AmCAT3

VERY EARLY DRAFT
this document is likely to contain many errors and omissions, please use the latest version

Wouter van Atteveldt

22nd September 2013

Contents

CHAPTER 1

Introduction

1.1 What is AmCAT?

AmCAT is an open-source document management and analysis system. AmCAT allows a user to easily upload and manage documents and analyse them automatically or manually. AmCAT is designed to be an open system: we encourage contributions in the form of code, testing, or documentation. We have tried to make it easy to add functionality, such as upload or scrape scripts or analyses, by using as open standards where applicable and by using a plugin-structure in the places where we expect extensions to be useful.

1.2 How to get / install AmCAT

AmCAT can be downloaded from the google projects page: `http://code.google.com/p/amcat`. Please note that AmCAT is not a typical end-user program: it works as a server to which many users can connect. This means that setting it up is slightly involved. Most users will want to connect to an existing AmCAT server, such as `http://amcat.vu.nl`. However, heavier users or users that which to have more control over their data are encouraged to set up their own server. The wiki page `http://code.google.com/p/amcat/wiki/InstallingAmcat` should make it fairly painless to do so on a computer running a Debian based linux, such as Ubuntu. We have also installed AmCAT on Mac computers, but this is more involved as not all packages can be downloaded as easily.

1.3 More information

The wiki mentioned above, `http://code.google.com/p/amcat/wiki` contains a lot of developer information on AmCAT. There are also two mailing lists: `amcat-user@googlegroups.com` and `amcat-dev@googlegroups.com`. The first list is for user questions and information, while the second is meant for developer questions. The user list will also be used to make announcements, such as new releases.

If you encounter any bug or have a suggestion, do not hesitate to report it using our issue system at google code: `http://code.google.com/p/amcat/issues`

1.4 Disclaimer and license

AmCAT is a not for profit open source initiative by scientists for scientists. We do our best to provide quality software. However, there can always be bugs, hard disk accidents, and other ways that data can be lost and corrupted. Please make sure that everything is always backed up outside the AmCAT system as well.

Slightly more legalistic:

1.5 Outline

The next chapter will show how to access AmCAT and change your user information. The rest of this book is divided over three chapters: Chapter 3 discusses project management and article uploading; Chapter 4 covers automatic analysis and article selection; and Chapter 5 discusses manual coding. Each chapter shows how the different functions of AmCAT can be accessed through the web site. At the end of each chapter, the programmatic API is briefly discussed, showing how the same information can be obtained automatically from R (or other programming languages).

CHAPTER 2

Getting started: Logging in and changing/resetting password

When you visit AmCAT the first time, you are greeted with the login screen showed in Figure 2.1. If you received a username from the administrator, you can use it here. If you installed your own server, it should have created a default user 'amcat' with password 'amcat'. If you don't have an account, you can create a new account using the *'Register'* button. If you've forgotten your password, you can ask for a password reset after a failed logon attempt, as shown in Figure 2.2

If you've correctly logged in, you should now see the Home screen as shown in Figure 2.3. This screen contains the main elements of AmCAT. The Projects button is the main button for researchers, as the articles and codings are defined as part of projects. The Coding button is the main button for coders, where the focus is on coding articles rather than managing the projects. My Details allows you to edit your personal details, while All Users opens the overview of users.

2.1 Changing your password

This is probably a good moment to change your password. This and other user settings can be accessed through the My Details button on the left. As shown in Figure 2.4, this form allows you to change your name, email address, and other information. Make sure that the email address listed here is correct, as that is where we will send password information to if requested.

The *'Change Password'* link on the top will open up the change password form, as shown in Figure 2.5.

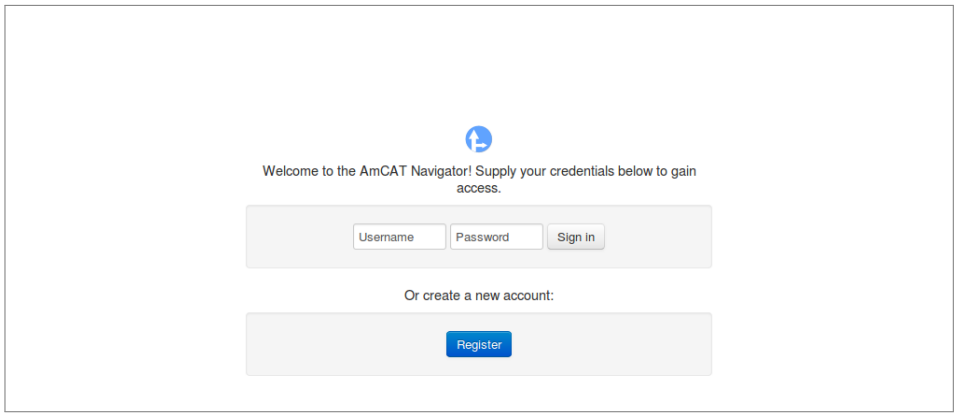

Figure 2.1: Login Screen

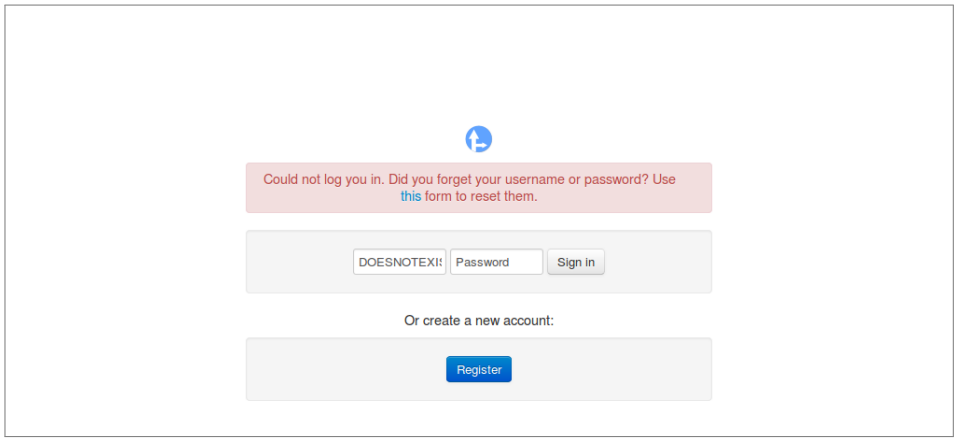

Figure 2.2: Login Screen after a failed login attempt

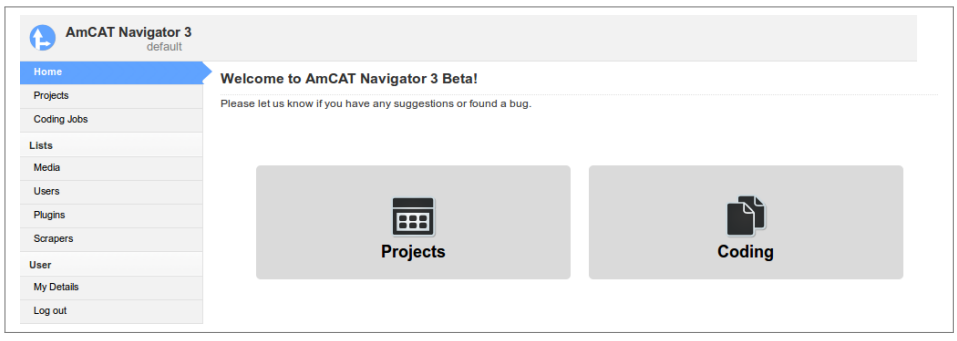

Figure 2.3: Home Screen

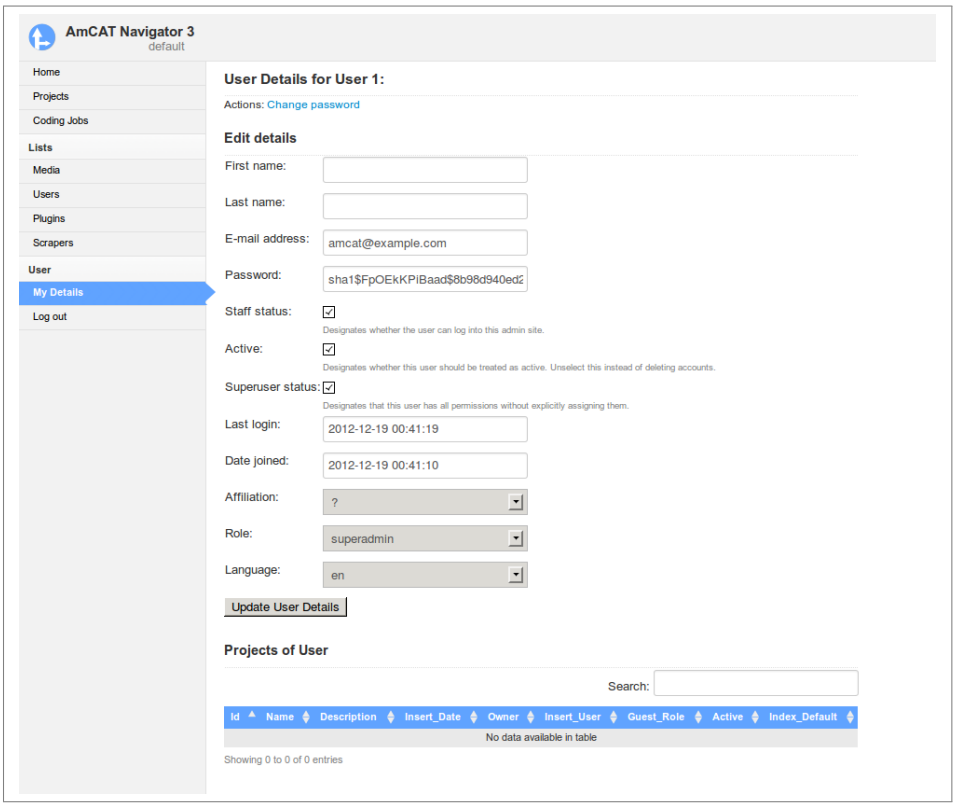

Figure 2.4: User Details

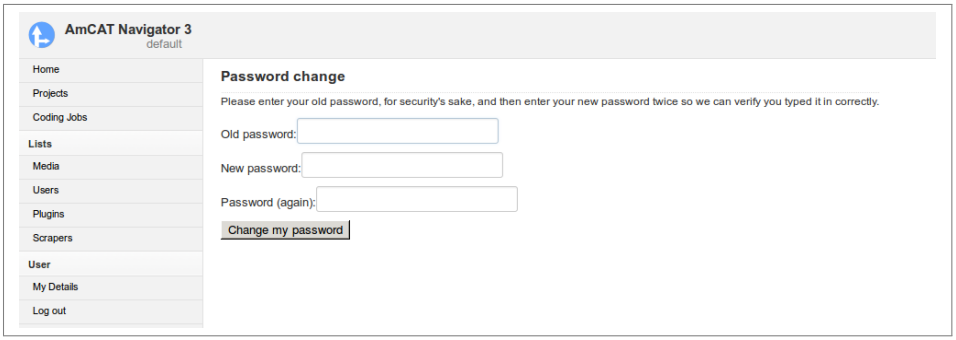

Figure 2.5: Change Password

CHAPTER 3

Managing Projects and Articles

Key concepts explained in this chapter:

Project: Projects are the main unit of organization in AmCAT. Projects contain article sets, codebooks, etc.

Article: Articles or documents are the heart of AmCAT. Essentially, an article is a piece of text with associated metadata, such as publisher, publishing date and author.

Article Set: Article sets are collections of articles and are used to organise the articles in a project. A single article can occur in more than one set.

3.1 Projects

The projects lists gives a list of the projects currently in AmCAT. By default, only active projects in which the current is active are displayed. In this case, the list is empty, as we have started from a clean installation. To create a new project, we can use the *'Add project'* button.

3.1.1 Creating a new project

Figure 3.2 shows the Add Project form (Figure 3.2). The name and description are simply names to call the project. By default, a new project starts out as active.

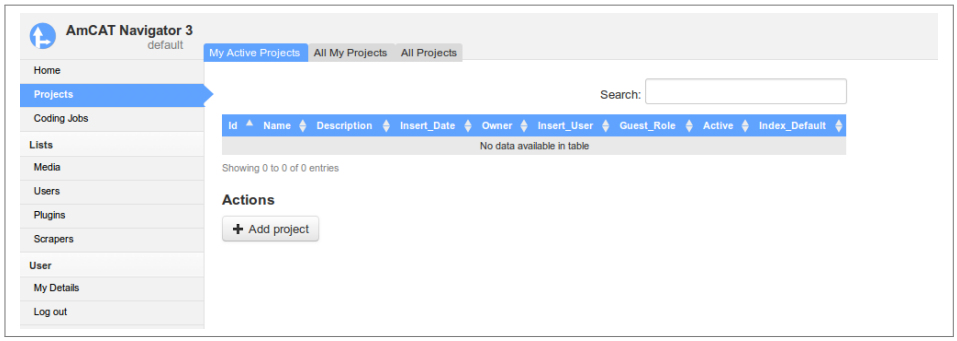

Figure 3.1: Projects List

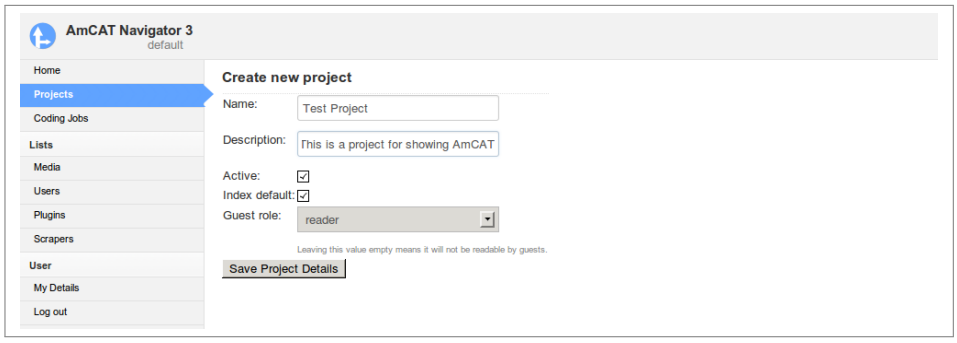

Figure 3.2: Adding a new project

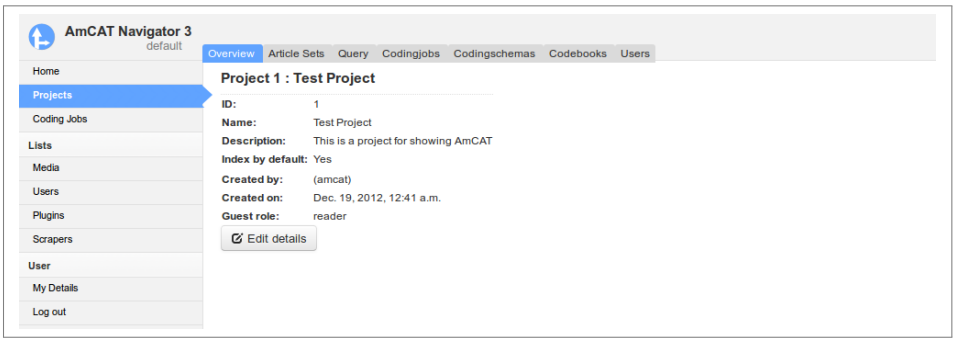

Figure 3.3: Details of the new Project

The owner is the person who 'owns' the project, meaning he or she can perform all management tasks on the project. The owner can later add other users to the project with various rights, see Section 3.1.2.

The Guest Role is the role that a user has in the project if he or she is not added to the project. By setting this to 'reader' (the default), all AmCAT users on this system can see the project and all information in the project. 'Metareader' means that all users can see the project and all information except for the texts of the articles (this is useful for copyright reasons). Finaly, by leaving the role blank (select `-----`), users that are not explicitly added to the project will not be able to access it at all. See Section 3.1.2 for the other roles.

After pressing 'Save project details', the project details screen is opened for the newly created project, as shown in Figure 3.3. This is the same screen that opens when you click on a project in the project list. On this screen, the project number (1) is listed, as well as the details entered in the previous step. An important extra piece of information is 'index by default'. If this is enabled, which is the default behaviour, new articles will be automatically added to the full-text index. All these details can be changed using the Edit Details button.

Upcoming Feature Index by default will be added to the new project form[a]

[a] There is a feature request accepted to improve this. See: `http://code.google.com/p/amcat/issues/detail?id=174`

Projects contain most of the resources involved in managing and analysing texts in AmCAT. These resources are listed in the tabs near the top of the screen.

Article Sets: Article Sets links to the articles that are contained in this project. See Section 3.2, below.

Query: This is the tab for exploratory and automatic analysis of the data. It can be used to get an overview of the data or to do keyword analyses. This tab will be discussed in depth in Chapter 4.

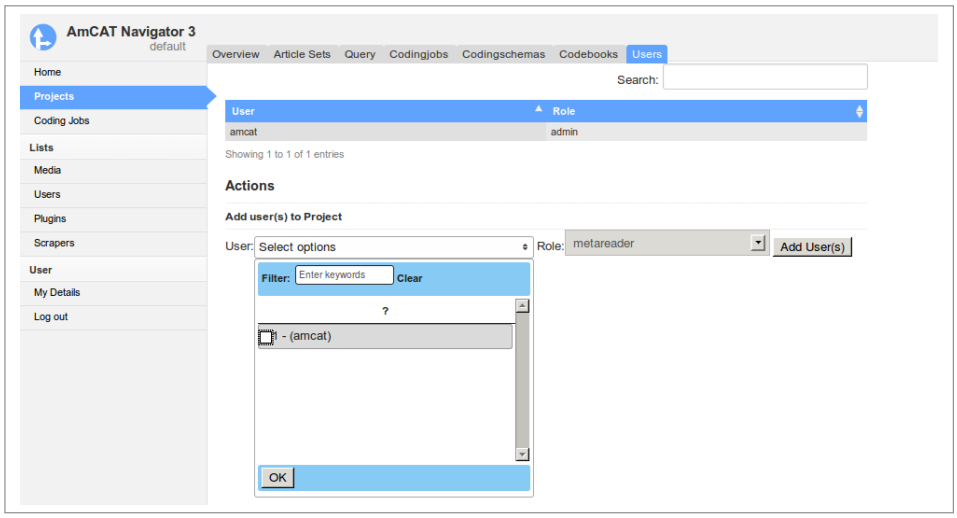

Figure 3.4: Project user Management

Codebooks, Codingschemas, and Codingjobs: These tabs contain the functionality for manual coding. Codebooks are hierarchical collections or categorizations of concepts that can be used for coding, for example a Topic codebook can contain codes Economy and Trade. Coding Schemas define which variables are to be coded and what values (codebook or other) they can take. Codingjobs are sets of articles that are assigned to be coded by a specific coder using a specific schema. These issues will be discussed in Chapter 5

Users: Finally, the users tab can be used to add or remove people from the project, or to change the roles of users. This will be elaborated on in Section 3.1.2.

3.1.2 Managing project users

Figure 3.4 shows the project users tab. The top part of this tab contains a list of the users currently involved in this project together with their roles. The bottom part allows choosing one or more users to add to the project. The selection dialog allows searching by name or scrolling through the list and checking one or more users to add. The role is a dropdown options box to select which role the new user(s) should get:

Admin: Administrators can make all changes to the project, including assigning other users and deleting resources.

read/write: Users with read/write access can see the whole project and add and remove resources, but cannot add new users or delete the project.

read: Users with only read access can read the whole project but not make any changes.

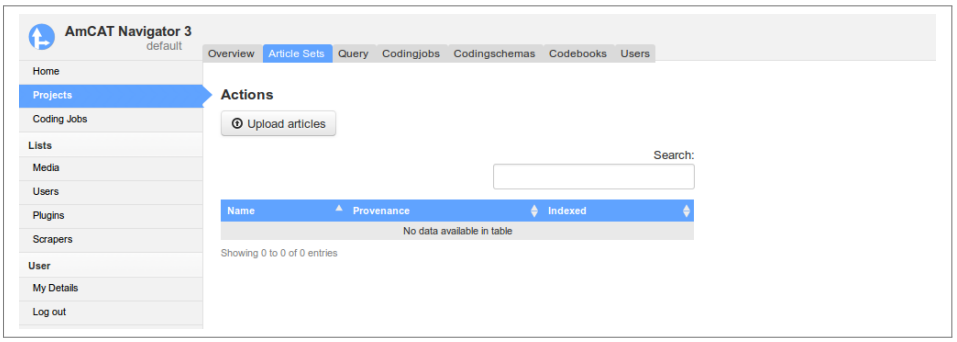

Figure 3.5: Article Set tab

metareader: Finally, users with metareader access can see the project and all resources contained in it, and use the query facility, but cannot read the underlying articles or make any changes.

Upcoming Feature The permissions system is currently quite primitive and has not been tested very thoroughly. It will be overhauled in the next release.[a]

[a] There is a feature request accepted to improve this. See: `http://code.google.com/p/amcat/issues/detail?id=176`

3.2 Articles

Articles or documents for the heart of any AmCAT database. An article consists of a piece of text with associated metadata, such as author, publisher, and date. Like most other objects, each article has a numeric identifier which is unique in the current AmCAT installation. An article also has a globally unique identifier called the UUID, which is unique across AmCAT installations.

In AmCAT, articles are contained in article sets as shown in the *'Article Sets'* tab (Figure 3.5). We can use the *'Upload Articles'* button to upload articles to the current project.

3.2.1 Adding Articles

When articles are added to the system, AmCAT needs to know which part of a file contains the body text and how the needed metadata are encoded. This is done using various article upload plugins. Figure 3.6 shows the plugins as available on a clean install of AmCAT. There are two general plugins, *'text'* and *'csv'*, and some plugins for specific sources such as LexisNexis and mediargus.

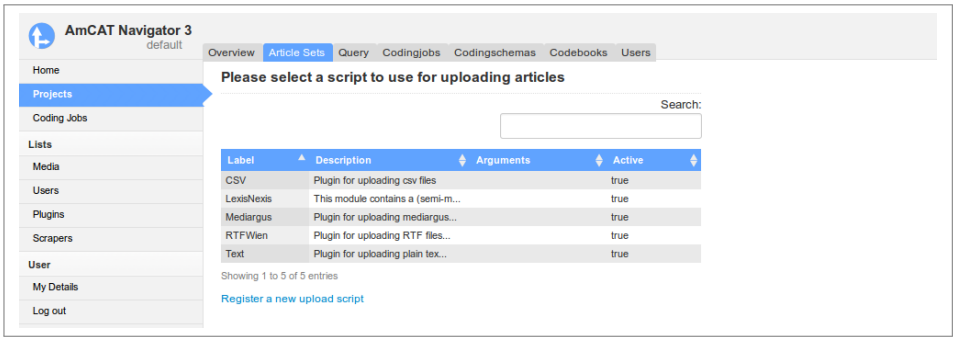

Figure 3.6: Upload articles: Choose an upload plugin

Let's assume that we want to upload a CSV file created in Excel that has columns title, body, and date. In the script selection page we select *'CSV'*. This displays a form where we have to specify the file to upload, the fields to use, and the article set and medium, as shown in Figure 3.7. For the fields to use, we select the csv column to use for each metadata field: *'body'* for the text, *'title'* for the headline, and *'date'* for the date. For article set name, we enter a new name to use for these articles. This will cause a new set to be created and the articles to be placed in that set. If we would have selected an existing article set using the dropdown button, the articles would have been appended to that set. Finally, the medium name is the name of the medium (publisher or source) of the added articles.

After pressing *'Upload'*, the articles are added to the database in a new article set as displayed in an information message, which also contains a link to the newsly created set.

3.2.2 Article Sets

Figure 3.8 shows the article set details for the newly created set. This set contains 9 articles, and each article has an ID, UUID, headline, date, author, medium, and parent. In this case, author and parent are empty, but in e.g. tweets or forum posts these fields are quite useful. The article set also contains a *'provenance'* field, which shows how the articles were obtained.

3.2.3 Full Text Indexing

An important piece of information in the article sets screen is the *'Indexed'* status. This shows whether this article set is ready to be used in full-text querying (keyword analysis). If the status is 'Fully Indexed', the set is ready to be used. A status of 'Indexing in progress' means that AmCAT is busy indexing these articles. Finally, 'Not Indexed' means that indexing is disabled. By using the *'Disable/Enable Indexing'* button, indexing for this set can be turned off or on.

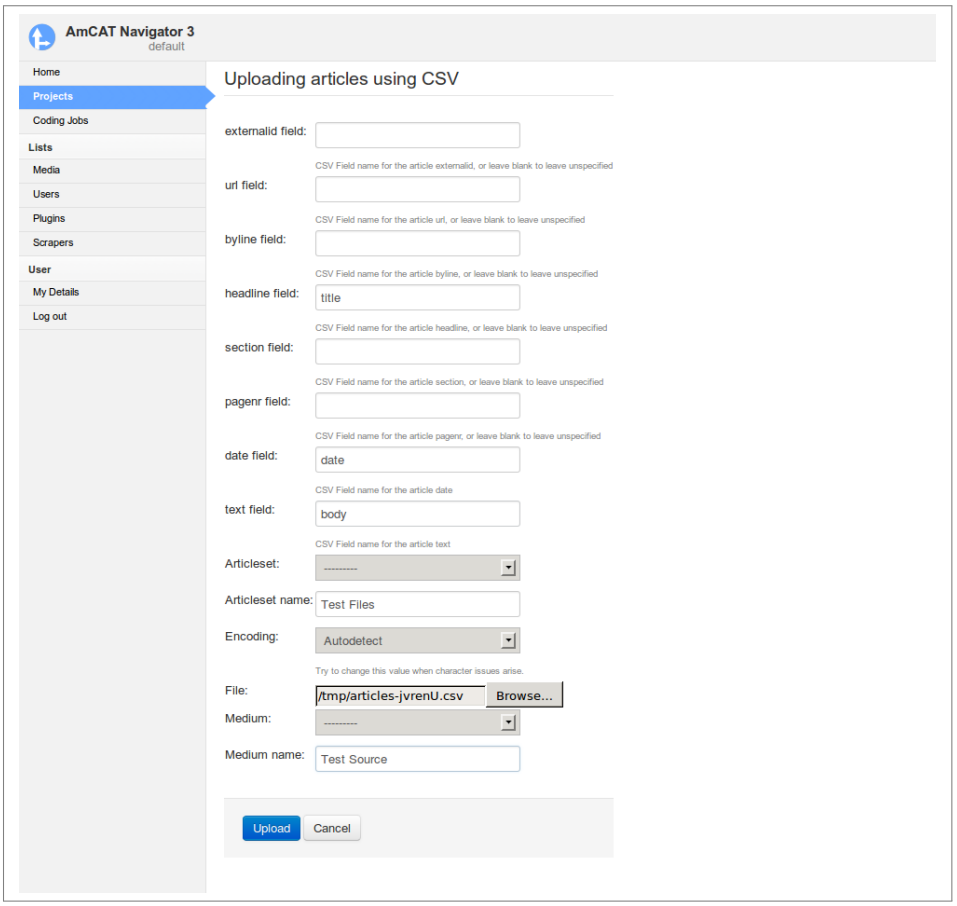

Figure 3.7: Upload articles: CSV field selection

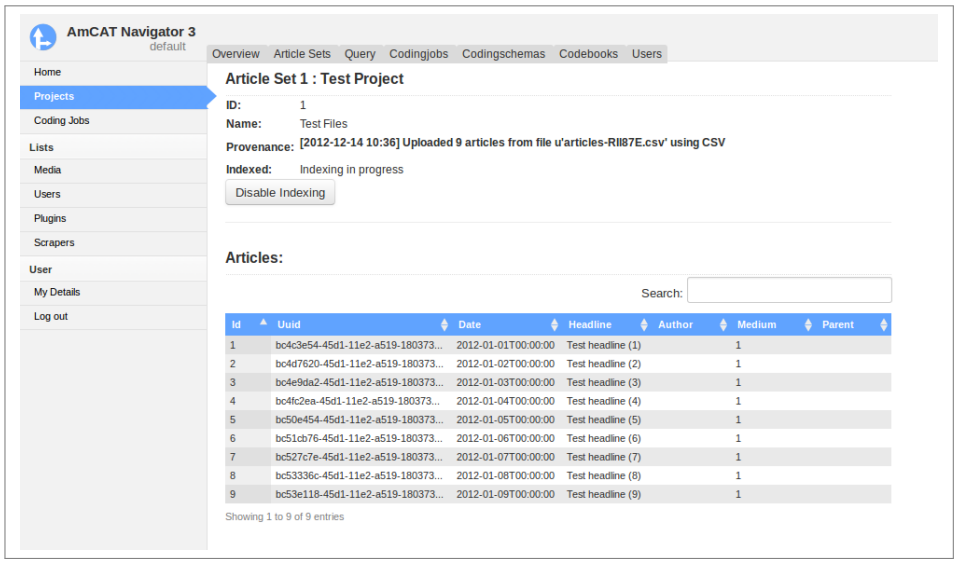

Figure 3.8: Article set with the uploaded articles

For a newly created set, whether indexing is on is determined by the *'index by default'* project setting.

3.2.4 Article Details

Clicking on one of the articles in the article lists brings up the article details screen, as shown in Figure 3.9. This shows the headline and text of the article on the left had side. On the right hand side, the metadata connected to this article is given, including the project and article sets that this article is part of.

3.3 Differences between AmCAT2 and AmCAT3

In AmCAT2, there were a number different objects that each represented a collection of articles. A *'batch'* was a collection of articles that were uploaded or scraped together, and had similar provenance (called *'query'* in AmCAT2). A *'storedresult'* was a save selection of articles from the article selection page. An *'index'* was a collection of articles that could be searched using keyword search. Finally, every coding job created an implicit collection of articles to be coded. In AmCAT3, all these collections are called *'Article Set'*. An AmCAT2 index is an AmcAT3 article set with indexing turned on.

Figure 3.9: Uploaded article details

3.4 Projects API: listing projects from R

AmCAT has a REST API (Application Programming Interface) that makes it easy to communicate with AmCAT from other programs. This API can be reached at `/api` from the AmCAT base url (e.g. if the navigator is at `amcat.vu.nl/navigator`, the API is available at `amcat.vu.nl/api`). If you browse to this location some information is given on which resources are available and how then can be queried, as shown in Figure 3.10. In fact, AmCAT uses this API internally to construct all the tables seen in the web site, so you can be sure that the API data is the same as the data on the site.

One easy way of using the API is from the R language. We have created a package `amcatr` that has a number of useful functions for interfacing with AmCAT. For example, `amcat.connect` connects to the AmCAT API and logs in:

```
> a = amcat.connect(host, username, password)

Succesfully connected to localhost:8765 as amcat
```

(as you can see, the AmCAT installation used for writing this manual is located on my local computer (localhost), but the host can also point to an external AmCAT server, e.g. `amcat.vu.nl`).

Another function, `amcat.getobjects` uses this connection to query the API for specific objects. For example, the code below retrieves the list of projects and article sets:

```
> p = amcat.getobjects(a, "project")
> print(p[, c("id","name","description")])

  id         name                          description
1  1 Test Project This is a project for showing AmCAT
```

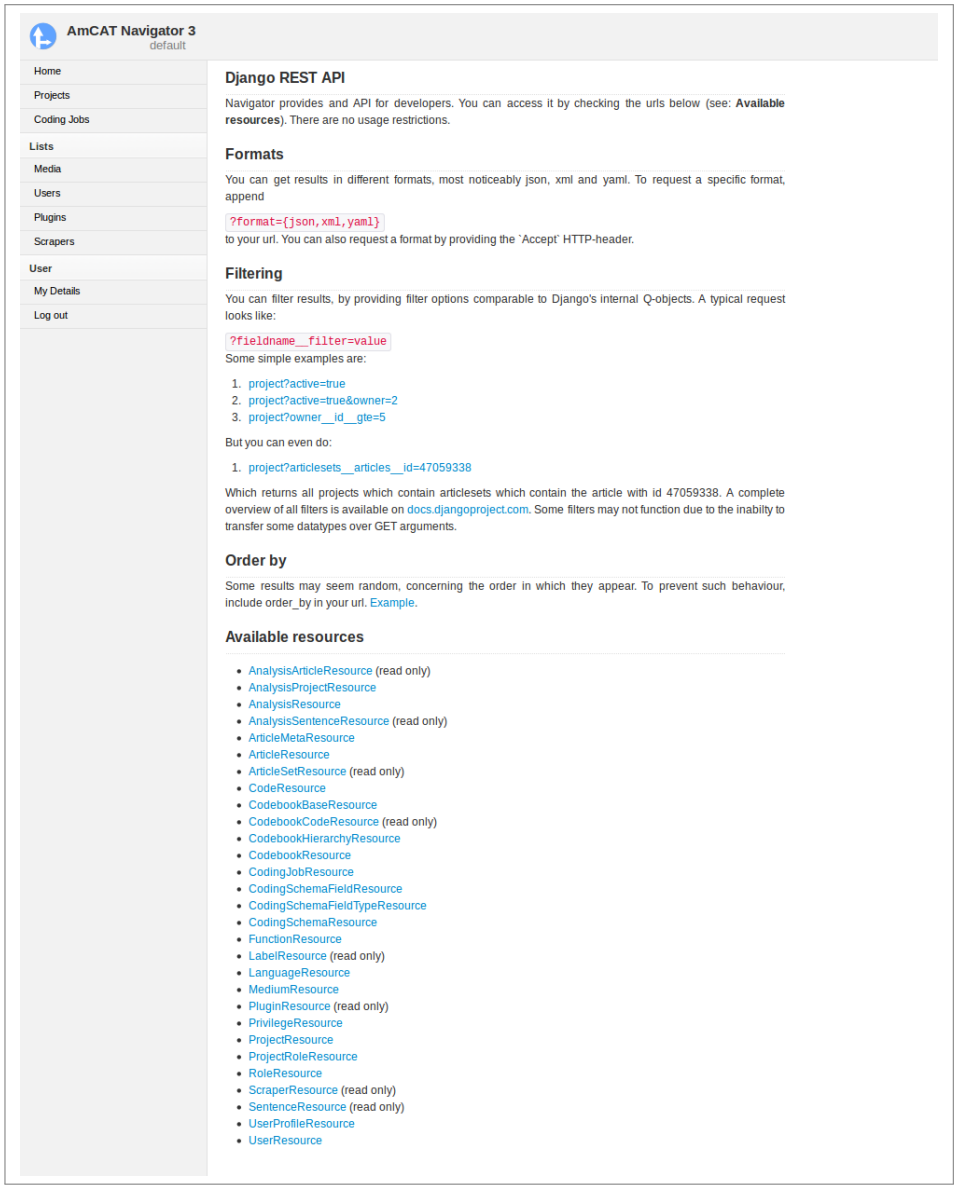

Figure 3.10: AmCAT API Documentation

```
> asets = amcat.getobjects(a, "articleset")
> print(asets[, c("id","name","indexed")])

  id       name              indexed
1  1 Test Files Indexing in progress
```

Of course, in most cases we only want to select a subset of objects. For example, the code below gets a list of all articles that are present in the article set named *'Test Files'*:

setid = asets$id[asets$name == 'Test Files'] articles = amcat.getobjects(a, "article", filter=list(articlesets=setid)) print(articles[, c("id","headline")])

Note: the `article` resource contains the headline and full text of the article. This is useful for browsing or exporting the results. If you only wish to get access to the meta information such as medium and date, use the `articlemeta` resource, which will be much faster for large sets.

CHAPTER 4

Querying

4.1 Counting Articles

4.2 Keyword Analysis

4.3 API Use / Integrating with R

CHAPTER 5

Manual Content Analysis

5.1 Codebook

5.2 Coding Schema

5.3 Coding Job

5.4 Coding

5.5 Viewing results

5.6 API use / Integrating with R