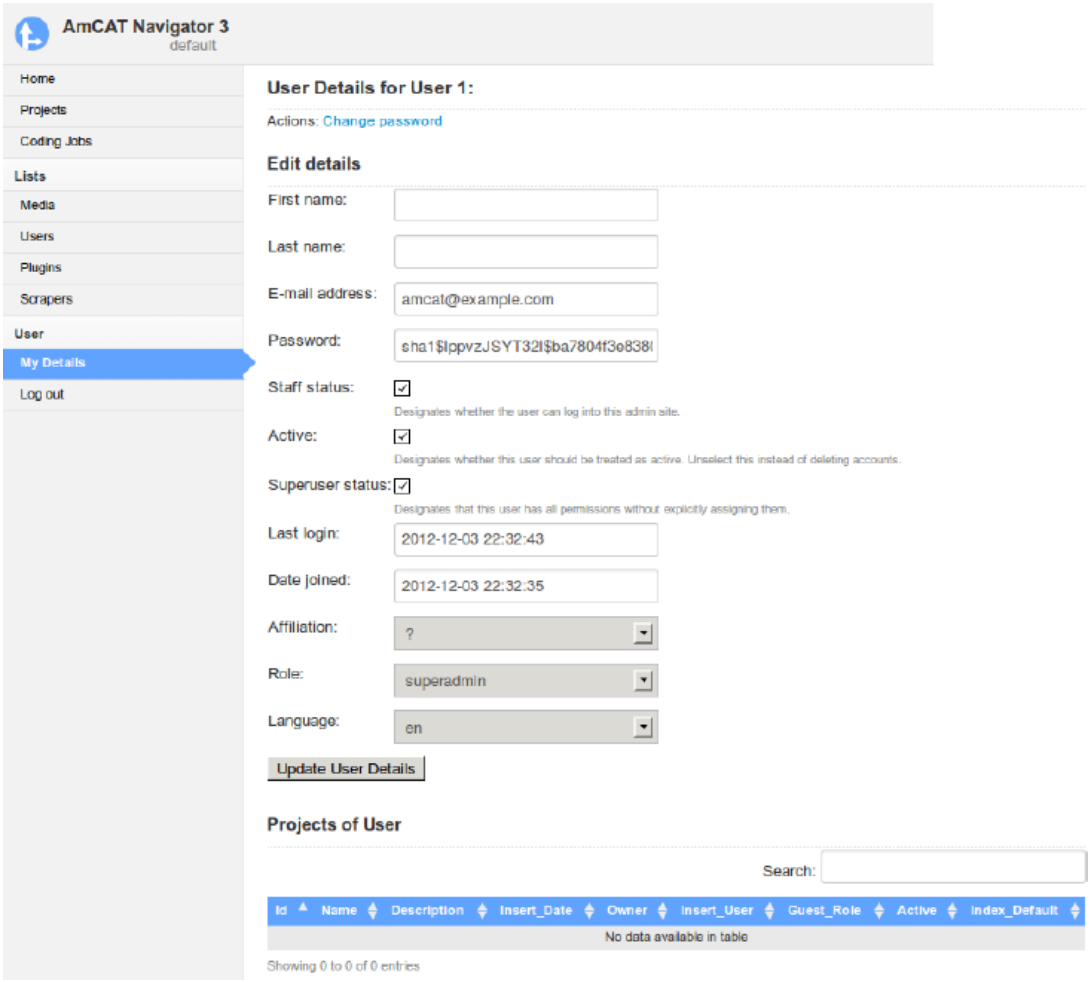


Figure 3. AmCAT account details.

Clicking 'change password' will lead you to a screen where you can change your password. Click 'change my password' here after filling out the form, and you can use your new password immediately. In case you forgot your old password, you can reset it by using the link on the login screen.

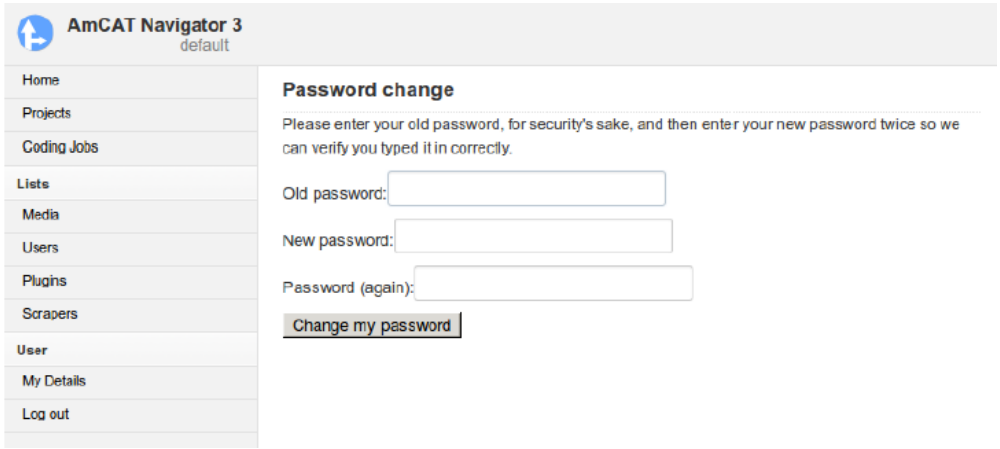


Figure 4. Changing your password.

N.B. It is not possible to delete your AmCAT account. In case you no longer want to use AmCAT, you can make your account inactive in the overview of your account details. You can do so by deselecting 'active' and then clicking on 'update user details'. This makes your account inactive, which means your username is no longer shown in lists of active users that can be added to projects.

1.5 Logging out

You can log out from AmCAT through clicking on 'Log out', the second to last option in the menu on the left side of the screen.

2. Projects in AmCAT

AmCAT categorizes all data and resources in projects. Articles, codingschemas and codings are all tied to a specific project. When you start using AmCAT for your own research, creating a new project is the first thing you'll have to do. Coders or researchers who join an existing project, will find all the data and resources they need categorized under that project in AmCAT.

2.1 Adding, selecting and changing projects

Select 'projects' in the menu to the left of the screen to go to the overview of all projects in AmCAT. Here you can select the project you want to contribute to, or add a new project.

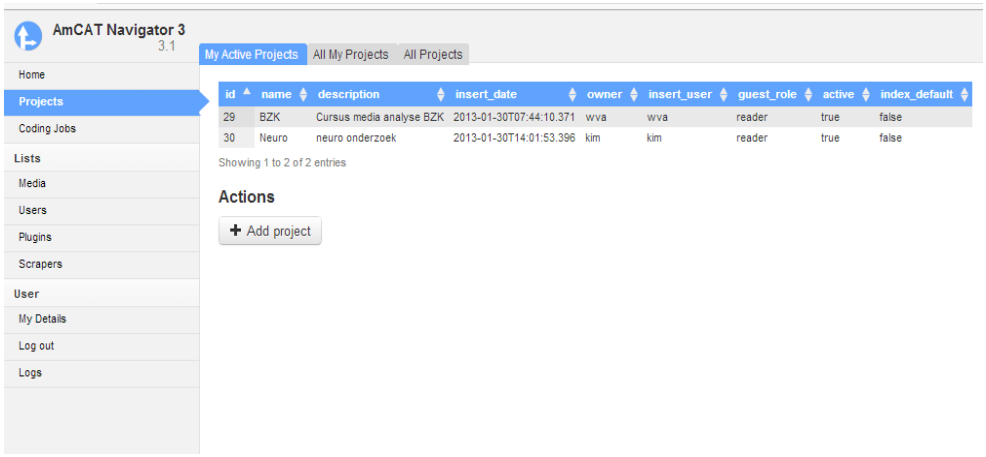


Figure 5. Overview of different projects.

This screen shows a table that contains all active projects that the user is added to. Every project has a unique number (id, the leftmost colunm in the table), as well as a name (the column 'name'). Click on a row in the table to go to the overview of that project. You can add a new project by clicking on the button 'add project' below the table.

Below you will find an overview of all columns in the projects overview table, with a short explanation of what they mean. 'Id' and 'name' are, as mentioned above, the number and name of the project.

- **Description**: a short description of the project. You have to include this when adding a new project.
- **Insert_date**: the date and time that the project was added.
- **Owner**: the owner of the project.
- **Insert_user:** the user who added the project.
- **Guest_role**: the role within the project of users who are not added to the project. A ' role' in AmCAT stands for the degree of access to the resources that are part of that project. The standard role of guest users is 'reader': guest users can read all resources linked to the project (including article texts) but they cannot make any changes. Chapter 8 describes the different user roles.
- **Active**: this signifies whether a project is active (true) or inactive (false). You can switch between active and inactive by going to the overview screen of the specific project, and

there click on ‘edit details’. Here, you can select or deselect the ‘ active’ option. Inactive projects do not appear in the overview ‘my active projects’. Whenever a project is finished or abandoned, you can give it the ‘inactief’ status and decrease the number of projects that appear in your active projects list.

- **Index_Default** : this shows whether indexing the articlesets in this projects is standard (true) or not (false). Indexed articlesets can be used as a search index, articlesets that are not indexed cannot. If you want to use AmCAT’s query function, having articlesets indexed by default is the preferred option. Indexing does take some time, however. Again, you can change this option by going to ‘edit details’ from the project overview. Here, you can select or deselect ‘ index_default’ .

In the blue line above the table, you see two small blue triangles on top of each column. By clicking on one of the triangles, you can sort the information in the table according to the variable in that row. The upper triangle is for a descending order, the lower one for an ascending order.

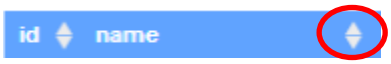


Figure 6. Click one of the triangles to sort the table according to the corresponding column.

In this way, you can quickly find information in larger tables. The sorting system applies to all overview tables in AmCAT.

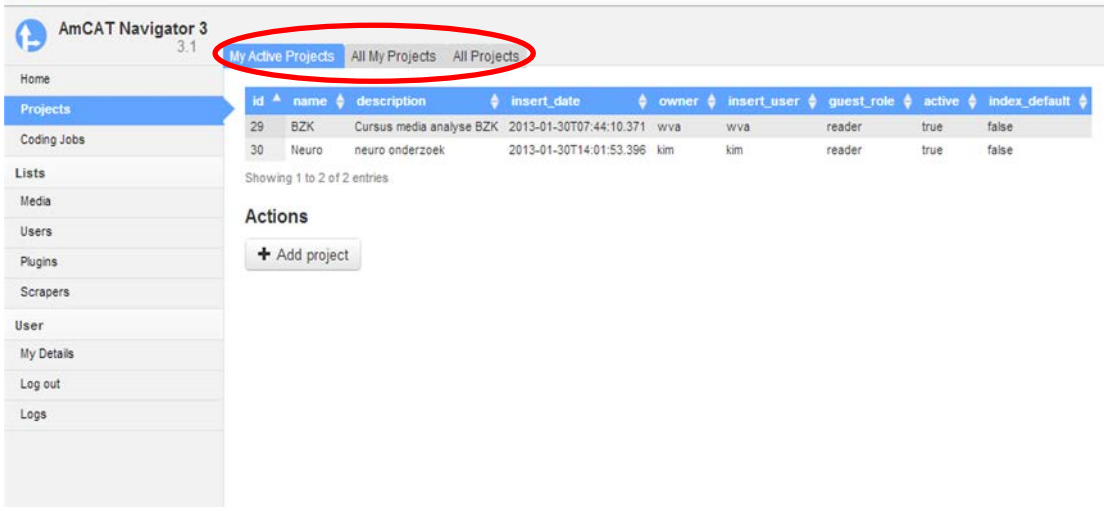


Figure 7. Tabs for changing between lists.

Above the project overview, there are three tabs. ‘My active projects’ shows all active projects of which the user is member, ‘all my projects’ shows the total of projects for which the user is a member, and ‘all projects’ shows all existing projects in AmCAT.

N.B.: When a user is not a member of a project (that is, he or she is not added to a project by the project owner), the project will not appear in the lists of ‘my active projects’ and ‘my projects’ for that user. You can add users to a project by selecting the project, and then clicking on the tab ‘users’. Here, you can make other users a member of the project through ‘add user’. See chapter 8 for a more detailed explanation regarding projects and users.

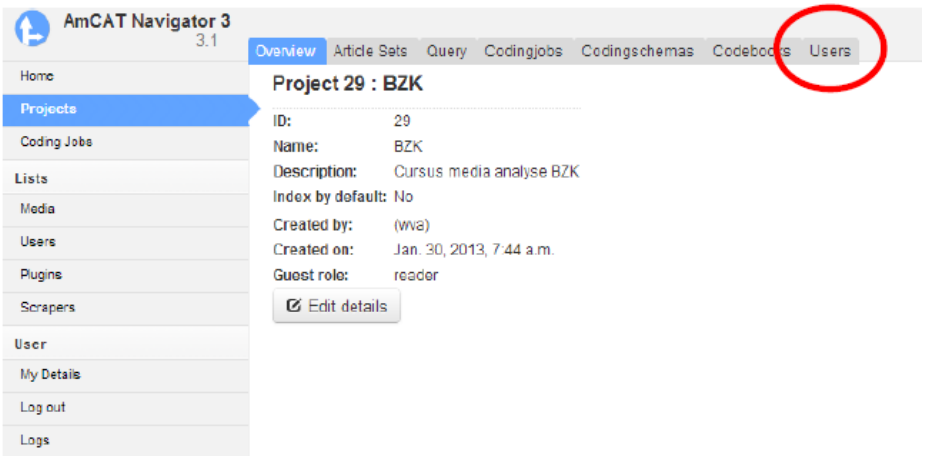


Figure 8. Through the tab 'users' in the project overview, you can add or remove users from a project.

Add project: When you want to create a new project in AmCAT, you click on 'add project', below the table that lists your projects. In the next screen, you give the project a name and a description. You select whether the project is active or not (active) and whether or not you want new articlesets to be indexed (index by default). If you want to use AmCATs query function, either for automatic content analysis or for making codingjobs, it is advisable to turn indexing by default on.

2.2 AmCAT-terminology

A project in AmCAT consists of articles, articlesets, codingschema's, codebooks, codingjobs, search indexes and users. A short explanation of these terms:

- An **article** is a text and its metadata, such as the date of publication, the medium of the text, its author and the number of words in the article.

- An **articleset** consists of one or more articles who are categorized into the same set in AmCAT, for example because they happened to be uploaded as a set, or because they are purposedly categorized together. One article can be part of multiple articlesets, and one articleset can contain articles of different media. Moreover, it is possible to add articles to existing articlesets. When an articleset is indexed (see paragraph 3.2) it can be used as a search index. An articleset can also be used as a basis for a codingjob (see chapter 5).

- A **codingschema** is a scheme used for coding, that consists of a number of variables. These variables can have different levels of measurement, such as answer categories or numerical values. This coding scheme is stored as a whole. Codingschemas are further divided into articleschemas and unitschemas. An **articleschema** is used for codings at the article level, while a **unitschema** is used for codings on the (nuclear)sentence level.

- A **codebook** is a set of answer categories, that can be attached to one or more variables in a codingschema. Codebooks can be part of a codingschema, but they are also stored

separately in AmCAT. Whenever you make a new codingschema, you can add previously used codebooks to that schema, also those used in other projects.

- A **codingjob** consists of an articleset that is linked to a codingschema (meaning, an articleschema and a unitschema) and a user (the coder). Codingjobs are used for manual coding in the AmCAT Annotator. Paragraph 5.1 gives an explanation of how you create codingjobs, and in chapter 9 we explain coding using the AmCAT Annotator.

- A **search index** is an indexed articleset. Through the tab 'query' you can search in a search index or use a number of techniques for automatic content analysis (see chapter 4).

- A **user** is a user of AmCAT who can be added to a project. Users can have different 'roles' within a project, determing their degree of access (see chapter 8). Regardless of their role, users can use search indexes and see the project's codebooks and codingschemas.

N.B. AmCAT often works with IDs: numbers for each articleset, medium, codingjob, codingschema and so on. Codes in a codebook also have ID numbers, which are not shown in AmCAT. All ID numbers are, however, used when exporting the data to Excel or SPSS. In SPSS, you can use these ID numbers in your syntax. For example 'select if medium = 4' selects all cases for which the medium ID is 4, which is the Dutch newspaper *de Volkskrant*. Set the display options in SPSS to 'values and labels' to see both the ID number and the label for each code.

3 Project overview and articlesets

When you click on the name of a project in the overview of all projects, you will see a more detailed overview of the specific project. From this screen, you can edit the project's details as well as select a number of tabs where you can perform further actions. These are discussed in the following chapters. Here, we will explain the project overview and the first tab 'article sets'.

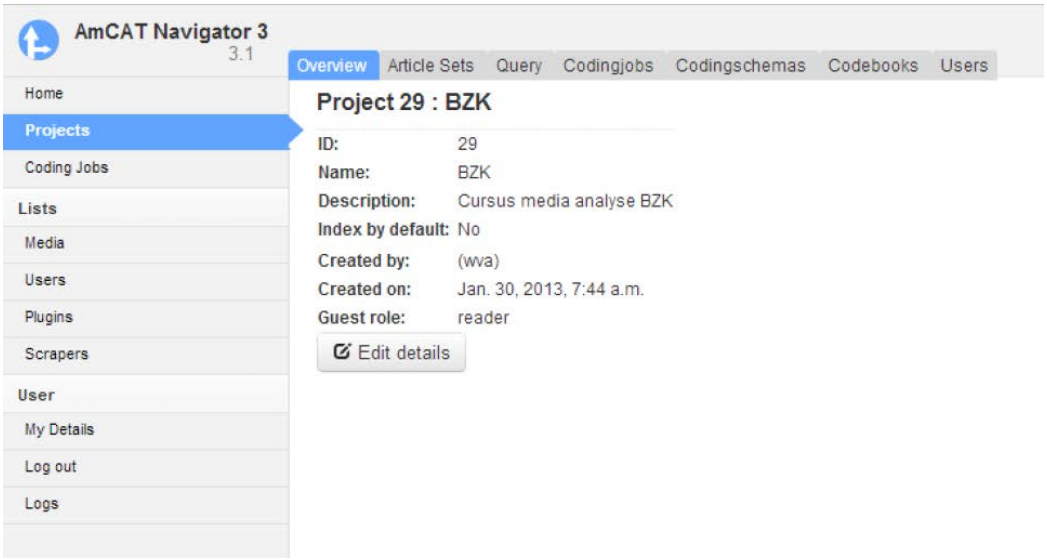


Figure 9. Overview of a project in AmCAT.

The project details in this overview are identical to those in the table that shows all projects, discussed in the previous chapter. Via 'edit details ' you can change a number of these details: the name and the description of the project , the role of guest users, as well as the project's status as active or inactive. Additionally, you can turn indexing articlesets in that project on or off.

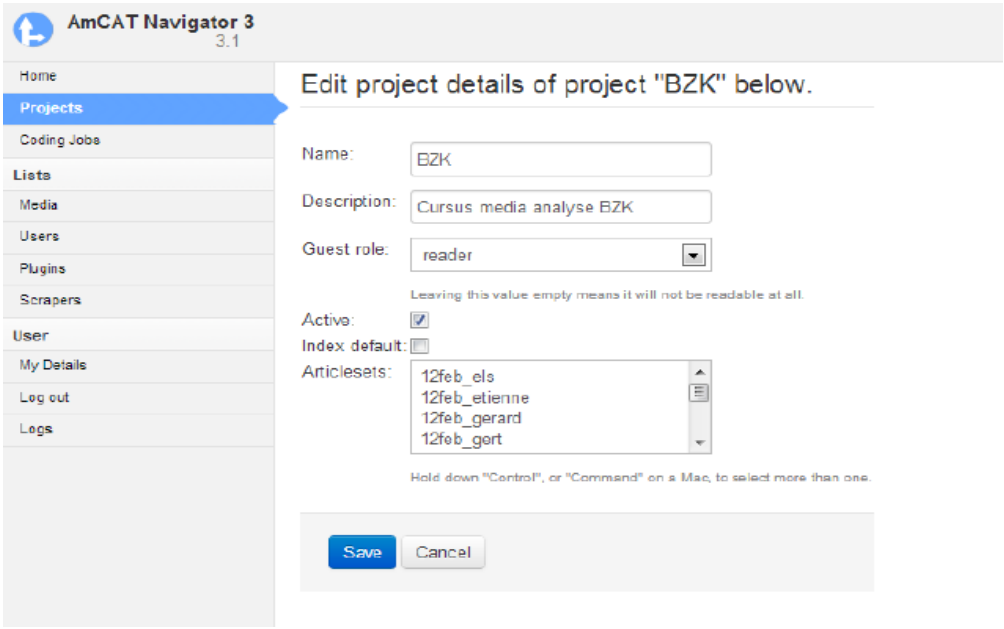


Figure 10. Editing project details.

In the field called ‘articlesets’, you see a list of articlesets categorized under that project. In case you want to remove articlesets from the project, you select the articlesets **that you want to keep** (hold down ctrl when selecting the sets if you want to select more than one). The other articlesets are removed from AmCAT! Changes to a project not applied until you click ‘save changes’.

3.1 Articlesets

The tab ‘Article Sets’ leads you to an overview of all articlesets categorized under that project.

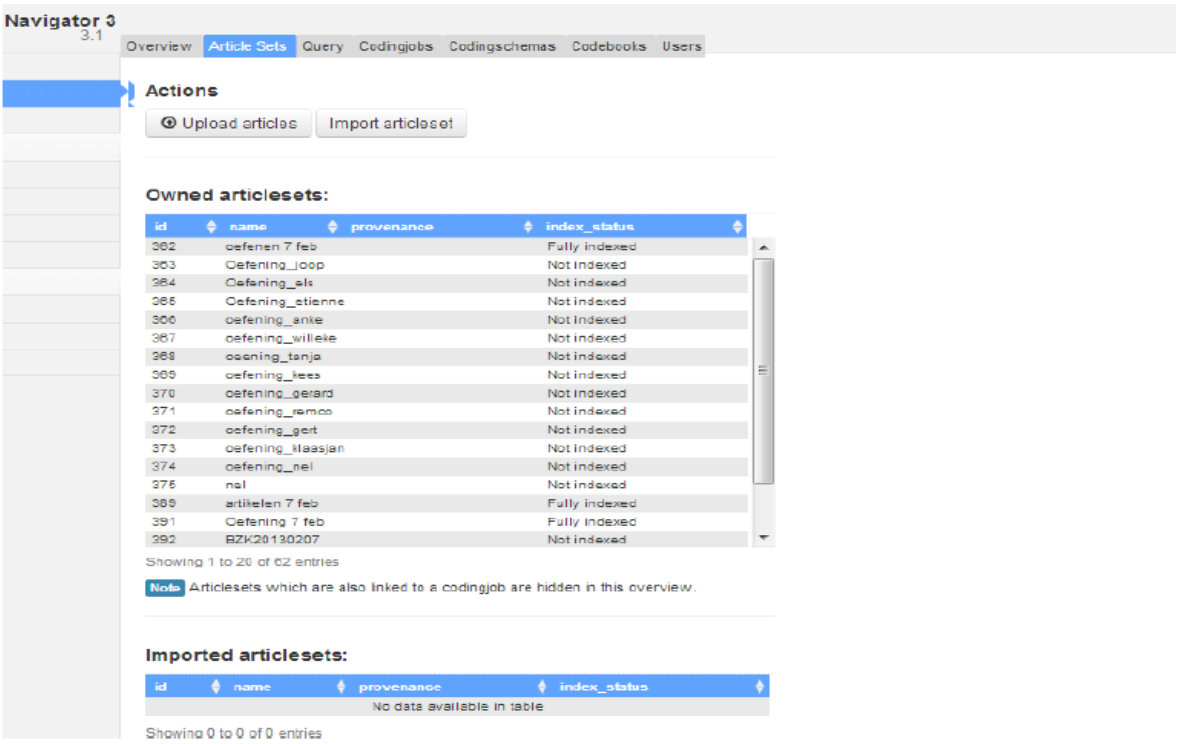


Figure 11. Overview of articlesets categorized under a specific project in AmCAT.

On the upper side of the screen, under the heading ‘actions’, there is a button for uploading articles into a new articleset, as well as importing articlesets from another project. These functions are explained in paragraph 3.3 en 3.4, respectively. An ‘owned’ articleset is an articleset that is created in the current project, an ‘imported’ articleset is imported from another project.

N.B. Articlesets assigned to a codingjob are not visible in this overview. You can consult these under the tab ‘codingjobs’.

The overview of owned and imported articlesets gives for each articleset the following details:

- I**D:** the unique identification number of each articleset
- **Name**: the name given to the articleset when it was created
- **Provenance:** a description of or comments on the articleset. You can add these through clicking on ‘edit details’ in the detailed overview of an articleset.

- **Index_Status**: ‘fully indexed’ means that an articleset is indexed, ‘indexing’ means that it is in the process of indexing (which means it cannot be used), ‘not indexed’ means that the articleset is not (yet) indexed.

Click on an articleset in the list to go to a detailed overview of that articleset.

3.2 Detailed overview of an articleset

From this screen, you can take a number of actions in order to edit a specific articleset. The table on the lower half of the screen gives an overview of all articles included in the articleset. The actions you can take from this screen are the following:

Edit details: Here you can change the name and/or the provenance of the articleset.

Enable indexing: This turns indexing off or on for that articleset.

Deduplicate: Deduplicating an articleset cleans it from articles that are included more than once. Duplicate articles occur regularly, especially when you use articles from databases such as LexisNexis. As duplicate articles can distort the results from automatic text analyses, it is advised that you check each articleset for duplicates before using it as a search index, and to deduplicate the set if necessary. Deduplicating articlesets cannot be made undone. AmCAT asks for an additional confirmation before starting the deduplication process.

Delete: This deletes the articleset from AmCAT. In general, deleted articlesets can no longer be brought back! Again, AmCAT asks an additional confirmation before deleting the set. Only users with the admin-role in a project can delete that project’s articlesets.

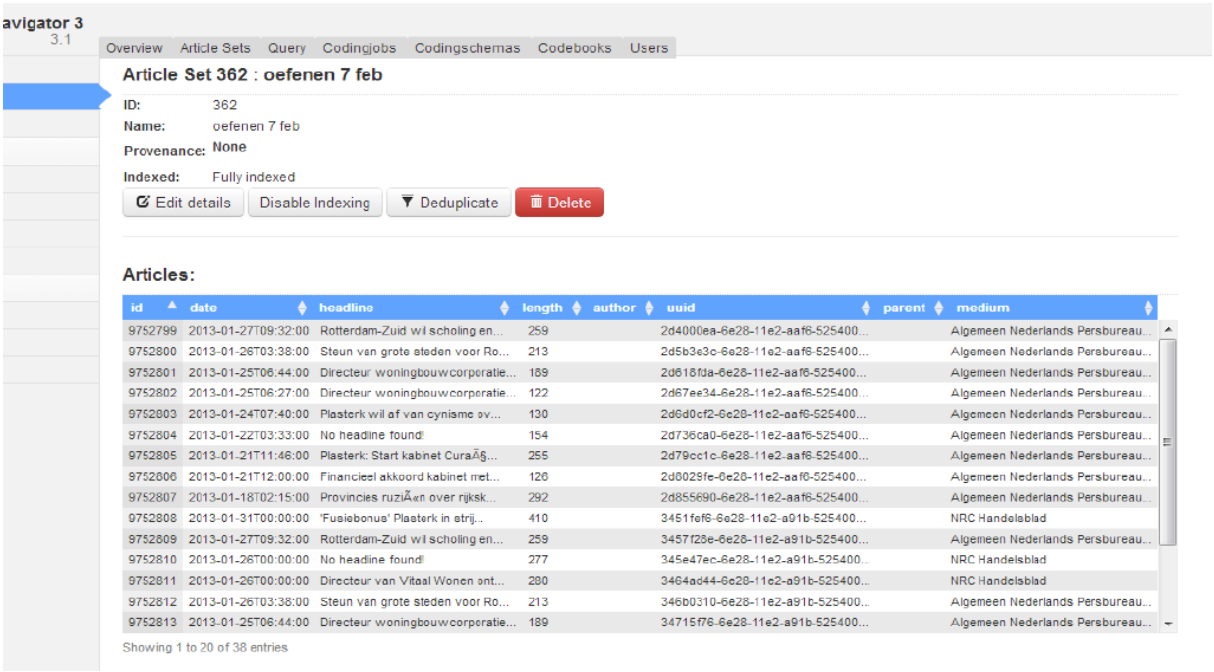


Figure 12. Detail overview of an articleset.

The table in the lower half of the screen shows the following metadata for each article in the set:

- **ID**: the unique number for each article in the set (for the current AmCAT server)
- **Date**: the date and time (if available) the article was published
- **Headline**: the article's headline
- **Length**: length of the article in number of words
- **Author**: the article's author
- **UUID**: the unique number for each article in the set (the same across all AmCAT-servers)
- **Parent**: in the case of internet sources (news sites or blog posts with comments) each comment is included as a separate article in AmCAT. In such cases, the parent is the original article or blog post the comment refers to.
- **Medium**: the medium in which the article is published.

By clicking on an article in the table you will see the article text and metadata, as shown in this screenshot.

Figure 13. Article text and metadata.

The ' in articlesets' metadata category to the right of the text does not work yet (April 2013)– in a future update to AmCAT, this will show all articlesets in which the article is included.

3.3 Uploading articles

In order to upload articles into AmCAT, you have to go back to the overview of all articlesets in your project, under the tab 'Article Sets'. Here, you click on the 'upload articles' button, which will show a list of different scripts that you can use to upload articles into AmCAT. Which script to use depends on the file format of the articles you want to upload. The .txt format script is for entering individual articles that use a plain text format, while the script for LexisNexis articles can upload a set with multiple articles from the LexisNexis in one go (file extension .txt, cover page should be included).

A number of projects have their own custom uploader plugin. As these plugins depend on a specific page layout, they might not work correctly when the articles you want to upload are even slightly different from this layout. If you encounter problems when uploading articles, you can report these as an issue on the AmCAT wiki (see page 2).

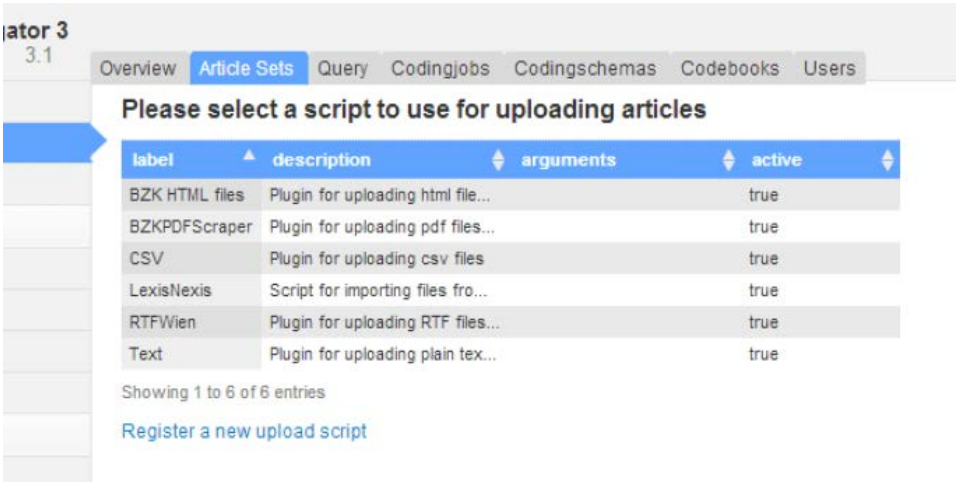


Figure 12. Selection of scripts for uploading articles into AmCAT.

Select the script you want to use, which will lead to the following screen:

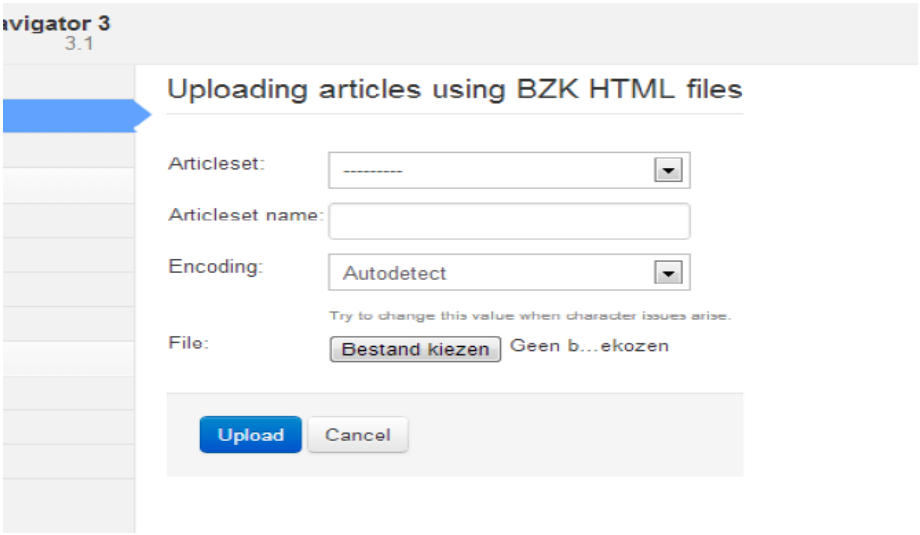


Figure 13. Uploading articles into an existing or new articleset.

Here, you will have to provide the folllowing details for articles you want to upload:

Articleset: If you want to add articles to an existing articleset, you select this set from the list in this field. The field 'articleset name' should be left empty.

Articleset name: In case you want to make a new articleset from the uploaded articles, you leave the field 'articleset' empty. You can enter the name of the new articleset in the field 'articleset name'.

Encoding: The default setting 'autodetect' means that AmCAT automatically detects what encoding to use when displaying the articles. You can also select the encoding manually, for which the options are ISO-8859-15, UTF-8, and Latin-1. When you upload articles and you see that certain letters (such as é,ö,ç, etc.) are displayed incorrectly, you can select one of these encoding options.

File: Here you select the file that contains the article(s) you want to upload. Click on 'upload' to start uploading.

N.B.: 'Cancel' only works before you start uploading. Also, it is not possible to remove articles from an articleset, so make sure you have the right articles and the right articleset before starting the upload! Making several smaller articlesets instead of adding to a larger one might be the safest option.

3.4 Importing an articleset from another project

In order to import articlesets from another project into the current project, you again go back to the overview of all articlesets under the ' Article Sets' tab. Here, you click on 'import articleset' , which brings up a list of articlesets from other projects that you can select and import. Select an articleset and it will be imported into the current project immediately. This can be undone by clicking 'unlink' in the detailed overview of that articleset.

N.B. To complete the importation of an articleset, you still have to assign the articleset to the current project. To do so, go to the overview of all articlesets, select the imported articleset from the list, and click on 'edit details'. In the field 'project', you select the current project from the list.

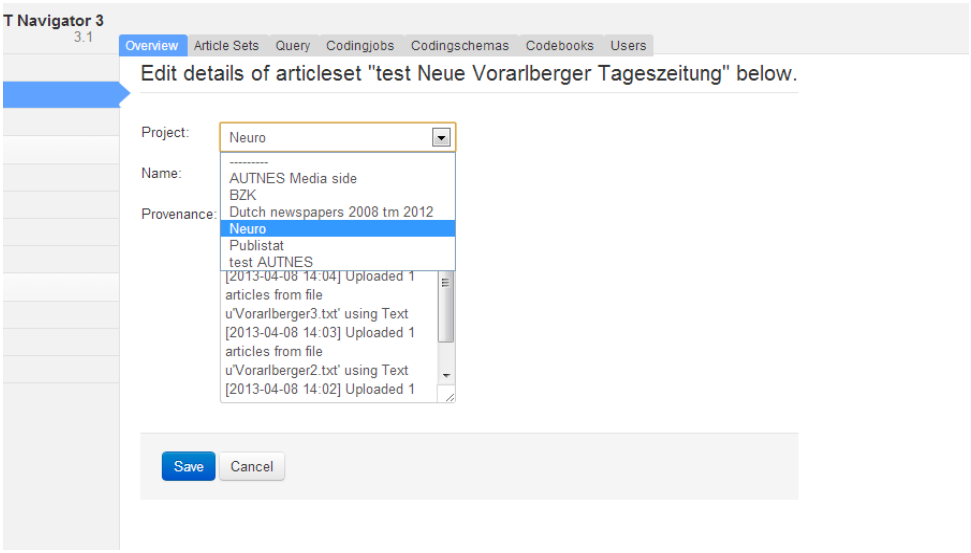


Figure 14. Assigning an imported articleset to the current project.

Click on 'save details' and the imported articleset is ready for further use in the current project, as codingjob or search index.

4. Queries in AmCAT

From the tab ‘Query’ you can use searchstrings or article IDs to search in search indexes categorized under the current project. In order to be used as a search index, an articleset should have the status ‘fully indexed’, which is shown in the overview of articlesets.

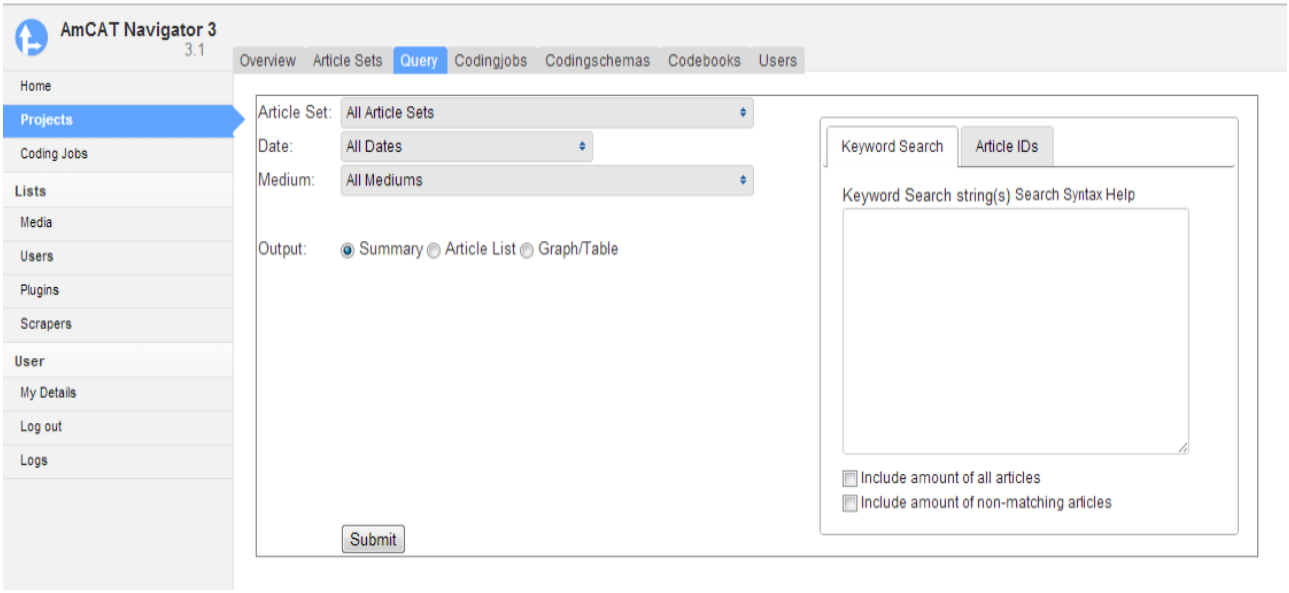


Figure 15. Query using keywords.

The above screenshot shows the input screen for keyword searches.

In the fields article set, date and medium, you can select one or multiple articlesets and media, as well as the time period for your search.

The default setting for ‘Article Set’ is ‘all article sets’, meaning all articlesets within the current project. By clicking ‘all article sets’ , a list is shown from which you can select specific articlesets. You can use the ‘filter’ field to search for articlesets using their name. ‘clear’ removes this search condition.

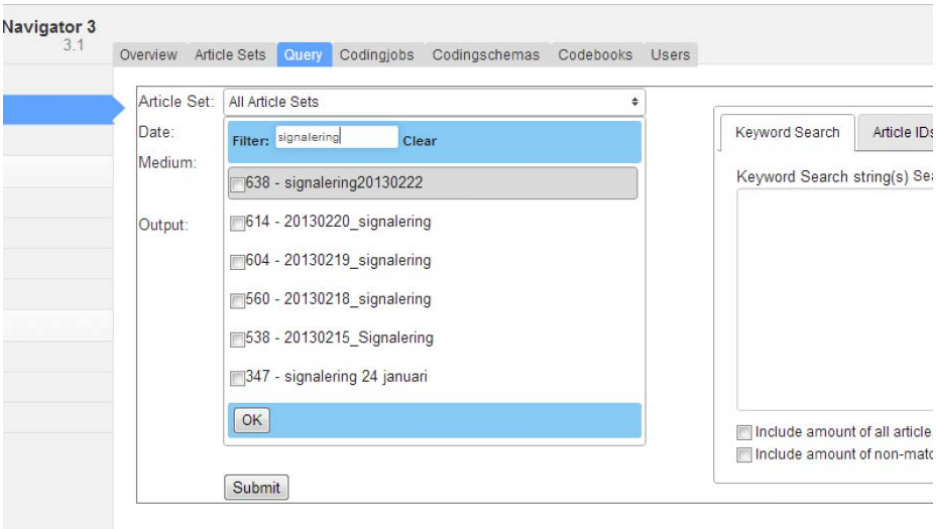


Figure 16. The 'filter' option for finding articlesets more efficiently.

The default for 'date' is the entire period for which the selected articlesets have articles included. By clicking 'date' you can specify the time period according to the following options: on, after, between en before. When you click on one of the fields below this option, a calendar appears from which you can select a specific date.

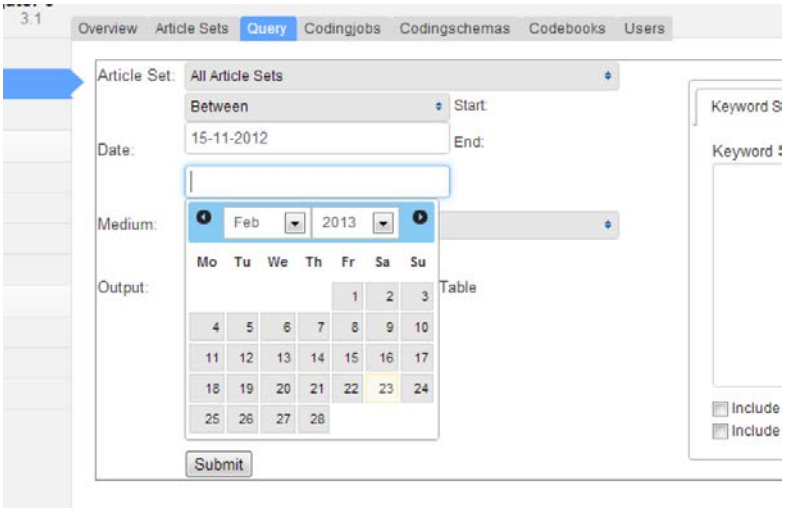


Figure 17. Delimiting your search by time period.

The option 'medium' allows you to select specific media in which to search. The default option is again 'all mediums'. As with the articlesets, this brings up a list from which you can select one or more media, as well as a 'filter' option to search for media within the list.

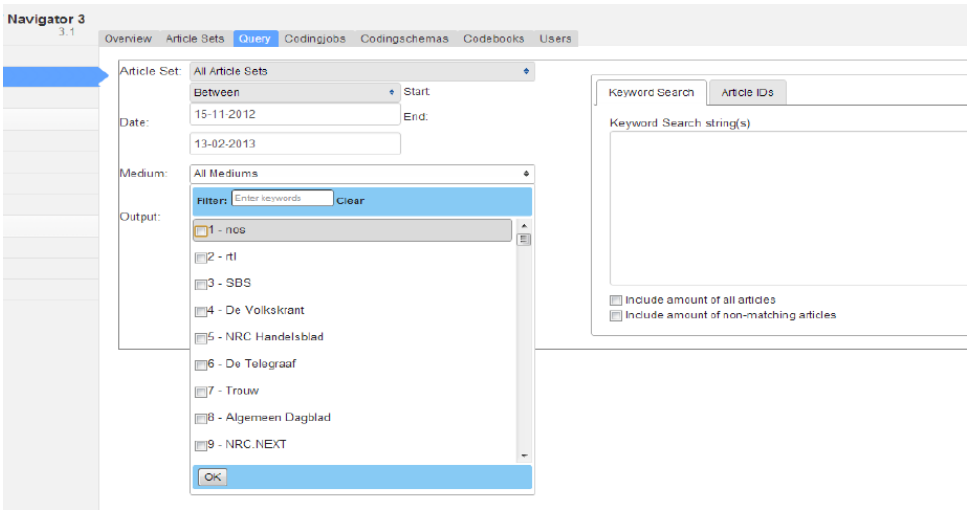


Figure 18. Delimiting your search by medium.

N.B.: A medium should be included in the articleset in order to search within medium (if, for example, you select an articleset that includes articles from The New York Times and The Washington Post, and from the list of media you select The Guardian, this will give you 0 results).

4.1 Query using searchstrings

You can search within search indexes using either searchstrings or article IDs. You can change the search option by clicking on the tabs above the search window (the red circle in the screenshot below).

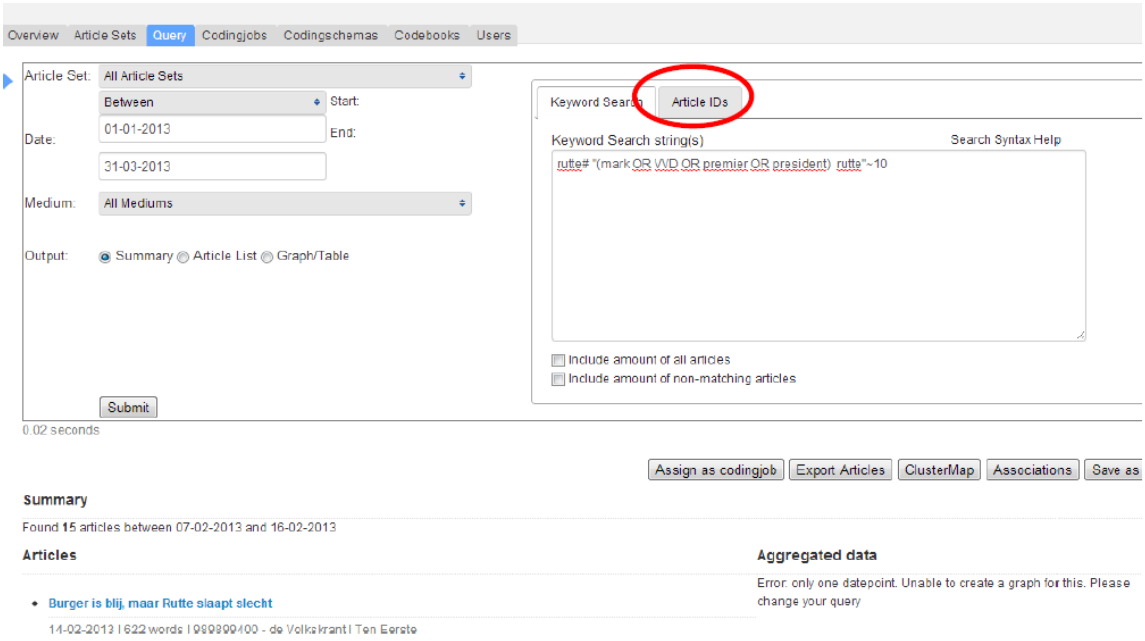


Figure 19. Entering a searchstring in AmCAT. Through the tabs above the search window you can switch between search using searchstrings or article IDs.

Enter your searchstring(s) in the search window and click on 'submit' (to the lower left of the screen) to search. You can enter multiple searchstrings at once by starting each searchstring on a new line.

The syntax for searchstrings in AmCAT is as follows:

- **Word1 word2; word1 OR word2** = articles that include either word1, word2 or both. Whenever there is a space between two words in a searchstring, AmCAT reads this as an OR term).
- **Word1 AND word2** = that include both word 1 and word2.
- **Word1 NOT word2** = articles that include word1, but not word2 staat.
- **"word1 word2"** = the exact phrase "word1 word2".
- "**word 1 word2"~5** word1 and word2 are at a maximum of five words distance of each other in the text. Whether word1 or word2 appears first does not matter. Instead of 5 you can enter any number higher than 1, and thus edit the maximal distance that the words can be apart.
- **( ) Brackets**: Using brackets you can make complex combinations of modifiers such as OR and NOT. Similar to mathematical formulae, the part of the searchstring that is between brackets comes first. For example: the searchstring '(word1 OR word2) NOT word3' retrieves all articles that include word1 of word2 but not word 3. The searchstring 'word1 OR (word2 NOT word3)' on the contrary retrieves all articles that include either word1 or word 2 but not word3. In this way, you can specify in much detail what exactly you want to search for.

- **“(word 1 OR word2) word3”~10** = either word1 or word2 is included in the article at a maximum distance of 10 words from word3. Instead of 10, you can enter any number higher than one.
- **? and * (wildcards).** A ? can stand for any (one) letter. For example, ‘gr?y’ retrieves both ‘grey’ and ‘gray’. A * stands for an unspecified letter or number of letters. For example, by searching for ‘minist*’ you will retrieve articles that include the word ‘minister’ as well as ‘ministry’. **N.B.** words in a searchstring cannot start with a wildcard.
- **Identifier#:** The name of the searchstring, ending at the #. This name, instead of the entire searchstring, appears in the output whenever AmCAT makes graphs or tables of search results. The word (s) in the identifier is not used as part of the searchstring. Example: rutte# “rutte (mark OR vvd)”~10 : here the name of the searchstring is ‘rutte’, the words that AmCAT searches for are “ rutte (mark OR vvd)~10.
- **headline:** = starting a searchstring with the modifier headline: restricts the search to article headlines.

Clicking ‘search syntax help’ on the upper right of the search window displays this explanation within AmCAT.

Further options:

Include amount of all articles: This option displays (within output tables) not only the number of retrieved articles, but also the amount of all articles in the selected articleset. At the moment of writing (April 2013) this option does not function yet.

Include amount of non-matching articles: This option displays (within output tables) not only the number of retrieved articles, but also the amount of articles that do not include the terms in the searchstring. At the moment of writing (April 2013) this option does not function yet.

4.2 Search using article IDs

You can select this option by clicking on the tab Article IDs above the search window.

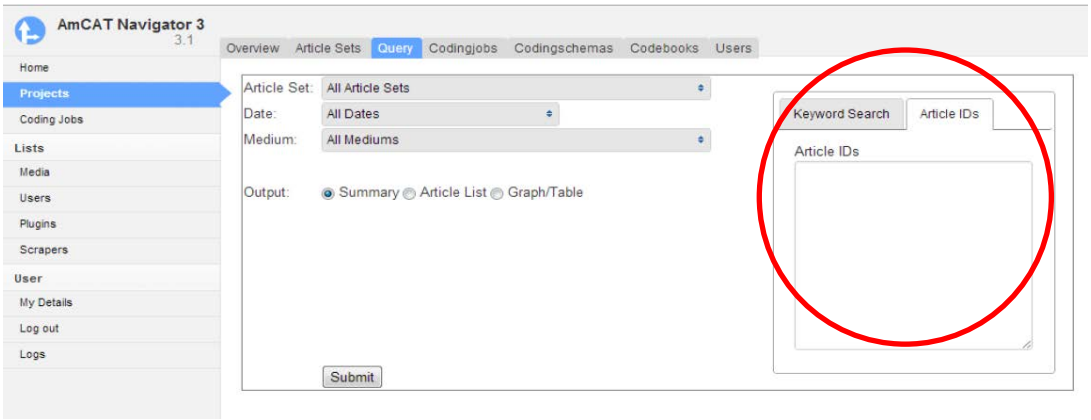


Figure 20. Seach using article IDs.

In the window in the red circle you can enter the article IDs of the articles you want to search for. This is especially useful when you want to export the texts and/or metadata of certain articles. You

can also use this procedure to make codingjobs or search indexes from those articles that you supplied the IDs for. To do so, select 'summary' as output type. Then, you can select 'assign as codingjob' and 'save as set' for creating codingjobs or search indexes, respectively.

4.3 Output options

AmCAT offers a number of possibilities for displaying and further editing the output of searches. In the 'query' screen you can choose from three ways of displaying the output: summary, article list and graph/table.

4.3.1 Summary

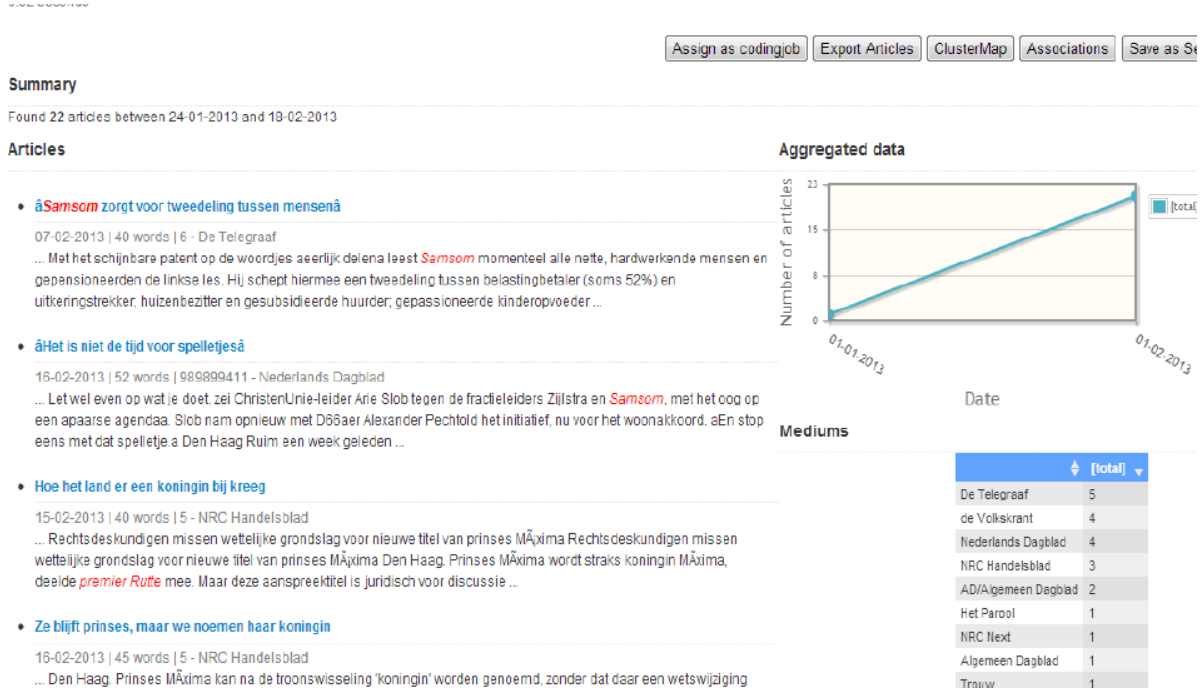


Figure 21. Search output displayed as summary.

The option summary summarizes the search result in three ways:

1. a list of all retrieved articles, consisting of each article's headline, publication date, number of words, and its medium. Also, for each article the first hit is displayed in context, meaning the sentence in which (part of) the searchstring was found, with the word(s) from the searchstring in red;
2. a graph showing the amount of articles found per day, of all entered searchstrings added together. For separate graphs per searchstring, you can select the output option graph/table;
3. a table that shows the amount of articles found per medium , of all entered searchstrings added together. Again, use the option graph/table if you want to see the results for each searchstring separately.

AmCAT gives the following options for further use of the search results:

Assign as codingjob: Here you can directly make a codingjob from the search results. Click on 'assign as codingjob' to go to a screen where you should enter the codingjob's name, the articleschema and

unitschema used, and the coder to which the job should be assigned. See chapter 5 for more information about creating codingjobs.

Export articles: Through this option you can export entire articles and/or a selection of metadata from AmCAT. When you click on 'export articles', a list appears from which you can select the metadata you want to export. Select 'article text', the last option on the list, to export the full text of the article. You can also specify the maximum number of articles you want to export as well as the file format. The options for formats are .csv (semicolon separated or comma separated), .xlsx, .sav and .html. The different types of metadata (for example publication date, medium, number of hits per article) are each in a different column in the output table. Each article is in a different row.

ClusterMap: This option is only possible for searches using two or more searchstrings. ClusterMap shows the degree of overlap between the searchstrings using a graph and/or a table. You can display these within AmCAT or export them in another format. The options for different formats are .csv (semicolon separated of comma separated), .xlsx en .sav. The output option inline html shows the clustermap within AmCAT, as shown below.

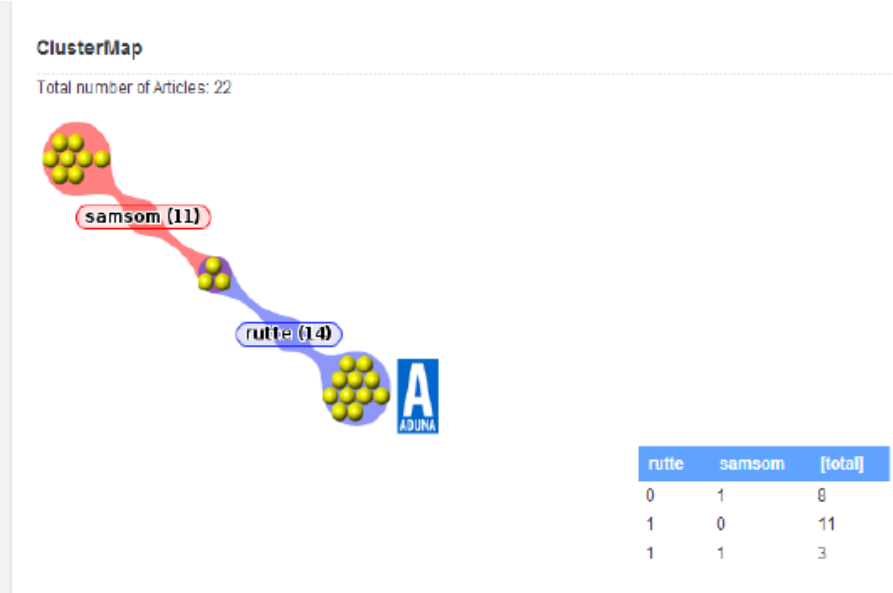


Figure 22. Clustermap of results for two searchstrings.

You can read the clustermap as follows: In the above graph, Rutte and Samsom are two search terms. The number behind each of the names show the amount of articles found for each of the search terms (11 for Samsom, 14 for Rutte). Every yellow dot stands for one article. All 11 dots encircled in blue are articles in which the search term Rutte appears, all 8 dots in the red area stand for articles in which the search term Samsom appears. The three dots in the middle mention both Samsom and Rutte, which is shown by the overlapping red and blue. By clicking on one of the yellow dots in the graph, you will go to the text of the article represented by that dot.

The table next to the graph will show the same information in numbers: there are 8 articles that do not (0) mention Rutte but do (1) mention Samsom, 11 articles that do (1) mention Rutte but do not (0) mention Samsom, and lastly 3 articles that mention both Rutte and Samsom (both 1).

When you choose not to display the clustermap within AmCAT, but export it directly in another format, you will only get the table as output, without the graph.

Associations: In a future update of AmCAT, this function will be included for measuring the associations between two or more search terms. Associations are measured using conditional probablities, meaning that whenever an article includes the first search term, the chance that the second search term is included as well. At the moment of writing (April 2013) this function does not work yet.

Save as set: Here you can save the search result as a new articleset (by entering a name for this articleset in the field 'articleset name') or adding them to an existing articleset (by selecting this set from the list in the field 'existingset').

4.3.2 Article list

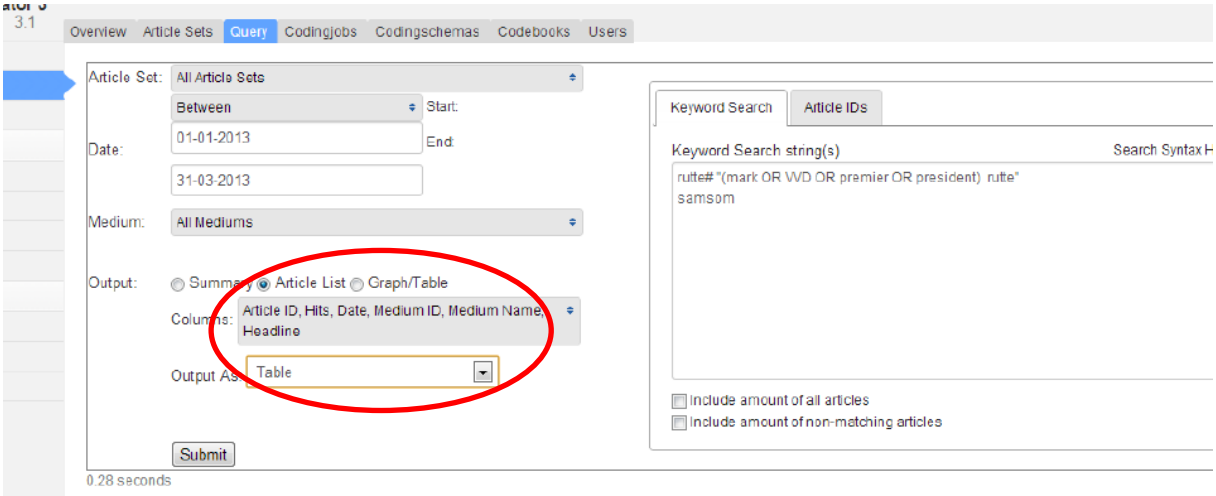


Figure 23. Options for displaying search results as a list.

This option uses either a table or a list as output type. For both types, you can select the metadata that you want to include. Article ID, date, medium ID, medium, and headline are displayed by default. You can select additional options such as number of hits per article, or remove metadata from your output list.

The output type 'list with snippets' is similar to the article list shown when selecting 'summary' as output option. The difference is that for 'article list', you can select the metadata to be included in the list. The options for further using the search results, such as displaying clustermaps or making codingjobs, are the same.

The output type 'table' gives the list of articles a more compact table format. You can not use any of the further options when you select table as output type.

4.3.3 Graph/table

This output option creates a graph or table for the search results. For graphs, when you choose to display your results per time interval, AmCAT will create a line graph. The graph type for results per medium is a bar graph. For tables too you can select whether you want to see the results per medium, per interval or both.

Figure 24. The graph/table output option.

The default setting is a medium per interval table, that is, a table with each medium in the columns ('y axis') and the time interval ('date interval') in the rows ('x axis'). The default setting for the values in the body of the table is the number of articles found. You can change all of these settings by selecting the preferred option from the list. For example, it is also possible to create a graph for the total number of hits per day, or a table for the number of articles per search term per medium.

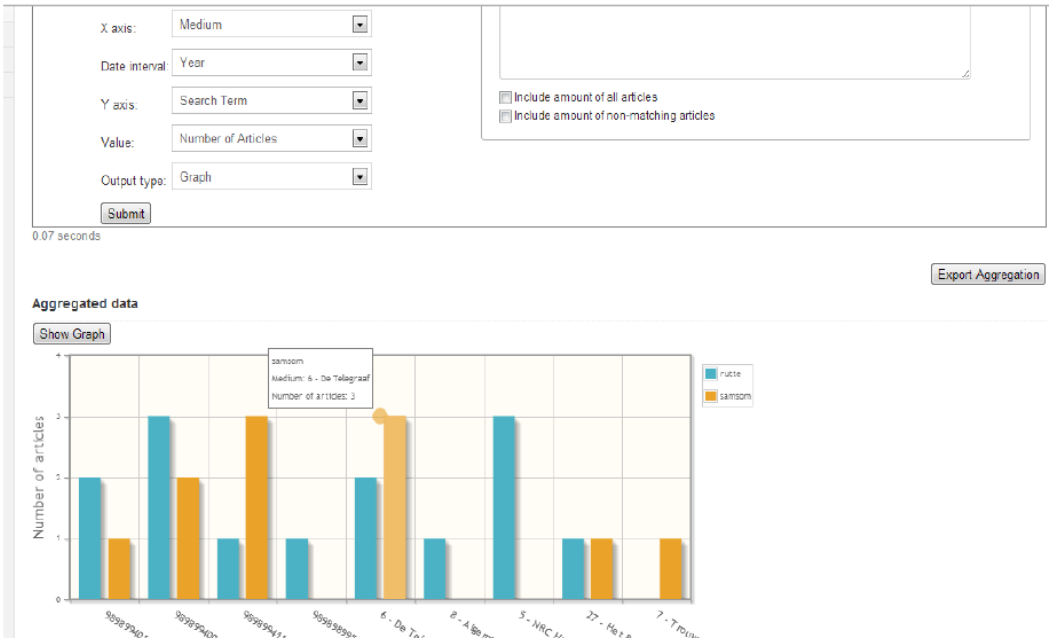


Figure 25. The type of graph AmCAT creates depends on the way you choose to categorize your search results: bar graphs are the graph type for results per medium.

You can hover the cursor over a graph to display detailed information for each data point. For example in the above bar graph, the number of articles found, the name of the search string and the medium in which the articles were found are shown when hovering the cursor over one of the bars.

Export aggregation: this exports the data for the table or graph in an external file format. Format options are.csv (semicolon separated or comma separated), .xlsx, .sav and .html. For graphs, the data are exported as a table, which you can use to recreate the graph yourself, for example in Excel.

4.4 Possible errors when using searchstrings

- *Error running webscript: HTTP code=400, reason=This query could not be parsed*: the searchstring has an error in its syntax, for example a bracket too few or too many.
- *Error: only one datepoint. Unable to create a graph for this. Please change your query*: in this case, AmCAT tries to create a graph for search results that are all within the same date interval. If you still want to create a graph, you can change the interval for the graph or change the search period.

5 Codingjobs

Go to the tab ‘codingjobs’ to get an overview of all codingjobs within the current project. By clicking on a codingjob in the table, you can change its details, such as the coder to whom the codingjob is assigned. Click on ‘New codingjob’ above the table for creating a new codingjob. This will be explained in paragraph 4.1.

N.B.: The tab Codingjobs is not the same as the option ‘Coding Jobs’ from the menu to the left of the screen in AmCAT. The tab ‘Codingjobs’ gives an overview of all codingjobs in the project as well as the option to change their details, while ‘Coding Jobs’ from the menu to the left leads to the AmCAT annotator, where you can code the codingjobs assigned to you.

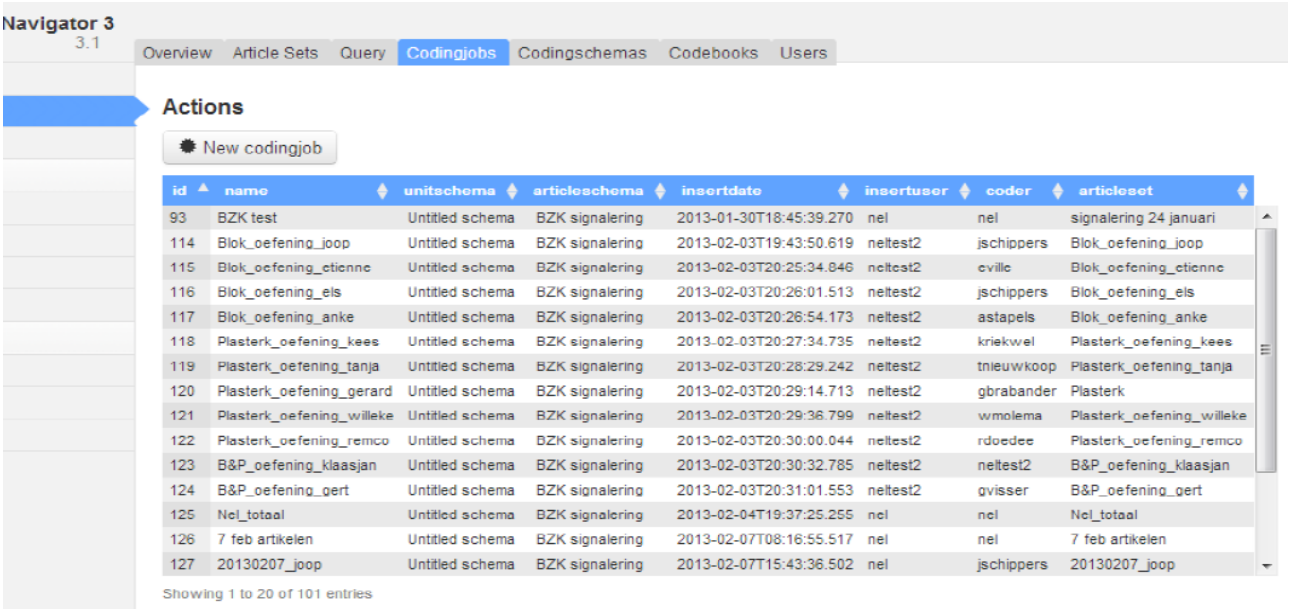


Figure 26. Overview of codingjobs within a project.

The table in this screen gives the following details for each codingjob:

- **Id:** the unique ID number for each codingjob
- **Name:** the name given to the codingjob when it was created
- **Unitschema:** the coding scheme for coding on the unit level, used for this codingjob
- **Articleschema:** the coding scheme for coding on the article level, used for this codingjob
- **Insertdate**: the date and time that the codingjob was created
- **Insertuser**: the user who created the codingjob
- **Coder**: the user to whom the codingjob is assigned
- **Articleset**: the articleset that contains the articles that are used in the codingjob.

Clicking on one of the codingjobs leads to a screen that gives a detailed overview of that codingjob.

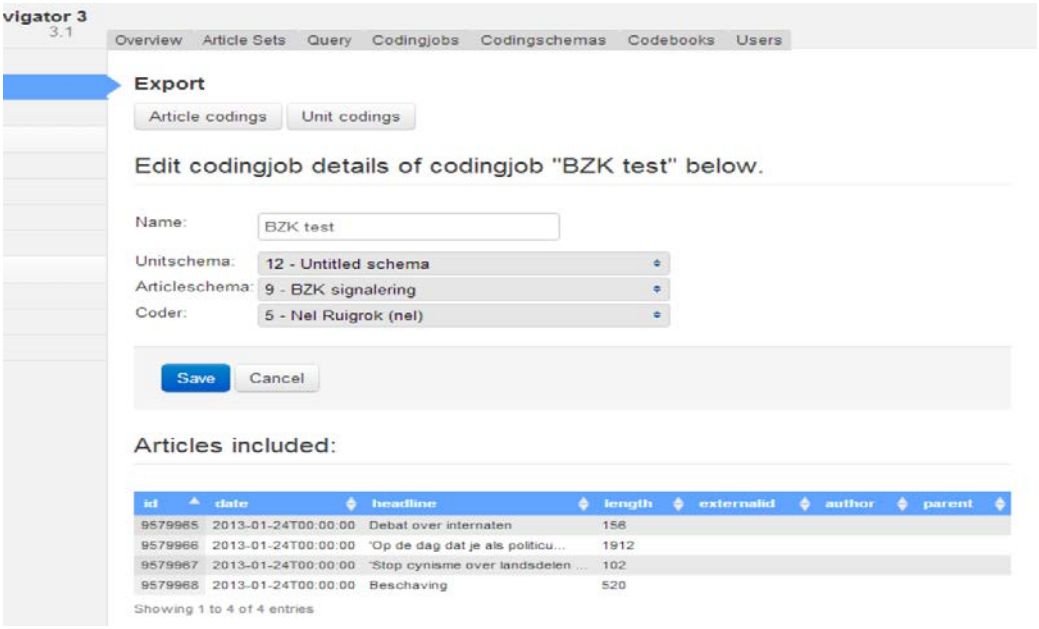


Figure 27. Detailed overview of a codingjob.

Export: An important function in the detailed overview of a codingjob is the option to export that codingjob's codings. Codings on the article level and on the unit level are exported separately, through clicking on the buttons 'article codings' and ' unit codings', respectively. If you want to create one dataset for both article and unit codings, you will have to merge the article codings dataset and the unit codings dataset in for example SPSS or Excel. At the time of writing (April 2013), .csv is the only output format available for exporting codings. More formats will be added soon.

Edit codingjob details: Here you can change the following details for each codingjob:

- **name**
- **unitschema**
- **articleschema**
- **coder**

By clicking 'save', the changes are applied.

Articles included: this list shows all articles included in the codingjob, showing their ID, headline and other metadata. You cannot make changes to this list. By clicking on an article in the list, you can see the entire text as well as the metadata.

N.B. You cannot add articles to a codingjob directly. If you want to add more articles to a codingjob, you can do so by adding them to the articleset that is used as a basis for the codingjob. Any changes made to this articleset are automatically applied to the codingjob as well.

5.1 Creating new codingjobs

AmCAT offers two ways to create new codingjobs. The first way creates a codingjob from an existing articleset. The second way is useful when you want to make codingjobs that encompass more than one articleset, or that are based on a selection of articles from one or more articlesets.

Creating new codingjobs, method 1: Go to the tab 'codingjobs' and click on the button 'new codingjob' . This will bring up the following screen:

Figure 28. Creating new codingjobs**.**

Here you name the codingjob and select its unitschema, articleschema, coder and articleset (that includes the articles to be coded). Click on 'save codingjob details' to create the new codingjob. It will then appear in the AmCAT Annotator for the appointed coder.

Creating new codingjobs, method 2: When you want to create a new codingjob that does not exactly overlap with an existing articleset, you can do so through creating a codingjob with a query. First, go to the tab 'query'. If you know beforehand which articles you want to include in the codingjob, you can enter the IDs of those articles in the search window (for a search using article IDs). Choose summary as output option, and click on 'assign as codingjob' on the upper right above the search result. Alternatively, you can use a searchstring to select articles from multiple articlesets and directly create a codingjob from the search results.

When you create codingjobs in this way, AmCAT automatically saves the articles included in the job as a new articleset. The articles are copied from their old articleset into the new one – they do not disappear from their original articleset.

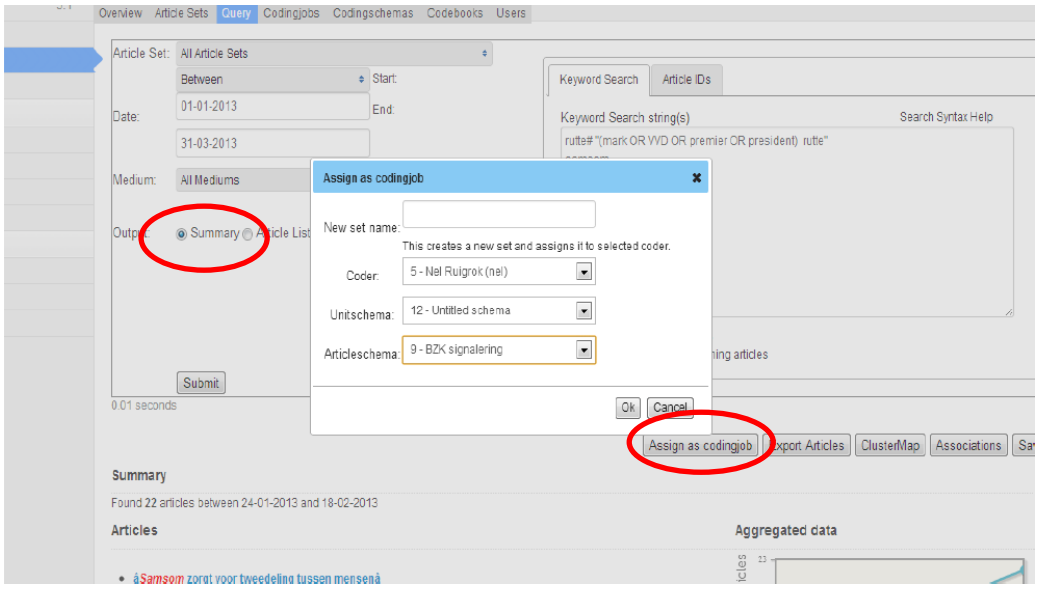
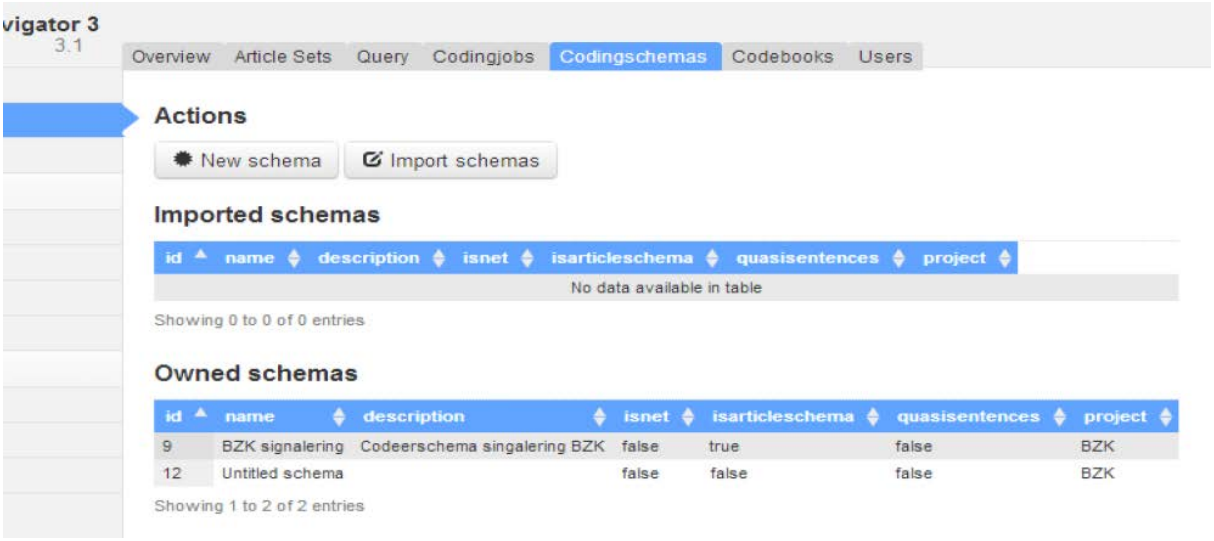


Figure 29. Creating new codingjobs through queries.

N.B. When you want to assign the same codingjob to multiple coders, an efficient way is to make a list from the IDs of all articles that should be included, and make new codingjobs from this same list for each coder.

6 Codingschemas

A codingschema (coding scheme) consists of a number of variables plus their answer categories. Codings on the article level and on the sentence level use separate schemas, both of which have to be included in each codingjob. In the tab 'codingschemas' you can create a new codingschema, to which you can add codebooks (a codebook is the list of possible values for each variable, used mostly for norminal and ordinal variables). If you want to use codebooks, but have not made these yet, you should do so before creating the codingschema (see paragraph 7.2). If you want to use existing codebooks and/or existing codingschemas, you can import these from other projects and edit them if necessary.



Figuur 30. Overview of codingschemas.

Under the heading 'Actions' you will find the button 'new schema' for making a new codingschema, explained in paragraph 6.2. The button 'import schemas' allows you to import codebooks or codingschemas from another project (discussed in paragraph 6.3). Afterwards, these codingschemas will appear in the list 'imported schemas'. 'Owned schemas' shows the codingschemas edited in the current project.

The overview of codingschemas shows the following details for each codingschema:

- **id**: the unique number of each codingschema;
- **name:** the codingschema's name (tip: include 'article' in the name for articleschemas and 'unit' for unitschemas);
- **description**: the description added for each codingschema;
- **isnet**: 'true' means that the codingschema is a unitschema, 'false' means that it is not;
- **isarticleschema**: 'true' means that the codingschema is an articleschema, 'false' means that it is not;
- **quasisentences**: at the moment of writing (April 2013) the option for coding on the quasi-sentence level (that is, a level in between articles and units)does not yet exist. This will be added in later versions of AmCAT.
- **project**: the project under which the codingschema is categorized.

By clicking on a codingschema in the list, you will see a more detailed overview of that codingschema.

6.1 Editing an existing codingschema

The screen showing the detailed overview of the codingschema also has options for editing, copying or deleting the schema. The table on the lower half of the screen shows all variables included in the schema, as well as the assigned codebooks.

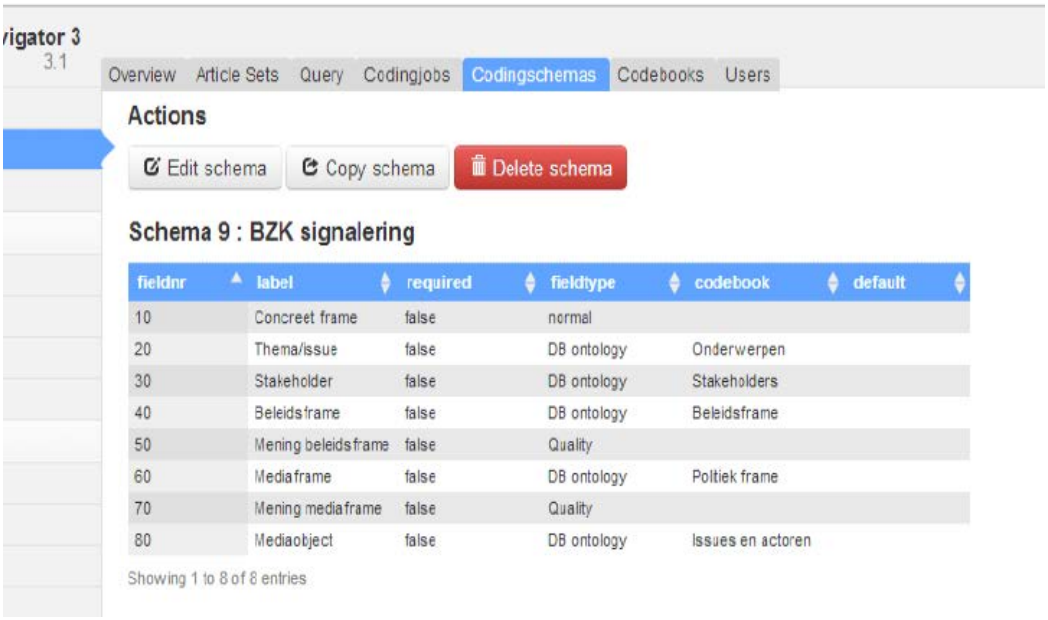


Figure 31. Detail overview of a codingschema.

The table lists the following characteristics of each variable:

- **Fieldnr**.: the number for each variable. This stays the same even if you change the order of variables in the codingschema.
- **Label:** this is the label that coders see when they code this variable in the AmCAT Annotator. For example: an articleschema consists of a number of questions. 'Does the article mention the private lives of politicians?' is one of these questions, and is used as the variable label in the codingschema.
- **Required:** 'true' means that a value must be entered for this variable (if not, AmCAT will not save the codings), 'false' means this is not required. At the time of writing, all variables are shown as being 'required', but in practice AmCAT also saves codings when no values are entered.
- **Fieldtype**: the variable type. AmCAT uses the following types of variables (fields):
    - ***Text***: a field where you can type in text freely.
    - ***Number***: a field where you can type in numbers freely.
    - ***Codebook***: a field where you can choose a value from a number of options (that is, a codebook).
    - ***Yes/No***: a field where you can enter either yes or no.
    - ***Quality:*** a field where you can select a number from a list, for measuring the sentiment in a sentence or article. The values are -1;-0,5; 0; 0,5 and 1.

- **Codebook**: the codebook, meaning the set of answer categories, linked to that variable (only used when the fieldtype is 'codebook').
- **Default**: the default value for that variable. If this is left blank, no default value is set.

You can add changes to the codingschema by clicking 'edit details' on the top of the screen.

This will lead you to to the following overview:

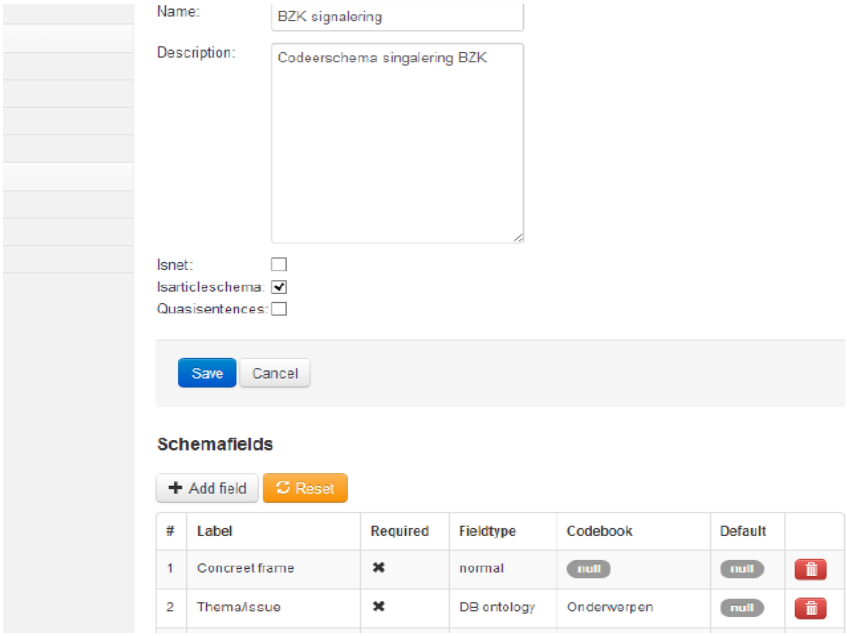


Figure 32. Editing codingschemas.

Here, you can change a numberof details for each codingschema , starting with its name, its description and its type (unitschema or articleschema). Under the heading 'Schemafields' you can make changes to the variables included in the codingschema, such as changing their order. You can do so by clicking on the variable you want to move, and dragging it to its new position in the codingschema.

Using the button 'add field' you can add a new variable to the codingschema. The new variable appears in the bottom row of the table. You can make changes to the fields in the table by double-clicking on the field you want to change . Doing so, you can edit the label, the required-option, the fieldtype, the codebook and the default for each variable. If you want to stop editing the codingschema, you can click outside the table or press enter.

You can remove a variable from the schema by clicking on the garbage bin icon for the correspoinding row, in the rightmost column. 'Reset' cancels all changes to the codingschema (that is, if those changes are not yet saved). You make changes definitive by clicking 'save'. 'Cancel' (instead of 'save', not afterwards) cancels all changes to the codingschema and returns you to the overview.

N.B. When you make a change to a codingschema or a codebook, it is directly applied to any codingjobs using that codingschema or codebook. Coders working on these codingjobs in the AmCAT Annotator sometimes need to refresh the page (F5) to see the changes.

6.2 Creating a new codingschema

When you want to make a new codingschema, you click on ‘new schema’ under the tab codingschemas. Make sure that you have already created or imported all necessary codebooks (see paragraph 7.2 and 7.3) if you want to use nominal or ordinal variables.

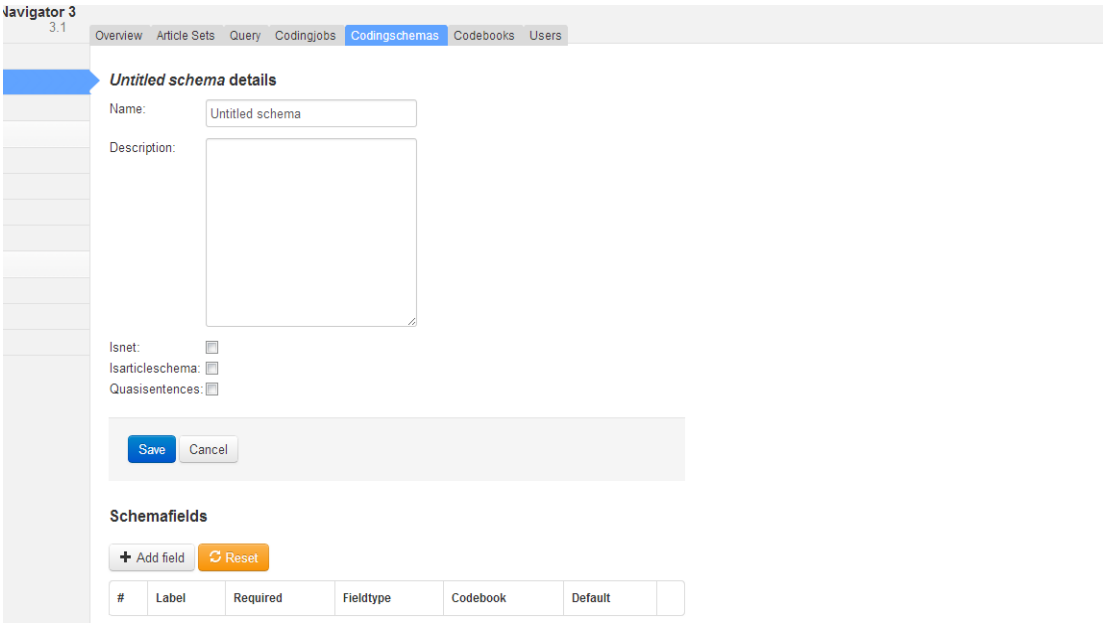


Figure 33. Creating a new codingschema .

You start by giving the new schema a name and a description. Then, you specify whether the codingschema will be used for codings on the unit level (you select ‘isnet’) or on the article level (you select ‘isarticleschema’). ‘Quasisentences’ will be added in a future AmCAT update, for articles on an intermediate level.

An articleschema consists of a list of variables, often in question form, that can be filled out by coders in the AmCAT Annotator. A unitschema consists of a table, in which each variable is assigned to a column, and each row to a unit (that is, a sentence or part of a sentence). Coders can add extra rows if there are additional units to be coded. Variables can only be added or removed in the AmCAT Navigator, both for articleschemas and unitschemas.

Under the heading ‘schemafields’ you can specify the variables to be included in the codingschema. First, you use the button ‘add field’ to add an empty variable, which you then specify further by double-clicking on each field you want to change. For each variable, you specify the label, which is what the coder sees in the AmCAT Annotator when coding that variable (for example ‘Does the article contain any war metaphors when describing politics?’). You can also specify whether a value must be entered for the variable or not (‘required’). For each variable, you specify the type (‘field type’, see page 30) and if necessary the codebook that must be linked to that variable. Lastly, you can enter a default value (‘default’) for each variable. If you leave the default as ‘null’, the variable will have an empty field as the default value. You can stop editing the codingschema by clicking outside the table or pressing enter.

Click on the ‘save’ button on the bottom of the page to save the new codingschema. **N.B.** AmCAT returns you to the overview of the codingschema when changes are saved correctly. Click on ‘save’ again if you stay in the edit screen, as this means that the codingschema is not saved yet!

6.3 Importing codingschemas

In order to import one or more codebooks or codingschemas into the current project, you click on ‘import schemas’ in the tab ‘codingschemas’. This brings up the following screen:

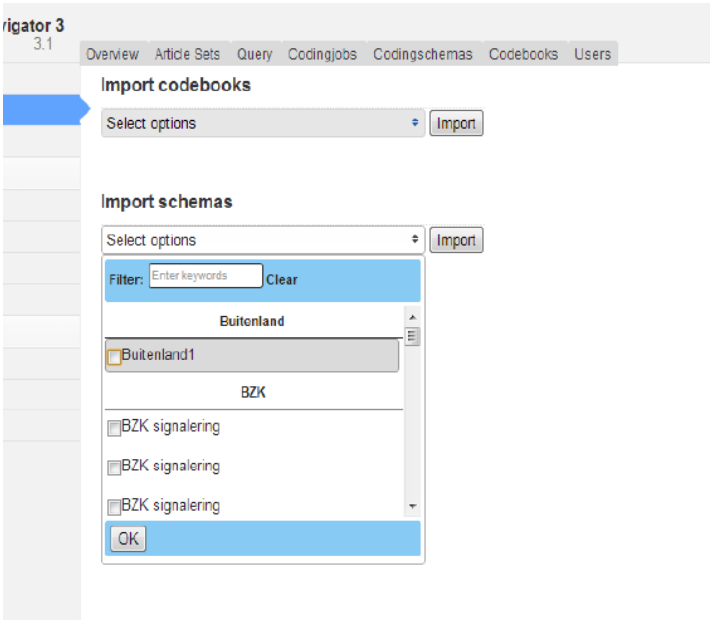


Figure 34. Importing codingschemas.

Here, you can select either codebooks or codingschemas from their respective lists. The ‘filter’ field allows you to search for codingschemas or codebooks by their name. When you have selected one or more codingschemas/codebooks, click on ‘import’ to import them into the current project. Imported codebooks can then be assigned to a codingschema in the current project. Imported codingschemas can either be edited or used in the current project as they are. When you import a codingschema, the codebooks used in that codingschema are only imported as part of that codingschema. They should be imported separately if you want to use them for other codingschemas in the current project.

You cannot edit imported codingschemas directly: AmCAT will ask you to copy the codingschema to the current project first. Click on ‘yes’ , re-name the codingschema and click on ‘copy’ to copy it to the current project. Now, you can edit the codingschema as described in paragraph 6.1. The codingschema will be shown in the table ‘owned schemas’ in the codingschema overview. You can edit imported codebooks in the tab ‘codebooks’, as explained in the next chapter.

7 Codebooks

A codebook in AmCAT is a set of answer categories that can be linked to one or more variables. To do so, variables should have 'codebook' as their field type (you can edit this as described in paragraph 6.1). The number of answer categories (codes) can be anywhere from two to a few thousand. A future update in AmCAT will make it possible for users to upload large codebooks in AmCAT directly. At the moment of writing, this is only possible through sending the codebook to Wouter, who will read it into AmCAT. Here, we will only describe the procedure for creating smaller codebooks in AmCAT manually.

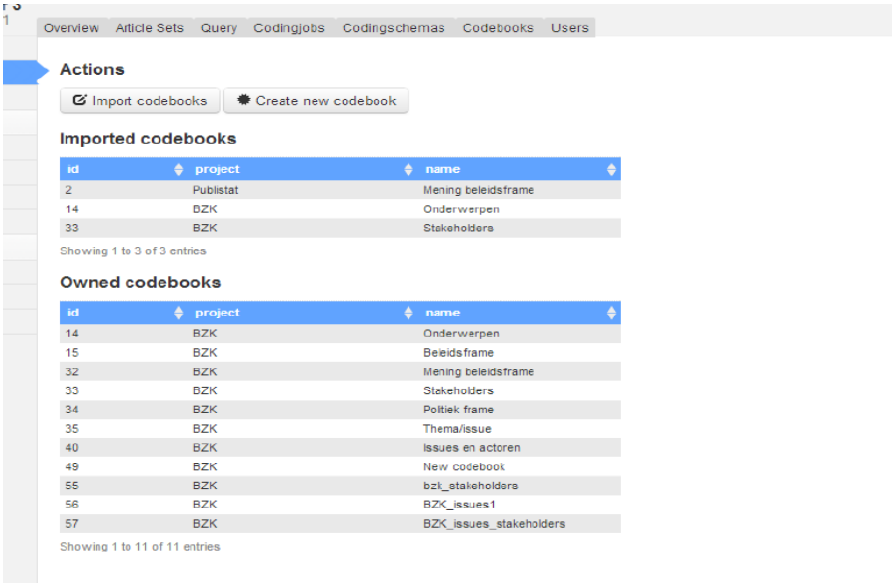


Figure 35. Overview of codebooks.

The tab 'codebooks' leads you to the screen shown above. The top of the screen shows two buttons for respectively importing a codebook from another project (explained in paragraph 7.3) and creating a new codebook from scratch (explained in paragraph 7.2). The tables 'imported codebooks' and 'owned codebooks' show all codebooks used in the current project.

Again, imported codebooks are created in and imported from another project, while owned codebooks are created in the current project. An imported codebook can be edited directly in the current project, as explained below. For each codebook, the table shows its ID (the unique number for each codebook within AmCAT), the project that the codebook was originally created in, as well as the name of the codebook. When you click on a codebook in the table, you will go directly to the codebook editor, where you can make changes to the codebook.

7.1 Editing codebooks

The codebook editor shows a list of all codes in the codebook. Using the search function on top of the screen, you can search for specific codes within the codebook. The screenshot below shows the codebook editor.

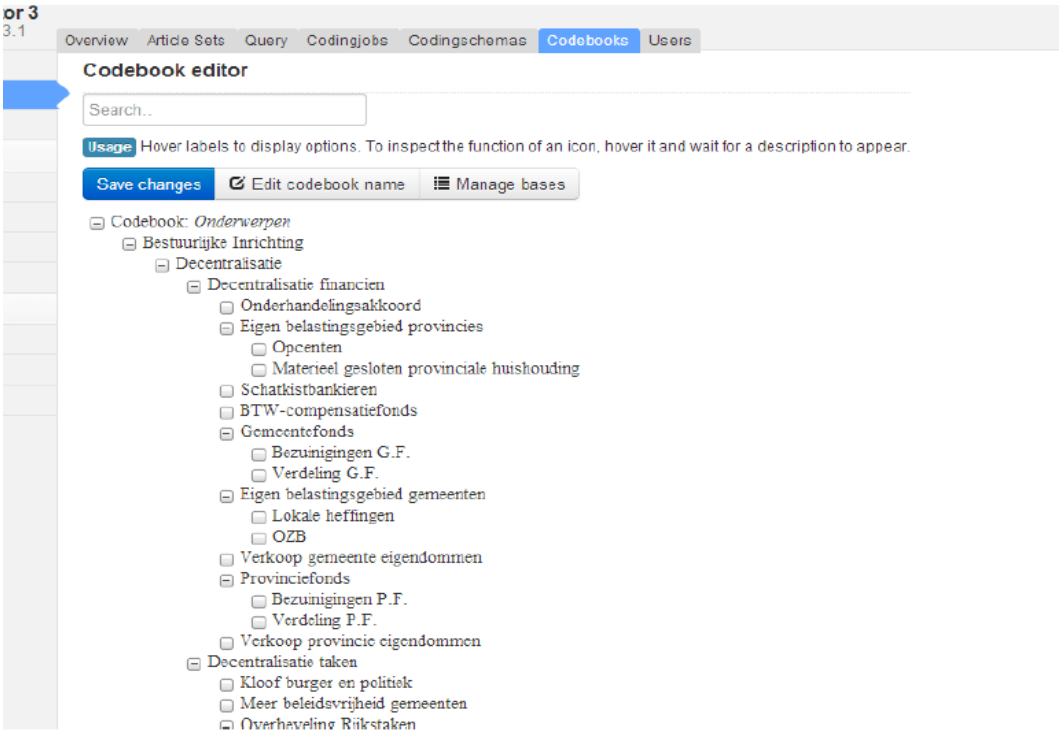


Figure 36. The codebook editor.

Through the button ‘edit codebook name’ above the codebook, you can change the name of the codebook. ‘ Manage bases’ does not function yet at the moment of writing. ‘Save changes’ saves any changes made to the codebook.

Below these three buttons, the codebook itself is displayed. A simple codebook consists of a short list of answer categories (codes). More complicated codebooks have a hierarchical structure: anwer categories (parents) are further divided into subcategories (children). You can toggle the display of a parent code’s children on or off by clicking on the small square to the right of the code.

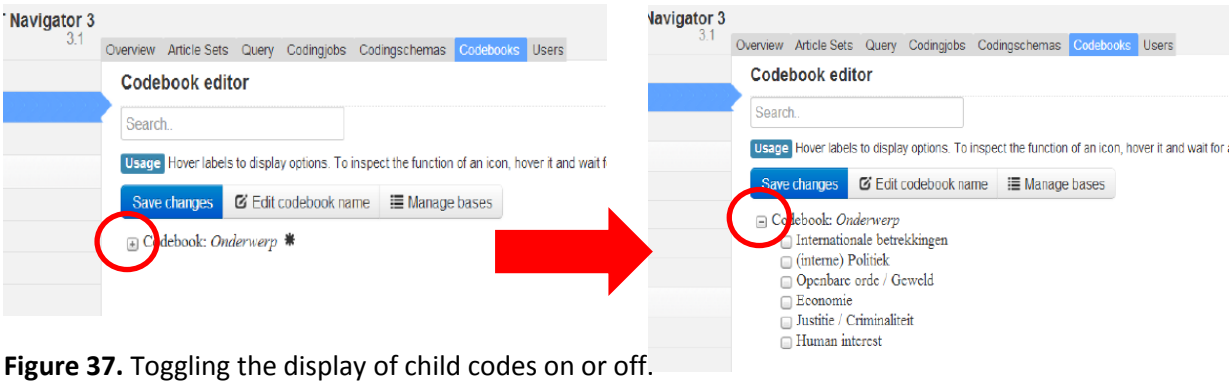


Figure 37. Toggling the display of child codes on or off.

If you want to make changes to the codebook, you can hover the cursor over the code you want to change. This brings up a number of symbols next to the code’s name. Each of these symbols is explained below.

New child: This creates a new code, that falls below the current code in the hierarchy. This means that if you want to create a simple codebook without an hierarchical structure, you should add each new code by clicking on the 'new child' symbol **next to the codebook name**. Clicking on the symbol brings up the following screen:

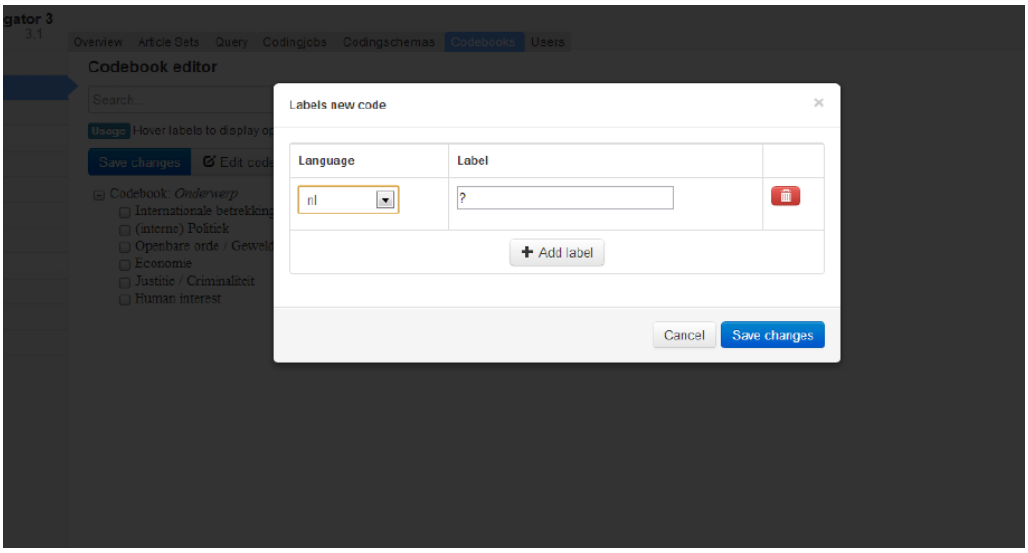
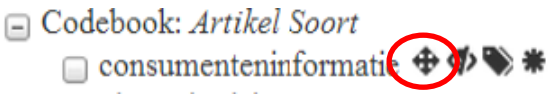


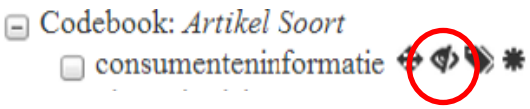
Figure 38. Adding a new code.

For each new code, you select the language, and label the code (the label is the name of the code that coders will see in the AmCAT Annotator. It is also displayed as the code's name in the codebook editor). Use 'add label' to add additional labels in another language. Which label coders will see depends on their language setting in AmCAT. You can change this setting in 'my details' from the menu to the left of the screen.

When you click on the red garbage bin icon, you remove the label in the corresponding row. 'Save changes' adds the code to the codebook, while 'cancel' brings you back to the codebook without making any changes.



Move code: By clicking on this symbol, you can move the code within the codebook. First, click on the symbol and then click on the code that should serve as the parent code for the code you want to move. You can only move codes between different hierarchical levels.



Hide code: This makes the code invisible in the AmCAT Annotator. Coders cannot see or use hidden codes. The codebook editor displays a line through hidden codes, as shown below:

Figure 39. A hidden code.

You can click on the eye-icon next to the code to unhide it again.

Show label: This brings up a screen where you can change the label of the code, and/or add a new label in a different language using the 'add label' button. Click on 'save changes' to save the changes to the code's label(s). 'Cancel' returns you to the codebook editor without making any changes.

7.2 Creating a new codebook

Click on 'new codebook' in the tab codebooks to create a new codebook. This brings you to the following screen, showing an empty codebook:

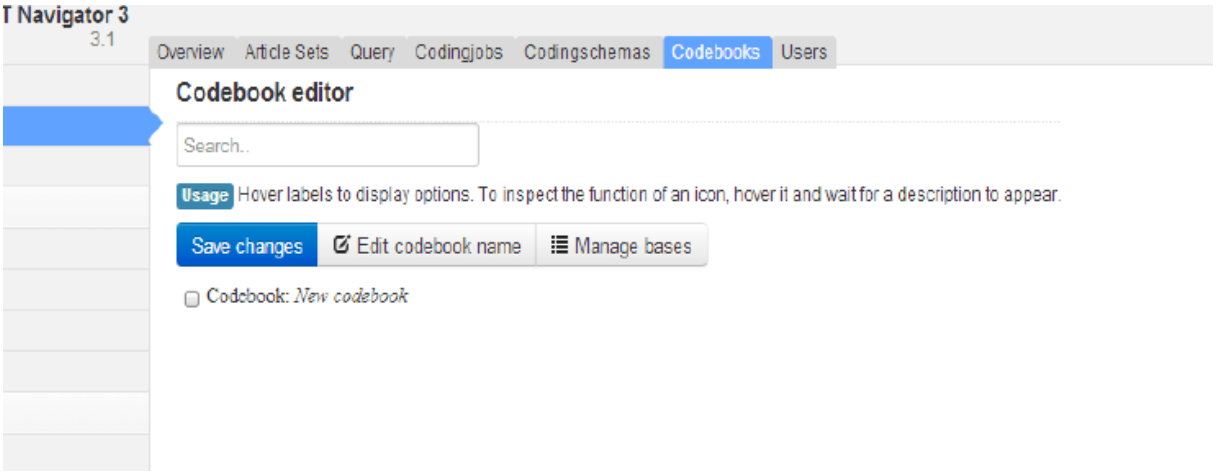


Figure 40. A new codebook.

First, you name de codebook by clicking 'edit codebook name'. You can recognize the codebook by this name when you add it to a codingschema. You can add new codes (children) through hovering

with the cursor over the codebook name, and clicking on the star icon that appears (see paragraph 6.1).

N.B. Mind the hierarchical structure of codebooks when adding new codes. Codes that are supposed to be on the same hierarchical level, should be added as child codes of the same parent code, **not** as child codes of one another.

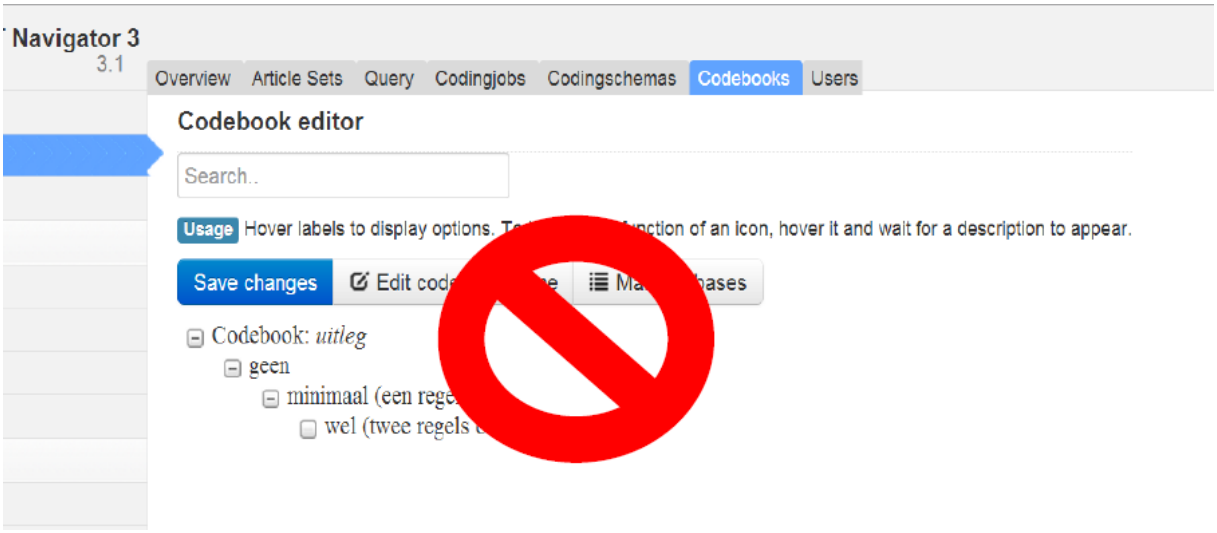


Figure 41. Wrong: these codes ('no', 'minimal – one sentence' and 'yes-two sentences or more') are supposed to have the same level in the hierarchy. However, each new code is added as a child of the previous code, thus placing each code on a lower hierarchical level. This might lead to problems when analysing the data.

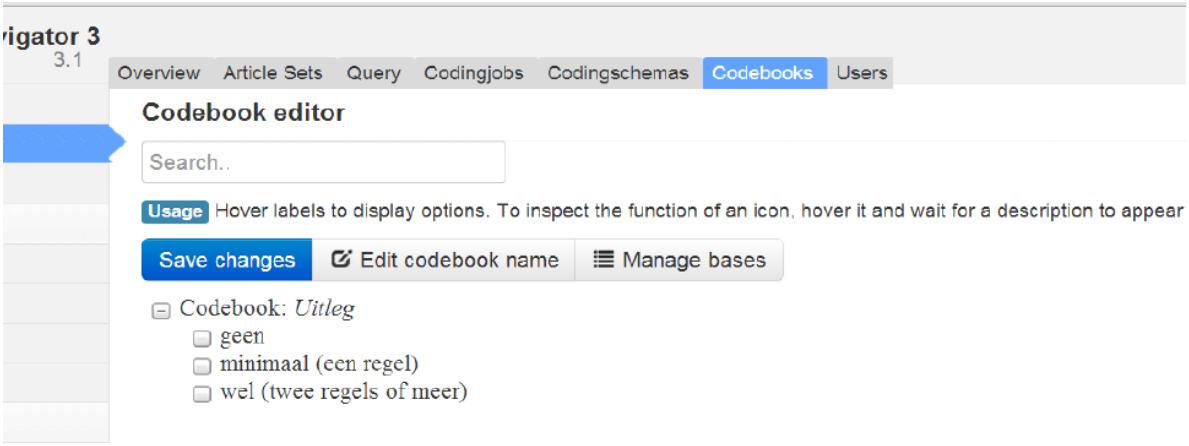


Figure 42. Good: Here, each code is added as a child of the same parent-code (the name of the codebook itself). This places them on the same level in the hierarchy.

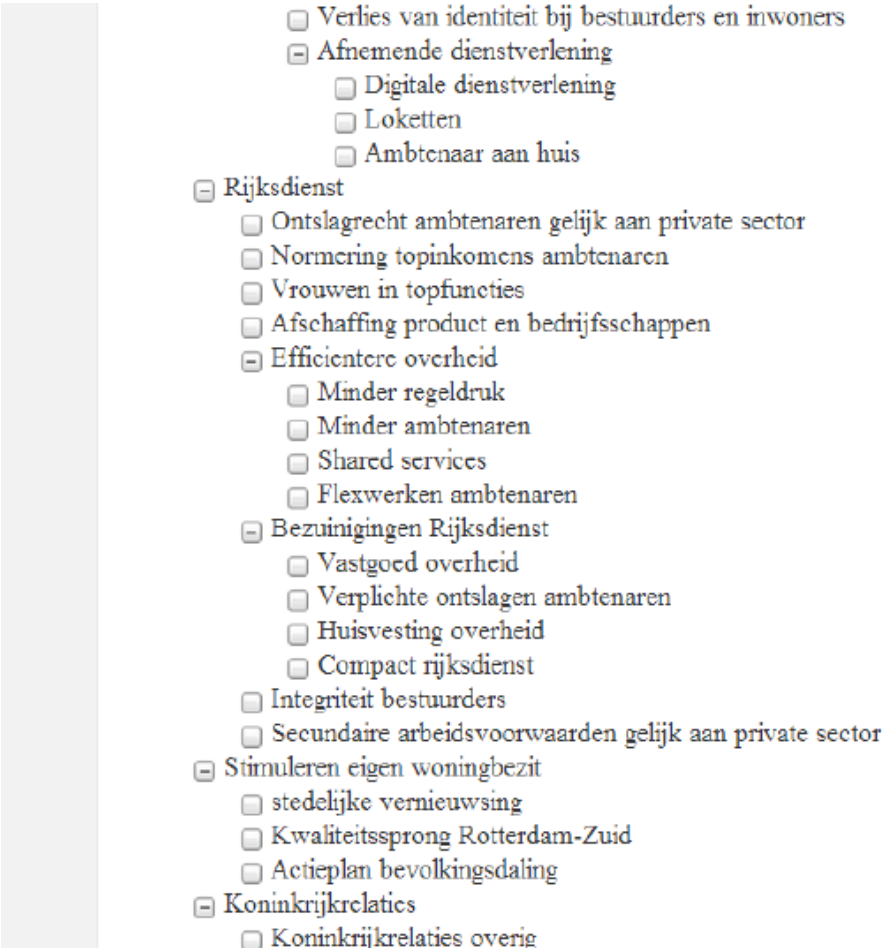


Figure 43. The correct use of the hierarchy in the codebook: each level corresponds to a different level on which the data can be aggregated.

The hierarchical structure of codebooks is not only used for ordening long codebooks, but they also have a function in the analysis of codings. Each level in the hierarchy corresponds to a different variable when extracting the data, which makes it easier to analyse coding data on various levels of aggregation. Coderings that used codes on the child-level are automatically assigned the parent-code's value for the variable corresponding to the higher level of aggregation. For the variable corresponding to the level of child codes, parent codes keep their original value.

As the export coding data function is still under construction, this is not guaranteed to work flawlessly yet.

7.3 Importing a codebook

Using the button 'import' in the tab codebooks, you can import a codebook from another project in order to edit and use it in the current project.

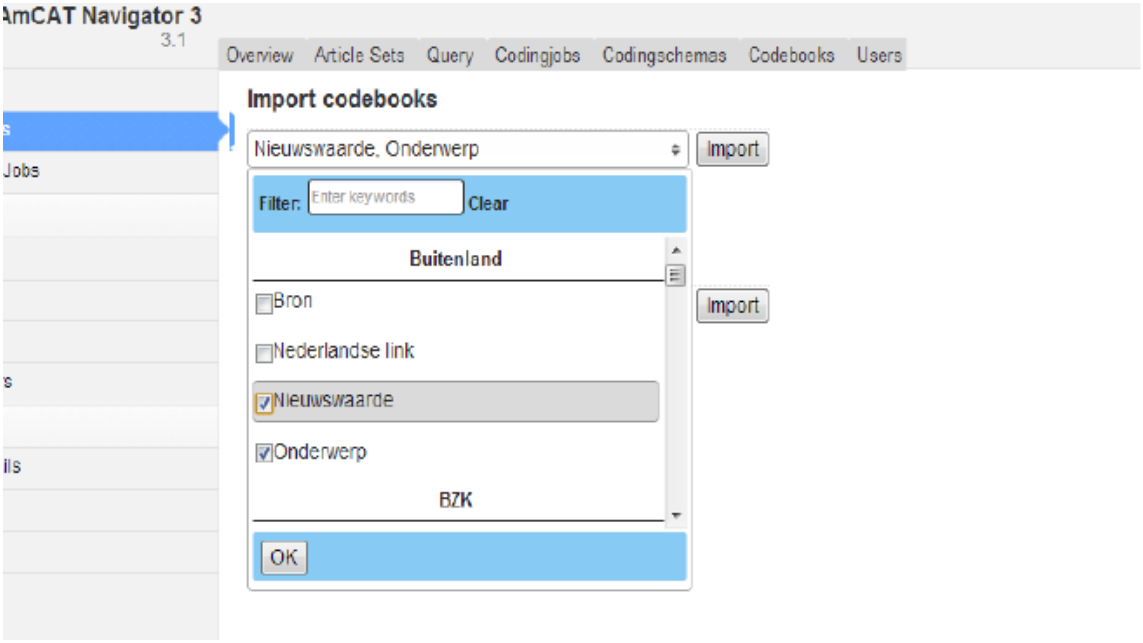


Figure 44. Importing codebooks.

This screen is similar to the one described in paragraph 6.3. Search for codebooks using the filter option, select the codebook(s) from the list and click 'import'. You can edit the codebook in the current project as described in paragraph 7.1, or add it to a codingschema as described in paragraph 6.2.

8 Users

The tab ‘users’ shows an overview of all AmCAT users who are added to the current project. The second column in the table shows the role in the project for each of these users. Roles determine the degree of acces that users have in the current project. You can change the role for each user by clicking on the corresponding row in the table.

The access options for the different rows are as follows:

- **Admin:** users with the admin role have complete access in the entire project. Only users with the admin role can add other users to a project or delete the project.
- **Read/write:** users with the read/write role can see the entire project, including the article texts. They can also create or delete articlesets and codingjobs within the project.
- **Reader:** users with the reader role can see the entire project, including the article texts. They cannot make any changes to the project.
- **Metareader:** users with the metareader role can see the project resources such as codebooks and articleset details, but not the texts of the articles included in the project. They cannot make any changes to the project.

Aside from users who are added to the project, guest users (who have acces to projects to which they are not added through selecting them from the list ‘all projects’) can also have these roles. ‘Reader’ is the default role for guest users. You can change this role by clicking ‘edit details’ in the project overview. **If your project contains any confidential texts, it is advised to make ‘metareader’ the guest role.**

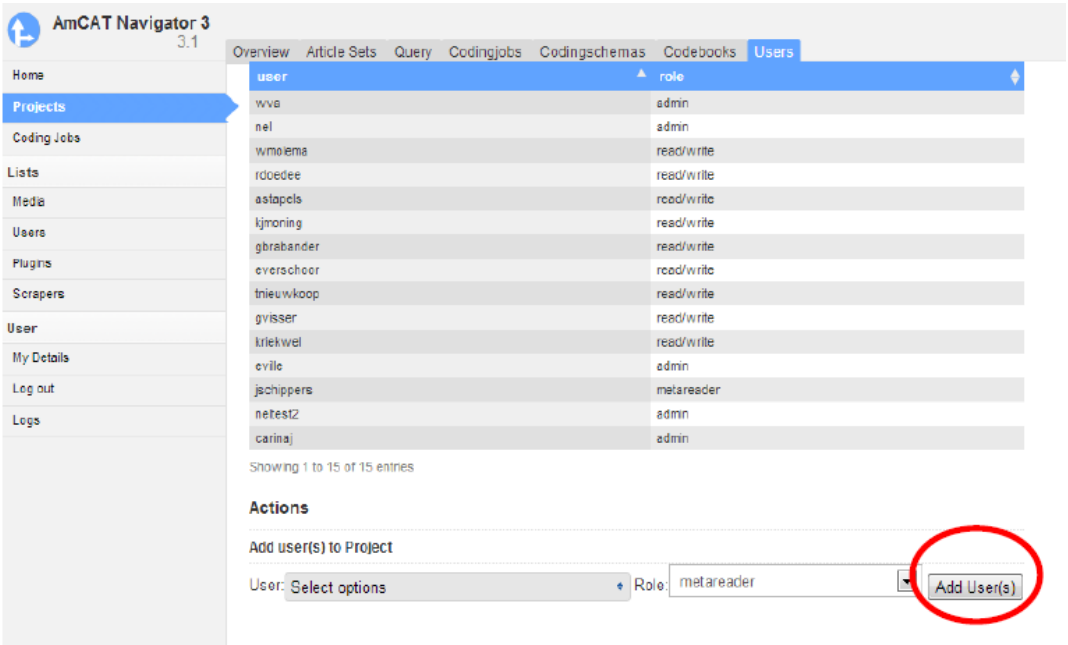
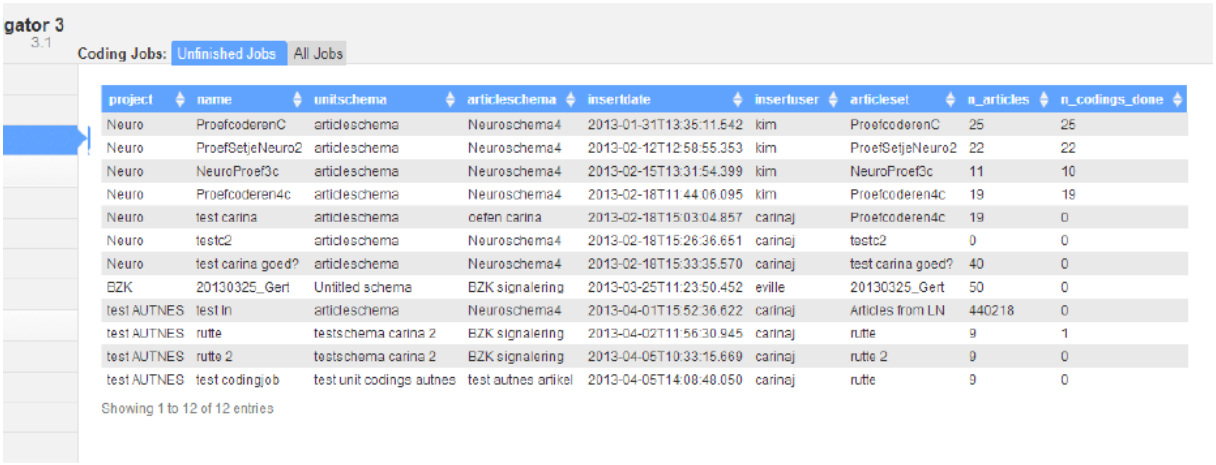


Figure 45. Adding a user to a project.

Below the overview of users in the project, there is a button for adding new users to the project, in the red circle on the above screenshot. Select the user(s) you want to add to the project to the list, select their role within the project, and click on ‘Add User(s)’.

9 The AmCAT Annotator

The AmCAT Annotator is used for coding data manually. Go to the Annotator by clicking on 'coding jobs' in the menu to the left of the screen, or by clicking on the 'codingjobs' panel on the start screen. You will then go to the overview of codingjobs for which you are assigned as the coder.



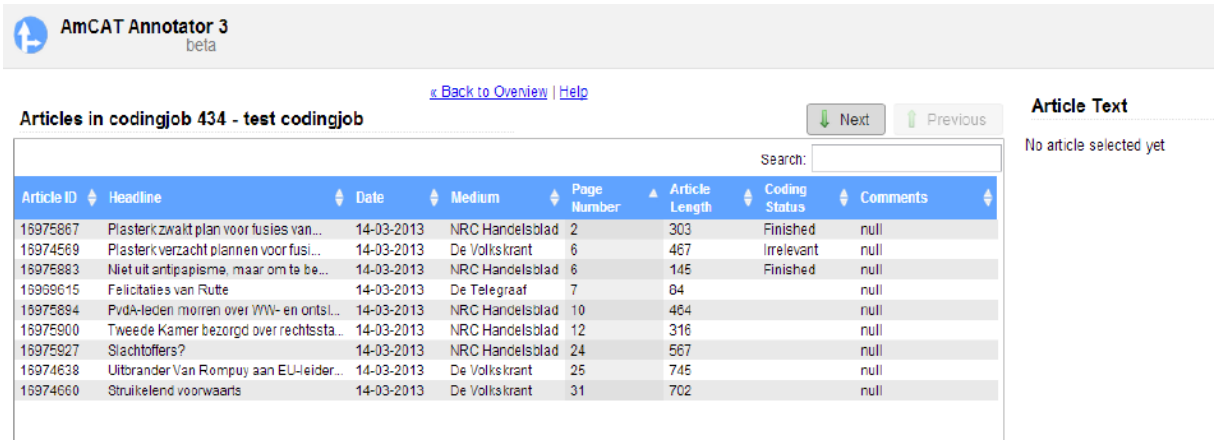
project	name	unitschema	articleschema	insertdate	insertuser	articleset	n_articles	n_codings_done
Neuro	ProefcoderenC	articleschema	Neuroschema4	2013-01-31T13:35:11.542	kim	ProefcoderenC	25	25
Neuro	ProefSetjeNeuro2	articleschema	Neuroschema4	2013-02-12T12:58:55.353	kim	ProefSetjeNeuro2	22	22
Neuro	NeuroProef3c	articleschema	Neuroschema4	2013-02-15T13:31:54.399	kim	NeuroProef3c	11	10
Neuro	Proefcoderen4c	articleschema	Neuroschema4	2013-02-18T11:44:06.095	kim	Proefcoderen4c	19	19
Neuro	test carina	articleschema	oefen carina	2013-02-18T15:03:04.857	carinaj	Proefcoderen4c	19	0
Neuro	testc2	articleschema	Neuroschema4	2013-02-18T15:26:36.651	carinaj	testc2	0	0
Neuro	test carina goed?	articleschema	Neuroschema4	2013-02-18T15:33:35.570	carinaj	test carina goed?	40	0
BZK	20130325_Gert	Untitled schema	BZK signalering	2013-03-25T11:23:50.452	eville	20130325_Gert	50	0
test AUTNES	test ln	articleschema	Neuroschema4	2013-04-01T15:52:36.622	carinaj	Articles from LN	440218	0
test AUTNES	rutte	testschema carina 2	BZK signalering	2013-04-02T11:56:30.945	carinaj	rutte	9	1
test AUTNES	rutte 2	testschema carina 2	BZK signalering	2013-04-05T10:33:15.669	carinaj	rutte 2	9	0
test AUTNES	test codingjob	test unit codings autnes	test autnes artikel	2013-04-05T14:08:48.050	carinaj	rutte	9	0

Showing 1 to 12 of 12 entries

Figure 46. Overview of codingjobs to be coded in the Annotator.

The overview shows the specifics of each codingjob: the project that the codingjob is part of, the name of the codingjob, the unitschema and the articleschema used in the codingjob (see page 32), the date that the codingjob was created, the user who created the codingjob, and lastly the articleset that contains the articles used in the codingjob. **n_articles** shows the total number of articles included in the codingjob, **n_codings_done** shows how many articles are given the status 'finished' by the coder (which, if done correctly, means that they are fully coded).

You can use the tabs above the table to switch between an overview of all codingjobs assigned to you, and an overview of all codingjobs assigned to you that still include unfinished articles. Click on one of the codingjobs to see an overview of articles included in that job, which you then can select for coding in the Annotator.



Article ID	Headline	Date	Medium	Page Number	Article Length	Coding Status	Comments
16975867	Plasterk zwakt plan voor fusies van...	14-03-2013	NRC Handelsblad	2	303	Finished	null
16974569	Plasterk verzacht plannen voor fusi...	14-03-2013	De Volkskrant	6	467	Irrelevant	null
16975883	Niet uit antipapisme, maar om te be...	14-03-2013	NRC Handelsblad	6	145	Finished	null
16969615	Felicitaties van Rutte	14-03-2013	De Telegraaf	7	84		null
16975894	PvdA-leden morren over WW- en ontsl...	14-03-2013	NRC Handelsblad	10	464		null
16975900	Tweede Kamer bezorgd over rechtssta...	14-03-2013	NRC Handelsblad	12	316		null
16975927	Slachtoffers?	14-03-2013	NRC Handelsblad	24	567		null
16974638	Uitbrander Van Rompuy aan EU-leider...	14-03-2013	De Volkskrant	25	745		null
16974660	Struikelend voorwaarts	14-03-2013	De Volkskrant	31	702		null

Figure 47. Overview of articles within a codingjob.

From this overview, you can select an article to start coding. When you click on the 'next' button, you start with the first article in the current codingjob. 'Back to overview' brings you back to the overview

of codingjobs, ‘help’ gives an overview of keyboard shortcuts for sentence codings, that you can use to code faster (also included in this manual, on page 45).

9.1 Coding

When you click on one of the articles in the list, the coding screen for that article is opened. The overview of all articles remains visible on the top of the screen. To the right of the screen, the text of the article is shown. Yellow words are important words that are highlighted. This function is still under construction.

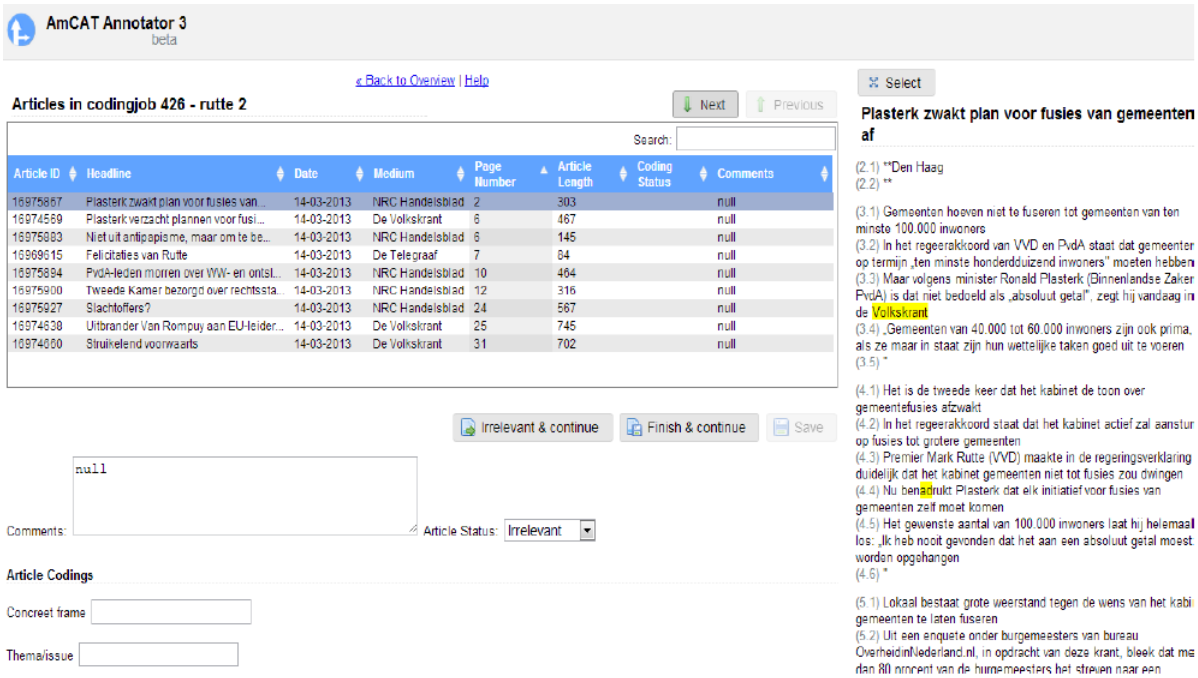


Figure 48. Coding an article in the AmCAT Annotator.

Below the table with the overview of articles, there are three buttons. ‘Irrelevant & continue’ marks the artikel as irrelevant, saves this status, and immediately shows the next article. Irrelevant articles remain in the codingjob, but in the output, most variables are empty. The button ‘Finish & continue’ gives the article the ‘finished’ status, saves all codings, and shows the next article in the codingjob. The button ‘save’ saves your progress thus far, and changes the article status to ‘in progress’.

The field ‘Comments’ above the article codings is for entering comments. These comments can only be read by the coder himself/herself, they cannot be used for asking questions to the project leader.

The field to the right of ‘Comments’ shows the article status. Articles can have the following statuses: ‘not started’, ‘in progress’, ‘finished’ or ‘irrelevant’. You can change an article’s status manually and than click on ‘save’ to save changes.

As said previously, codings in AmCAT are divided into codings on the article level and codings on the unit level. Articles can be coded on one or both levels in the same codingjob. In the AmCAT Annotator, the article codings are displayed on top, followed by the codings on the unit level.

Figure 48. Article codings in the AmCAT Annotator.

Codings on the article level consist of variables (often in the form of questions) and fields where you can fill out the value for each variable. Click on 'save' to save your progress. It is advised that you do so regularly. When you have filled out a value that is not in line with the possible values for that variable, AmCAT will refuse to save your codings, and marks the field that contains the error with a red line. You can save your codings after correcting the value in this field.

Below the codings on the article level, the table used for the sentence (unit) codings is shown.

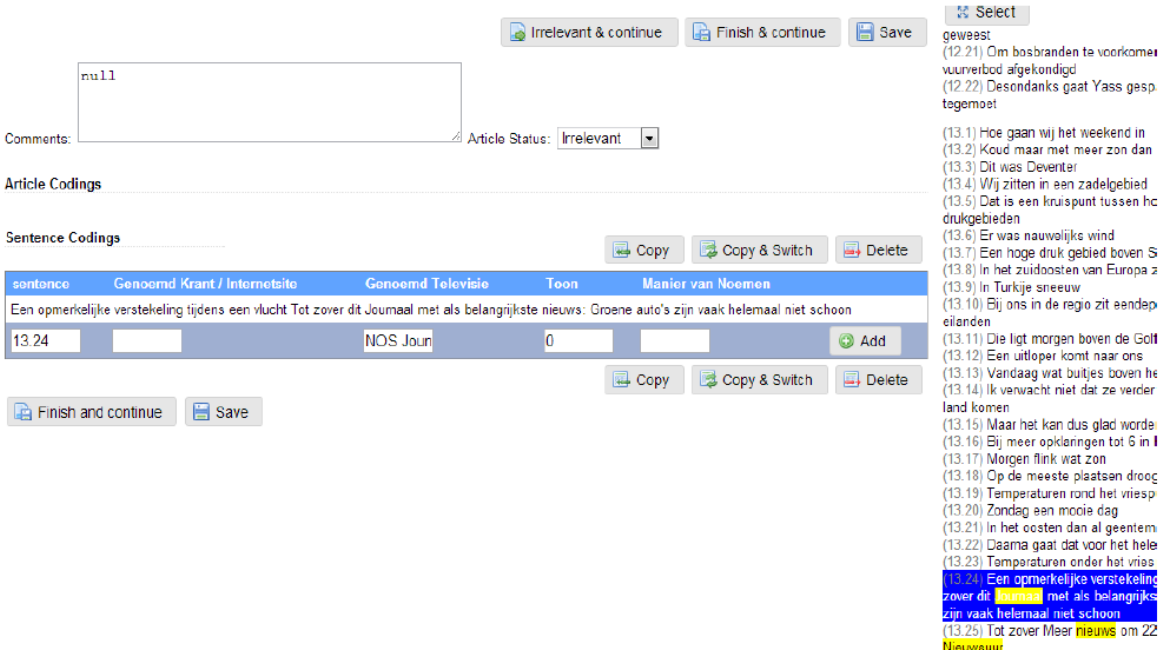


Figure 49. Sentence codings: codings on the unit level in the AmCAT annotator.

Unit codings are displayed in a table. Each column shows one variable, each row contains one unit, that is, one sentence or part of a sentence. You can use the button 'add' , to the right of each row, to add more units if necessary.

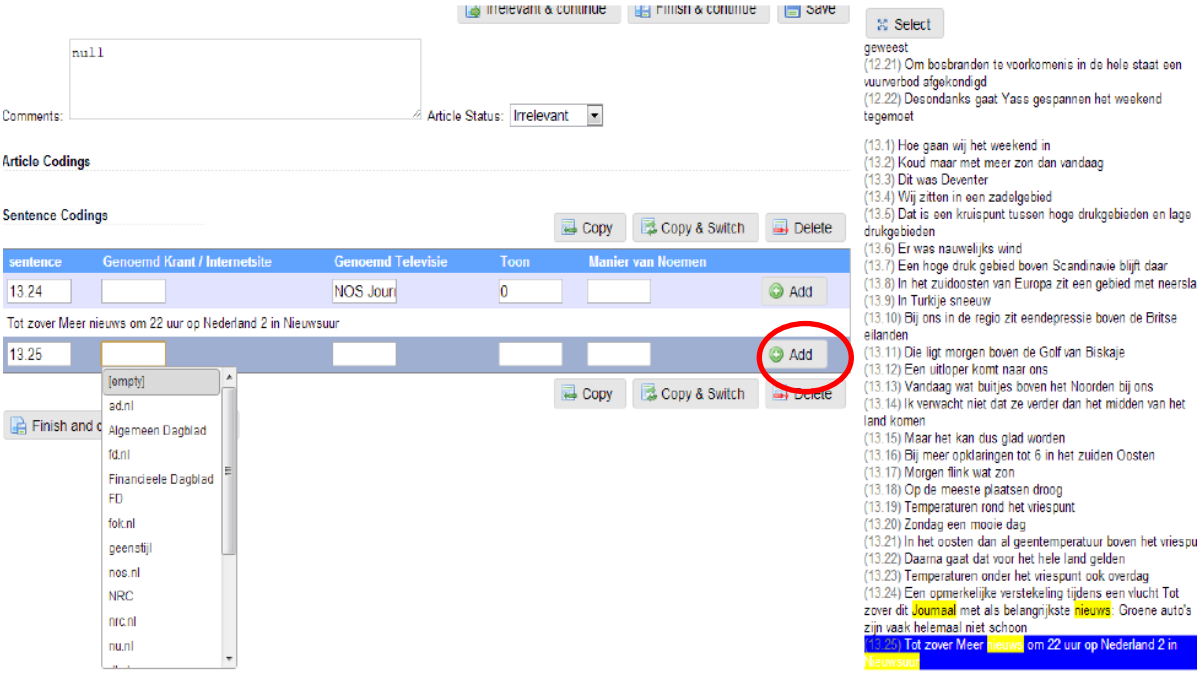


Figure 50. Adding rows for unit codings.

The first variable for unit codings is always the sentence number. When you select the sentence number, the corresponding sentence from the text appears above the coding fields, which is convenient for coding quickly. The same sentence is highlighted in blue in the article text. As with the article codings, you code by filling out or selecting the correct value in each field. The button ‘copy’ on top of the unit codings table, copies the current row, including all codings. ‘Copy & Switch’ works for coding in the NET-methode, and switches subject and object when copying the sentence. The ‘Delete’ button deletes the current row.

All above functions can also be used through keyboard shortkeys instead of clicking on-screen buttons, which will make coding go faster. The table below explains these shortkeys. The tab key is used to jump to the next field, selecting an answer option for each variable can be also be done with the arrow keys and the enter key.

Table 1. Shortkeys to use when coding.

Toets	Uitleg
CTRL + S	Save
ESCAPE	Close dialog
CTRL + arrow key down	Add a new row in the sentence codings table, for the same sentence
CTRL + SHIFT + arrow key down	Add a new row in the sentence codings table, for the next sentence
SHIFT + arrow key down	Copy the current row in the sentence codings table
ALT + arrow key down	Copy the current row in the sentence codings table, and switch subject and object (only for NET coderings)

Click on ‘finish & continue’ when you have finished all article and unit codings for the article. This saves all codings, switches the coding status of the article to ‘finish’ and displays the next article in the coding job. This does not work for the last article in a codingjob; here, you can put the article status on ‘finished’ manually and click ‘save’. AmCAT gives a warning when you try to exist the screen without saving your progress. Non-saved codings will be lost.

N.B. You cannot make any changes to the variables, their type of answer categories (open questions, numerical values, etc.) or to the answer options within the AmCAT Annotator. If you want to make any changes to the codingjob or the codingschema, you first have to go back to the Navigator, select the project, and make your changes from there.

10 Additional lists in the AmCAT Navigator

The AmCAT Navigator contains a number of lists that give an overview of all resources included in AmCAT, regardless of project. You can exit these through the menu options under the heading 'Lists' in the menu to the left of the screen.

10.1 Media

'Media' brings up a list of all media registered in AmCAT. You have to add a new medium to this list before you can upload new articles from that medium.

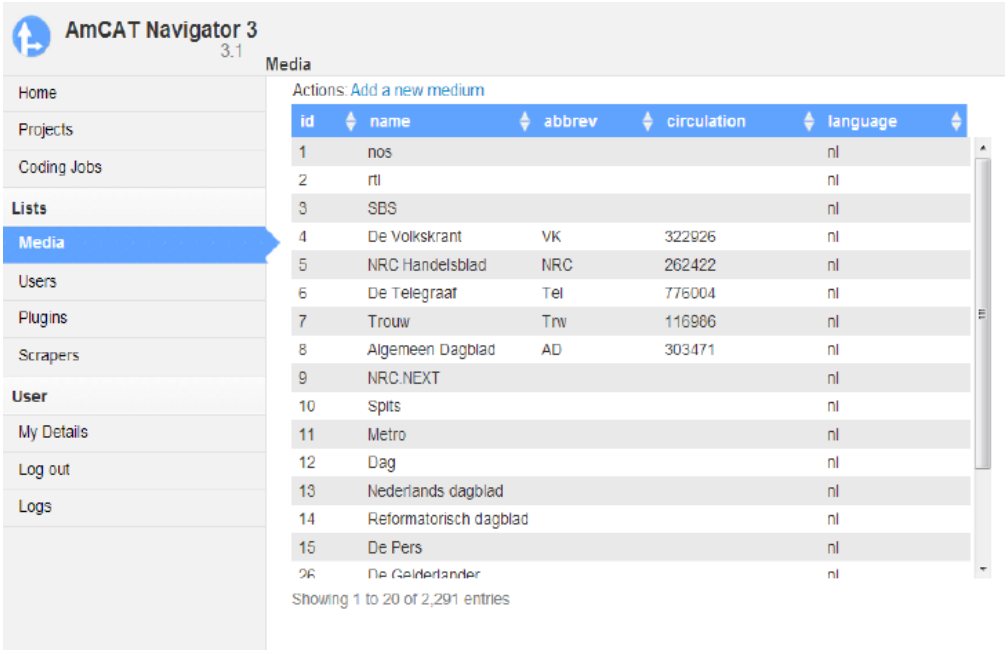


Figure 51. The media list.

The media list shows the ID (the unique number for each medium in AmCAT), the name and the language for each medium. In case an abbreviation for the medium name and/or its circulation number have been entered at the time the medium was registered, the table shows these as well.

By clicking 'add a new medium' above the media list, you can register a new medium in AmCAT. To do so, you enter its name and language, and optionally and abbreviation of the medium name, as well as circulation numbers. Click on 'add medium' to add the medium to the list. AmCAT assigns an ID number to the medium.

10.2 Users

'Users' shows a list of all AmCAT users.

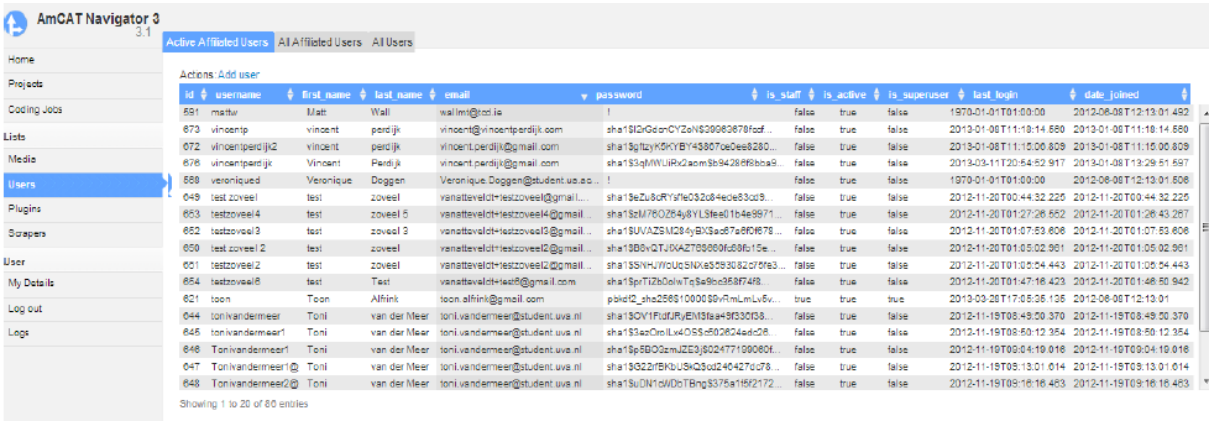


Figure 52. User list.

Using the tabs, you can switch between lists of 'active affiliated users', 'affiliated users' and all AmCAT users. The way of categorizing users into affiliations will be updated in the near future.

By clicking on 'add user' you can add one or more users to AmCAT. Enter the new user's personal details and click on 'submit'. The new user will receive a password by email. If you want to add multiple new users at once, you can upload their user details as a .csv file, using the following layout:

username,"email","first_name","last_name"
arnolds,"arnold@gmail.com","Arnold","Schwarzenegger"

The details for additional new users can be put in a new row for each user.

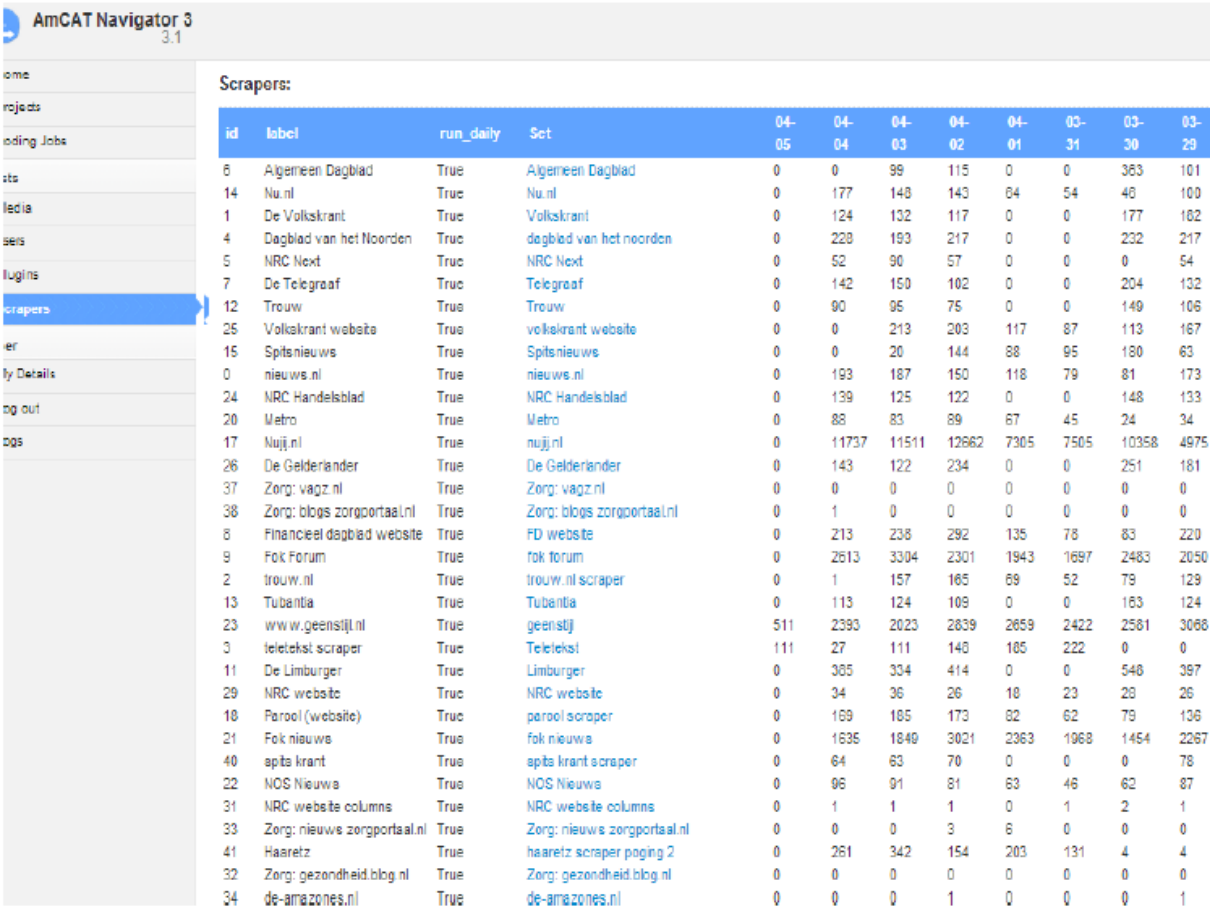
N.B. The user list also has a column for 'password' . The passwords shown in this row are encrypted. Other users cannot see your AmCAT password.

10.3 Plugins

A future update will show a list of all active plugins in AmCAT here.

10.4 Scrapers

'Scrapers' shows a list of all scrapers currently linked to AmCAT. Scrapers are programs that 'scrape' (extract) the content of a web site at timed intervals (for example, the news added to a news site, scraped on a daily basis), and put it in an assigned articleset in AmCAT.



id	label	run_daily	Set	04-05	04-04	04-03	04-02	04-01	03-31	03-30	03-29
6	Algemeen Dagblad	True	Algemeen Dagblad	0	0	99	115	0	0	363	101
14	Nu.nl	True	Nu.nl	0	177	148	143	64	54	46	100
1	De Volkskrant	True	Volkskrant	0	124	132	117	0	0	177	162
4	Dagblad van het Noorden	True	dagblad van het noorden	0	228	193	217	0	0	232	217
5	NRC Next	True	NRC Next	0	52	90	57	0	0	0	54
7	De Telegraaf	True	Telegraaf	0	142	150	102	0	0	204	132
12	Trouw	True	Trouw	0	90	95	75	0	0	149	106
25	Volkskrant website	True	volkskrant website	0	0	213	203	117	87	113	167
15	Spitsnieuws	True	Spitsnieuws	0	0	20	144	88	95	180	63
0	nieuws.nl	True	nieuws.nl	0	193	187	150	118	79	81	173
24	NRC Handelsblad	True	NRC Handelsblad	0	139	125	122	0	0	148	133
20	Metro	True	Metro	0	88	83	89	67	45	24	34
17	Nuji.nl	True	nuji.nl	0	11737	11511	12662	7305	7505	10358	4975
26	De Gelderlander	True	De Gelderlander	0	143	122	234	0	0	251	181
37	Zorg: vagz.nl	True	Zorg: vagz.nl	0	0	0	0	0	0	0	0
38	Zorg: blogs zorgportaal.nl	True	Zorg: blogs zorgportaal.nl	0	1	0	0	0	0	0	0
8	Financieel dagblad website	True	FD website	0	213	238	292	135	78	83	220
9	Fok Forum	True	fok forum	0	2613	3304	2301	1943	1697	2483	2050
2	trouw.nl	True	trouw.nl scraper	0	1	157	165	69	52	79	129
13	Tubantia	True	Tubantia	0	113	124	109	0	0	163	124
23	www.geenstijl.nl	True	geenstijl	511	2393	2023	2839	2659	2422	2581	3068
3	teletekst scraper	True	Teletekst	111	27	111	140	185	222	0	0
11	De Limburger	True	Limburger	0	385	334	414	0	0	548	397
29	NRC website	True	NRC website	0	34	36	26	18	23	28	26
18	Parool (website)	True	parool scraper	0	169	185	173	82	62	79	136
21	Fok nieuws	True	fok nieuws	0	1635	1849	3021	2363	1968	1454	2267
40	apts krant	True	apts krant scraper	0	64	63	70	0	0	0	78
22	NOS Nieuws	True	NOS Nieuws	0	96	91	81	63	46	62	87
31	NRC website columns	True	NRC website columns	0	1	1	1	0	1	2	1
33	Zorg: nieuws zorgportaal.nl	True	Zorg: nieuws zorgportaal.nl	0	0	0	3	6	0	0	0
41	Haaretz	True	haaretz scraper paging 2	0	261	342	154	203	131	4	4
32	Zorg: gezondheid.blog.nl	True	Zorg: gezondheid.blog.nl	0	0	0	0	0	0	0	0
34	de-amazones.nl	True	de-amazones.nl	0	0	0	1	0	0	0	1

Figure 53. Scraper list.

Each scraper in the list has an ID and a name. The column ‘run_daily’ shows whether the scraper is supposed to scrape on a daily basis (true) or not (false).

On the right side of the overview, there is a number of columns headed by a date, such as 03-29 or 03-30. The number below these dates shows the number of articles scraped on that day. This gives an indication of whether the scraper functions correctly or not: a scraper for a news site that scrapes 0 articles on a weekday is probably broken. On a Sunday, this number is to be expected.

The column ‘set’ shows the name of the articleset to which the scraped articles are added. The scraper thus automatically adds articles to the articleset (do **not** make codingjobs from these articlesets). If you click on the name of the articleset, you will see its detailed overview as described in paragraph 3.2.

10.5 Logs

This displays the error log for the current user, which can be consulted by the AmCAT-developers.