

# QualCoder



QualCoder is free software for qualitative data analysis

## **Copyright Notice**

The MIT License (MIT)

Copyright (c) 2021 Colin Curtain

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1. Edition February 2021

Author: Colin Curtain

ccbogel@hotmail.com

# Table of Contents

1	What is QualCoder.....	5
1.1	Why use QualCoder?.....	5
2	Downloading files and dependencies.....	7
2.1	Preparatory Downloads.....	7
2.1.1	Microsoft Windows.....	7
2.1.2	Debian/Ubuntu Linux.....	7
2.1.3	Fedora/CentOS/RHEL Linuxes.....	8
2.1.4	Mac OS.....	8
3	Starting QualCoder.....	10
3.1	Linux.....	10
3.2	Windows.....	10
3.3	Mac OS.....	10
3.4	Backups.....	10
4	How to use.....	11
4.1	Settings.....	11
4.2	Create a new project.....	12
4.3	Files and Cases Menu.....	13
4.3.1	Manage Files.....	13
4.3.2	Text file editing.....	15
4.3.3	Viewing audio and video.....	15
4.3.4	Transcribing audio and video.....	16
4.3.5	Manage Cases.....	17
4.3.6	Attributes.....	20
4.3.7	Journals.....	21
4.3.8	Manage bad links to files.....	22
4.4	Coding menu.....	22
4.4.1	Categories and codes.....	22
4.4.2	Coding text.....	23
4.4.3	Coding images.....	26
4.4.4	Coding audio and video.....	27
4.5	Reports.....	30
4.5.1	Coding Reports.....	30
4.5.2	Node graph.....	31
4.5.3	Coding Comparison.....	32
4.5.4	Code relations.....	33
4.5.5	Code frequencies.....	34
4.5.6	SQL Statements.....	35
5	Imports and exports.....	37
5.1	Import survey.....	37
5.2	REFI-QDA.....	38
5.3	RQDA.....	38
5.4	Using the project other operating systems.....	38
5.5	Codebook.....	38
6	Other details about QualCoder.....	40
6.1	Acknowledgements.....	40
6.2	Publications citing QualCoder.....	40
7	Future plans.....	41

8 About the author.....	42
-------------------------	----

# 1 What is QualCoder

---

QualCoder is free, open source software for qualitative data analysis.

With QualCoder you can code text and images, write journal notes and memos. You can categorise codes into a tree-like hierarchical categorisation scheme. Coding for audio and video can be performed and requires the VLC media player.

Reports can be generated for text coding and for coder comparison using the Cohen's Kappa statistic. A graph displaying codes and categories can be generated to visualise the coding hierarchy. Most reports can be exported as html, open document text, csv or plain text files.

QualCoder is written in python 3 using Qt5 for the graphical interface. A Sqlite database is used to store the coding data.

The most current QualCoder is available from <https://github.com/ccbogel/QualCoder>.

Releases are available from: <https://github.com/ccbogel/QualCoder/releases> These contain source code and Linux Debian packages (you still need to install some modules using the terminal).

There is also a wordpress site at <https://qualcoder.wordpress.com/>.

The minimum supported python version is 3.6. The minimum recommended screen size is 1024 x 600 pixels.

## 1.1 Why use QualCoder?

**QualCoder is free of charge.** Many qualitative analysis software requiring expensive one-time fees or monthly subscriptions. Not everyone can afford expensive fees.

**QualCoder is easy to use.** It has all you need to perform qualitative analysis without the complicated interfaces of some alternatives.

**QualCoder works offline.** Internet is not always available and QualCoder does not require internet to work.

**QualCoder is not tied to a computer.** If you change workplace you do not have to worry about being tied to your former workplace's license or to buy a new license. QualCoder license allows you to use the software regardless of where you work or on what computer it is installed on.

**QualCoder is multi-platform.** It runs on Linux, Windows and Mac, this means that you do not have to worry if you change operating systems, and it also means you can collaborate with colleagues on different platforms.

**QualCoder relies on the community.** If you find a bug or have a feature request or feedback, write

it on QualCoder's page on github <https://github.com/ccbogel/QualCoder>.

**QualCoder is always improving.** QualCoder is actively developed meaning that newer, improved versions are released with improvements.

**QualCoder supports open standards.** QualCoder aims to support the REFI-QDA Standard, see <https://www.qdasoftware.org/> You may exchange codebooks and projects with your colleagues even if they do not use QualCoder, as long as the software they use supports the REFI-QDA Standard. It means that you do not risk your data being unavailable. Note, as of December 2020 REFI-QDA Project is still experimental so there may be bugs. Further testing particularly for audio and video exports and imports and relatively-linked files is needed.

**QualCoder can be modified.** You can modify and adapt QualCoder to your needs, or if you do not how you ask someone to do it for you, as long as you release your changes to everyone. This also means you can copy the software and give it to your colleagues or students free of charge.

## 2 Downloading files and dependencies

---

Download the latest QualCoder from GitHub: <https://github.com/ccbogel/QualCoder>. Unzip the folder. The main program code is in the *QualCoder* folder. The Examples folder contains some example files which can be loaded into a test QualCoder project.

### 2.1 Preparatory Downloads

#### 2.1.1 Microsoft Windows

You will need to download and install Python 3. Then install PyQt5, lxml and Pillow modules. A Python 3 release (minimum version 3.6) can be obtained here: <https://www.python.org/downloads/windows/>. Finally, you need to have the VLC media player installed. You can get this from the Windows Store or from their website: <https://www.videolan.org/vlc/download-windows.html>. Please ensure that the Python and VLC downloads are in the same architecture – that is – 32 bit or 64 bit for both. Otherwise QualCoder will not work.

Once a Python 3 release is installed you need to install the required modules using the following command in the Windows Command prompt (The prompt is under Windows System):

```
python -m pip install pyqt5 lxml Pillow ebooklib ply chardet pdfminer.six  
openpyxl
```

Sometimes there are problems recognising the audio/video VLC library file: libvlc.dll

Some solutions are to add the path of the file to PATH here:

<https://stackoverflow.com/questions/42045887/python-vlc-install-problems?noredirect=1>

#### 2.1.2 Debian/Ubuntu Linux

I create QualCoder.deb packages for easy install on Debian/Ubuntu systems. These are found in the releases: <https://github.com/ccbogel/QualCoder/releases>. You will still need to install the additional required modules using the terminal commands shown in the last two sentences at the end of this section.

If you want to install manually, follow these instructions to create a program launcher:

Go to your downloads folder. Make the *install.sh* script executable. The example script below assumes you are in your home directory. Open a terminal and type the following to install QualCoder:

```
cd Downloads/QualCoder
```

Run the *install.sh* file. The install process will ask for your permission to install QualCoder into the /usr/share directory. It will also download the additional python modules.

```
./install.sh
```

The install.sh script will also install various modules as well as the VLC media player. If you do not use the install script, you need to run these commands to install the modules:

```
sudo apt install python3-pip python3-pyqt5 python3-lxml python3-pil vlc qpdf
python3-ebooklib python3-ply python3-six python3-chardet
```

```
sudo python3 -m pip install pdfminer.six openpyxl
```

### 2.1.3 Fedora/CentOS/RHEL Linuxes

Retrieve the current package code from this repository

```
git clone https://github.com/ccbogel/QualCoder.git
```

Install dependencies

```
sudo dnf install python3-pip python3-lxml python3-ply python3-six python3-
chardet python3-qt5 python3-pillow
```

QualCoder uses an Ebook library that you can currently install via a work-around, specified at <https://github.com/ccbogel/QualCoder/issues/72#issuecomment-695962784>

The UNTESTED install\_fedora.sh should install the dependencies and a desktop start icon for Fedora. The script is for python version 3.8.

### 2.1.4 Mac OS

Install [Python3](#) and [VLC](#).

Download Qualcoder-master Zip file and copy it into /Applications.

In a Terminal run these commands to use python 3.9 (you can also use newer versions):

```
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py python3 --version
```

Select 3.9 as an answer

```
python3 get-pip.py
cd /usr/local/bin
sudo ln -s ../../Library/Frameworks/Python.framework/Versions/3.9/bin/pip pip
pip install pyserial
pip install pyqt5 lxml pillow six ebooklib ply chardet pdfminer.six openpyxl
/bin/bash -c "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/master/install.sh)"
brew install qpdf
```

To run QualCoder

```
cd "/Applications/QualCoder-master/qualcoder" python3 qualcoder.py
```

There is no desktop icon launch right now for QualCoder on Mac OS. Open a new Terminal window in the directory and launch with:



```
python3 qualcoder.py
```

Another option to run Qualcoder is shown here [www.maketecheasier.com/run-python-script-in-mac/](http://www.maketecheasier.com/run-python-script-in-mac/)

This means you can right-click on the qualcoder.py file and open with --> python launcher. You can make an alias to the file and place it on your desktop.

## 3 Starting QualCoder

---

### 3.1 Linux

If you have successfully installed a Debian package or successfully run the `install.sh` script there will be a program launcher. Double-click to start.



*Figure 1: QualCoder launcher*

Alternatively, using the terminal, move to the inner QualCoder folder and open the `qualcoder.py` file:

```
python3 qualcoder.py
```

### 3.2 Windows

In Windows create a desktop shortcut to the `qualcoder.py` file and double click to open. Alternatively, using the command prompt, move to the directory and open the `qualcoder.py` file:

```
python3 qualcoder.py
```

From the Windows Start menu (example):

```
C:\Windows\pyw.exe "C:\Other programs\QualCoder-master\qualcoder\qualcoder.py"
```

### 3.3 Mac OS

Using the terminal prompt, move to the directory and open the `qualcoder.py` file:

```
python qualcoder.py
```

### 3.4 Backups

QualCoder has several backup options that can be changed in the settings. QualCoder can make a backup of a project every time the project is opened. Up to five data and hour stamped backups are created, with the oldest being deleted if there are further backups created. Backups can be restricted to only non-audio/video files, for faster smaller but less complete backups.

Another option is to not backup the project every time it is opened. However, I recommend that you make a backup before doing any substantial changes, such as reorganising codes and categories.

## 4 How to use

---

There are a few sample files in the Examples folder. These can be used to test importing files of different document formats and of importing an image. There are also example files to test importing case attributes, and for importing a survey.

### 4.1 Settings

The settings dialog allows you to change several features including language, coder name and font type and font size. Multiple coders can code the same text. A new coder name can be entered in the text box. You can choose an existing coder from the drop down box.

The default language of QualCoder is English. Currently French, German, Italian, Japanese, Spanish are alternative languages. These languages have not been thoroughly reviewed for accuracy of translations. When changing a language, you need to close then reopen QualCoder for the change to occur.

Checking the Show IDs box displays the numeric IDs for codes files cases and more. These IDs may be useful when writing sql queries.

For transcribing audio and video, The time format and bracket type can be selected.

Project backup can be set to automatically backup every hour, or no backup. QualCoder keeping the most recent five backups with a date and hour time stamp. The backup name is: *projectname\_BKP\_yyyymmdd\_hh.qda* Where *hh* is 24 hour time. This also means multiple backups are not done within an hour. So a new backup cannot overwrite another backup created within the same hour. Up to five backups are kept, older ones are deleted.

Rename or move to another location particular backups that you need to keep. When the backups are set, an additional option is to choose to backup the audio/video files. For completeness of backups it is recommended to back up audio and video, however this may slow down opening of the software if you have many large files.

The default project directory is where many file operations will default to, for export or looking to import files.

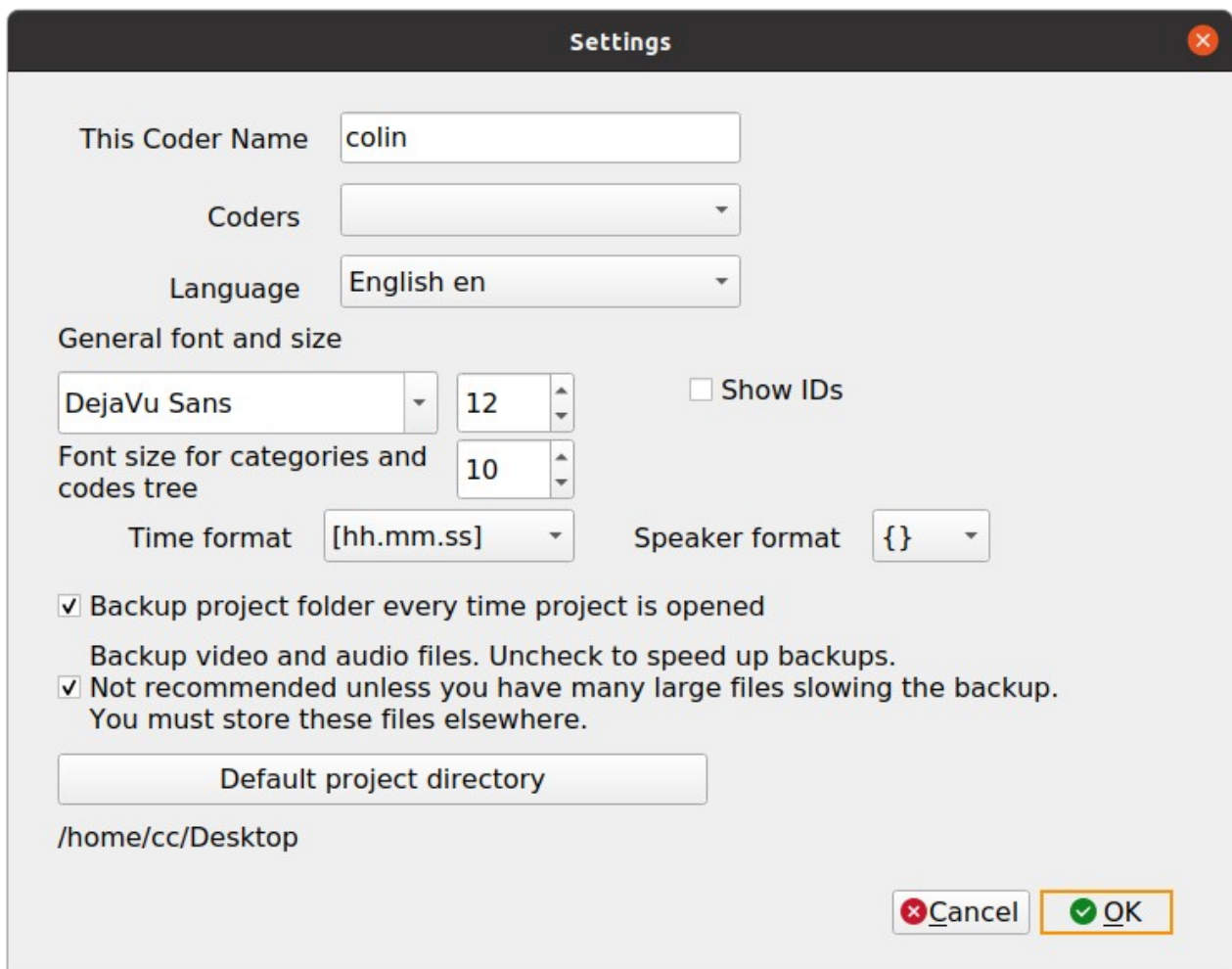


Figure 2: Settings dialog

## 4.2 Create a new project

Create a new project under the ‘Project’ menu, call it test. The project will be saved as *test.qda*. Test.qda is actually a folder containing subfolders which hold the database and other files. Additionally, in your home folder QualCoder will create a .qualcoder folder with a QualCoder.log file for logging events, a config.ini file which will contain the current coder’s name, preferred fonts and a preferred working directory. There will also be a recent\_projects.txt file which stores the most recently opened file. When opening QualCoder the most recent file will automatically be opened.

Under the *Project* menu click on *Project memo*. This is a memo about your project. Type “A test QDA project” and click the OK button.

The main window displays settings and the current project in the Action log, with a menu bar at the top. It has several tabs used for coding, managing files/journals/cases, reports and actions performed.

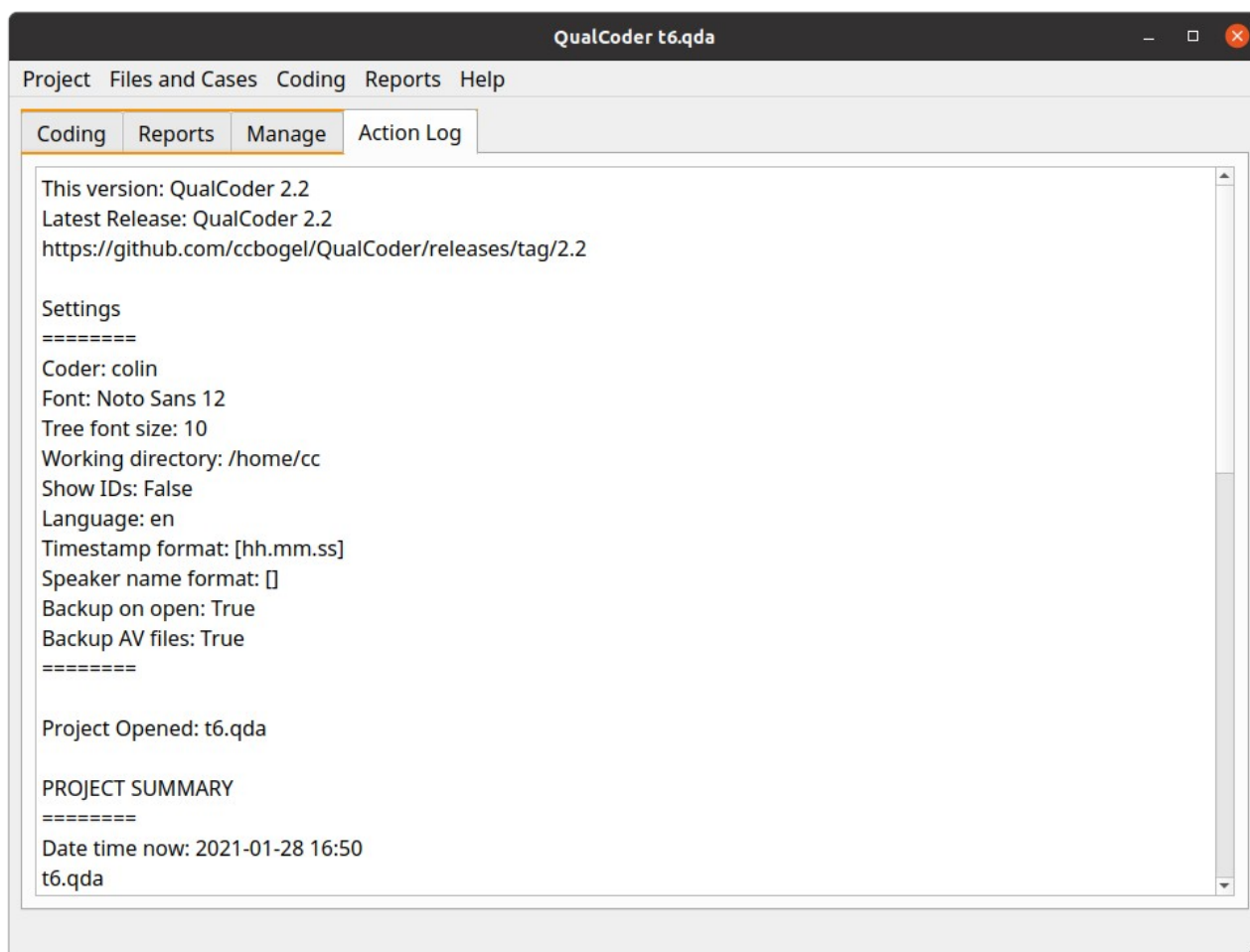


Figure 3: Main screen showing information and the tabs

## 4.3 Files and Cases Menu

### 4.3.1 Manage Files

You will usually want to load text files into QualCoder prior to coding. You can import text from plain text documents (txt), docx, odt, html, htm, md, epub and pdf documents. If the file format is not recognised, QualCoder will try to import as text. Html text is loaded but may need further editing to suit, as formatting will not match the original page. Pdf importing can sometimes be problematic and may need editing. Large Pdf files take a long time to import. Another option is to manually enter text from within QualCoder. Image files in the following formats can be imported: jpg, jpeg and png.

Video (mov, mp4, wmv formats) and audio files (wav, mp3, m4a formats) can be imported. Loading an audio or video file will also automatically create a blank text transcript file. This file will have the same name as the audio or video file, but have a '.transcribed' suffix. Initially, this text transcription will be empty. You can transcribe the file via viewing the audio or video in the Manage Files dialog. You can also get the file professionally transcribed and copy and paste the text into the transcribed text file.

Files can be imported into the project folder, or they can be external to the project folder. External files are linked to. Linked files are shown with a small red link in the file icon. If an externally linked file is moved or deleted, QualCoder will warn you that there is a bad link when the QualCoder project is opened. Links to text files are less important, as the plain text is always imported. Links to images, audio and video need to be correct to work.

A right-click context menu allows you to view, export, delete a current file. The context menu also allow you to import an externally linked file into the QualCoder project folder. It also allows you to export an internal file from the project folder to an external folder location. The small red link in the icon indicated the file is external to the project folder. The menu also allow you to re-order the files according to alphabet, date, and file-type and when right clicking in the name or date columns.

The icons replicate many of these functions. The red icon is for deleting multiple files. The blue plus icon is for creating attributes. The pencil icon is to create a text file. There is no option available to rename the files. It is best to rename files before they are incorporated into the project.

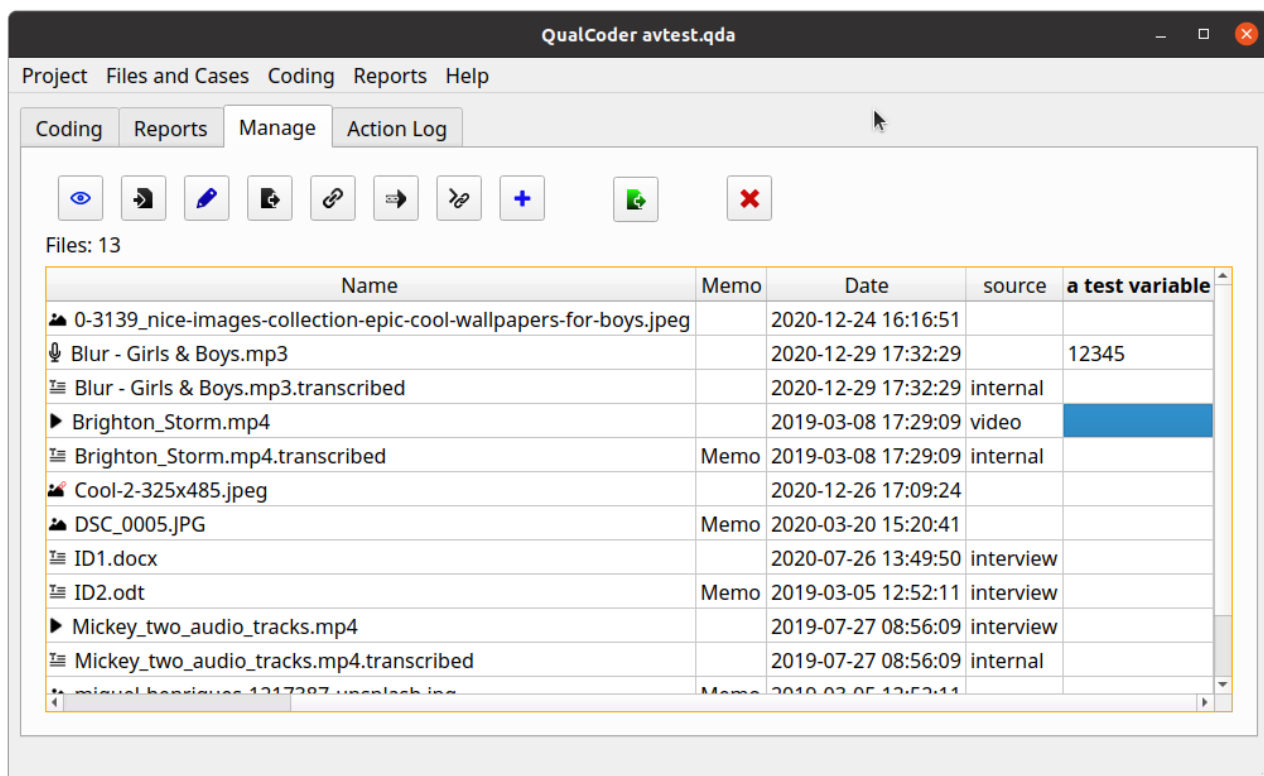


Figure 4: File management

## Attributes

Attributes are variables that can be used to describe or classify the files. These can be added here or through the Manage Attributes menu option. You can show only selected attribute types, if the right-click occurs in an attribute column, shown below. Attributes for files can be exported using the green export button.

Name	Memo	Date	source	bbb
ID2.odt	Yes	2019-03-05 12:52:11	interview	View
Mickey_two_audio_tracks.mp4		2019-07-27 08:56:09		Show this value
Mickey_two_audio_tracks.mp4.transcribed		2019-07-27 08:56:09		Export
miguel-henriques-1217387-unsplash.jpg	Yes	2019-03-05 12:52:11		Delete
psycho-trimmed.mp4		2019-08-09 10:50:48		bb8

Figure 5: Right click on an attribute to show only those files that match

Files larger than 2 Gigabytes are not stored internally. They will always be linked externally to the project.

### 4.3.2 Text file editing

Text files can be edited providing no coding or annotations or case assignment have been performed with the text file. If the file has already had some coding or annotations assigned, the sections of the text file will be shown underlined in red. Editing text is now restricted to small text selections up to 20 characters. This is achieved by selecting some text then right click and select the edit text option.

### 4.3.3 Viewing audio and video

You can open an audio or video file to view. The video file might contain multiple audio tracks. There is a drop down box that allows you to choose another audio track.

Viewing opens two dialogs, one for viewing the video and one for the controls. A transcribed text file must have the same name as the video file, but with a '.transcribed' suffix. The transcribed text file is stored within the sqlite database, but can be exported to a text file.

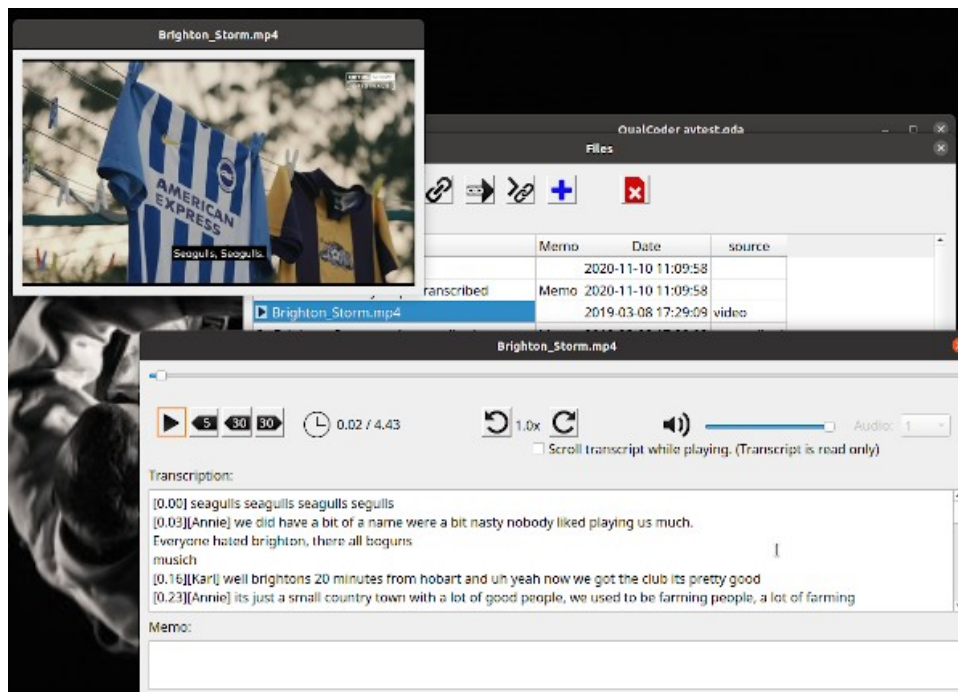


Figure 6: Viewing audio and video

If you have a .srt file (a translation file that is read by VLC) you can place this alongside the video inside the project video folder, shown below. When the video is played, the translation wording will be shown as subtitles in the video. You can open the .srt file in a text editor, and copy and paste this into the video.mp4.transcribed text file. Then this text will be shown as the transcription for the video.

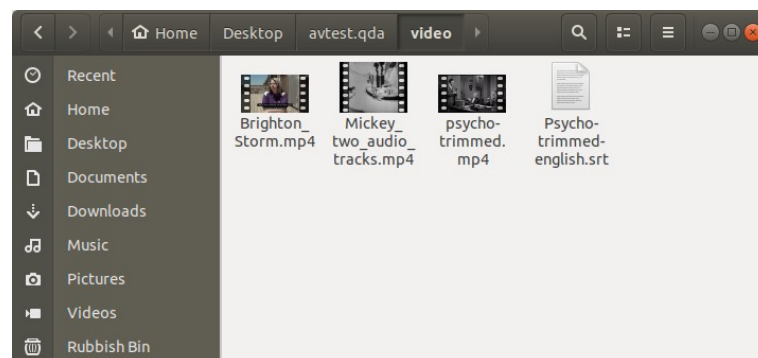


Figure 7: srt file location

#### 4.3.4 Transcribing audio and video

To transcribe open the Mange files menu option, then view the audio or video file. The audio/video will load and there will be a text area to enter transcribed text. QualCoder does not have an automated audio to text feature. Other services such as otter.ai may assist you.

Transcriptions may contain timestamps indicating when the text is being pronounced during the



video. The following formats are recognised by QualCoder, where SSS are milliseconds:

[hh:mm:ss], {hh:mm:ss}, [mm:ss], [hh.mm.ss], [mm.ss], #hh:mm:ss.SSS#

hh:mm:ss,SSS --> hh:mm:ss,SSS

Transcriptions may contain speaker names indicating who is speaking. Speaker names are bracketed in this format: [name] or {name}. Dots ‘.’ and colons ‘:’ cannot be used in speaker names.

Manually transcribing audio and video is helped with some keyboard shortcuts. Transcribing and adding or editing text can only occur if the existing text has no codes or annotations.

The shortcuts available are:

Alt R            Rewind 5 seconds

Shift R            Rewind 30 seconds

Shift F            Forward 30 seconds

Ctrl S or Ctrl P   Stop/Start toggle audio/video. From stop to play will rewind 2 seconds.

Ctrl T            Insert timestamp in this format: [hh.mm.ss]

Ctrl +N           Add a speaker name. This also pauses the audio/video.

Ctrl D            Delete one or more speaker names.

Ctrl 1 to 8       Insert speaker name in this format: [name]

Ctrl Shift >    Increase play rate up to 2 times

Ctrl Shift <    Decrease play rate down to 0.1

### 4.3.5 Manage Cases

A case is typically a source of information and can be associated with one or more files. A case can be classified using attributes. An example case may be a person who has been interviewed (an audio or text interview file) and have published works (e.g. blog posts as files). The person can be assigned attributes such as their location, gender or occupation.

The *Manage Cases* dialog lists the cases in a table. On the right hand side the text of a case is displayed. The number of files associated with a case is shown. Click on the files cell in the table to add or changes files associated with a case. This displays the case file manager dialog, described below.

Cases are useful for seeing text and images linked to particular cases and for assigning attributes

such as age and gender to interview participants. You can rename a case by double-clicking on a case name.

Practical example: Add the three students (or cases) here by clicking the *Add case* button. Call each student: ID1, ID2 and ID3.

If you have already loaded the example id1 and id2 files from the examples folder into your qualcoder project. You can then add each student's file to each student. Click on the cell in the Files column for case ID 1 for example. This opens the case file manager dialog.

The number of files associated with each case is shown in the Files column. Clicking on the files column, or selecting a case and clicking the *Case file manager* icon opens the case file manager where files can be linked to cases.

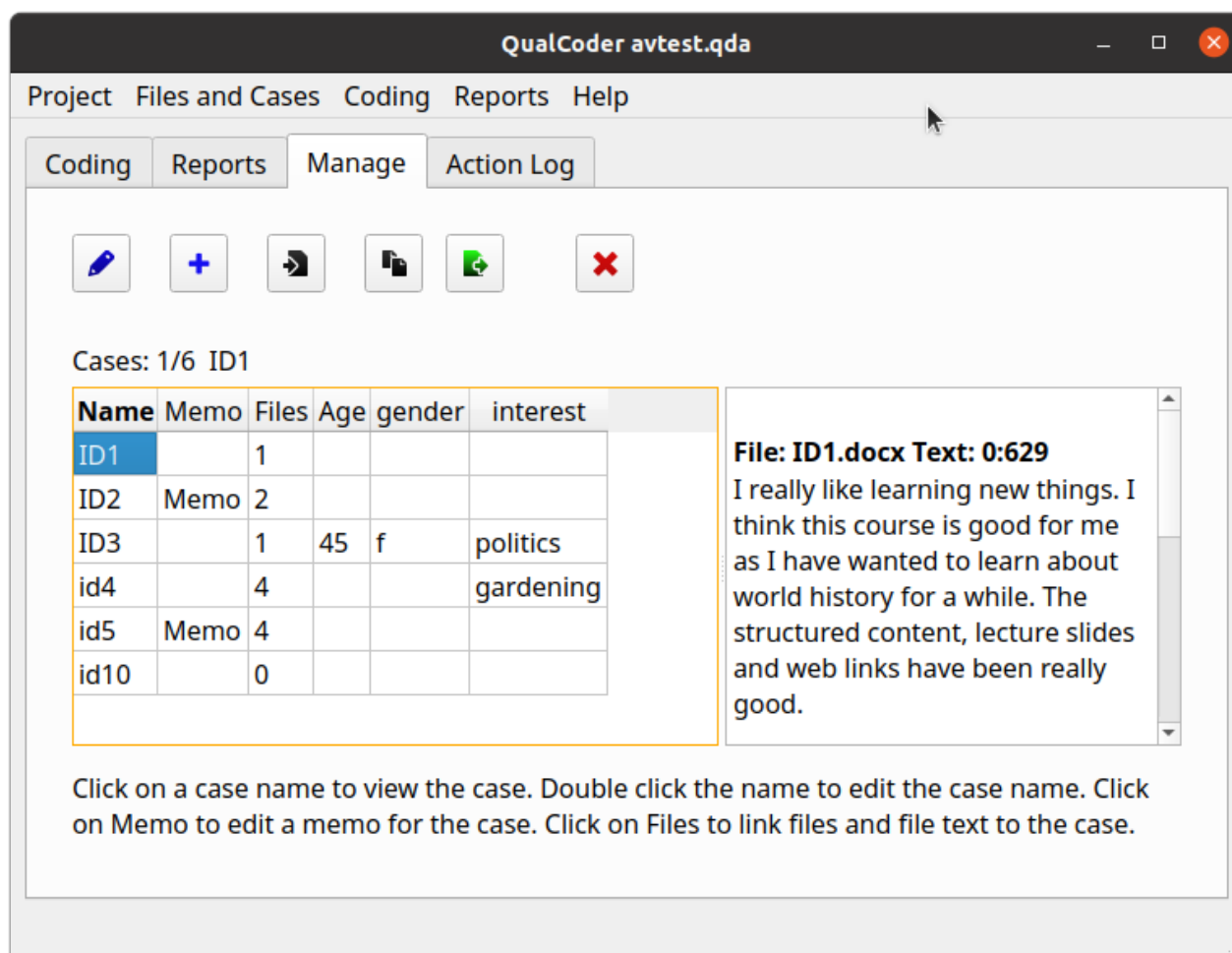


Figure 8: Case management

### Case file manager

The case file manager links all file types to each case.

For example, in the *case file manager*, click on a case, say ID3, then click *Add selected files to case* button. Select one or more files in the files list. Add the file(s) to the case. For example add the miguel-henriques.jpg to ID3. You can remove files and view the files associated with the case. You will see file text will be underlined in red which indicates this text is associated with this case.

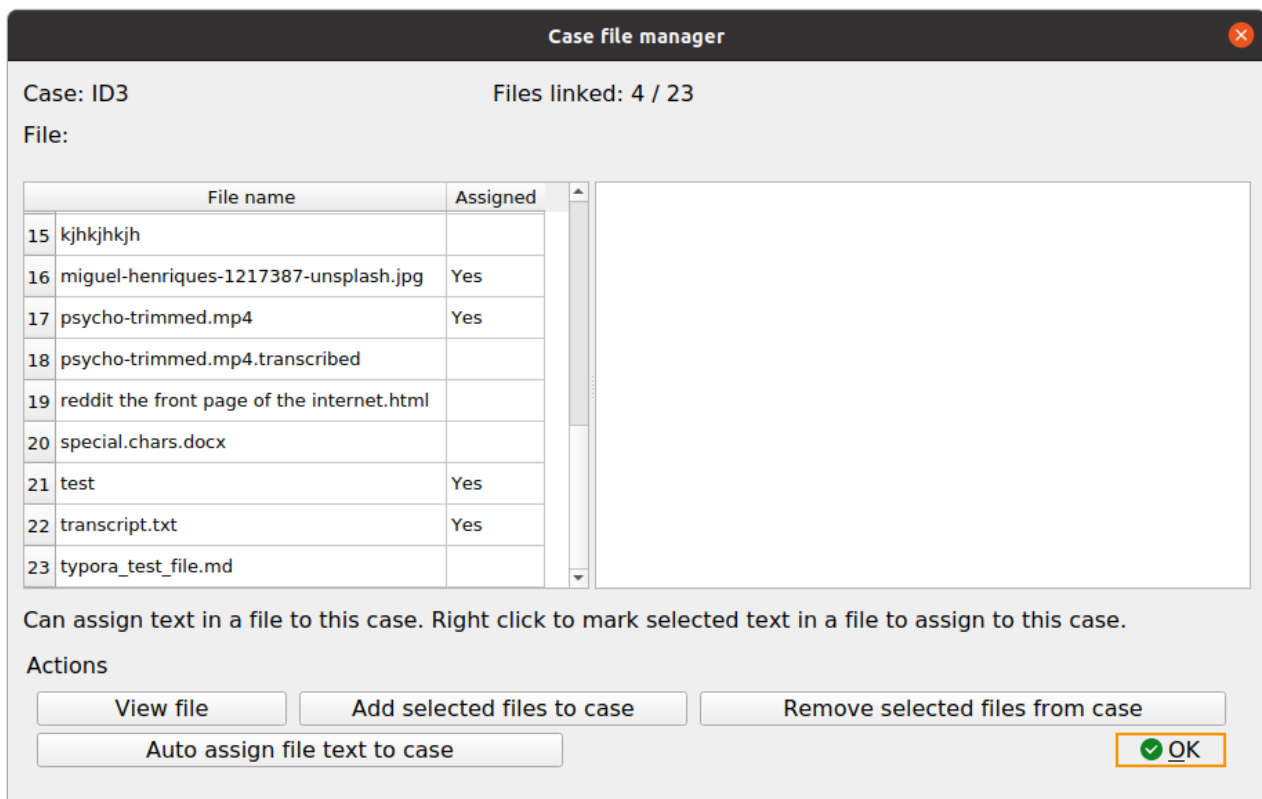


Figure 9: Case files manager

You can select sections of a text file to assign to a case. For example, paragraphs within a focus group transcript. You can also select a case text file, manually highlight text, and right-click to mark (or assign) the text to the case. You can also automatically assign sections of text to a case shown in an example below.

Now open the transcript.txt file. Notice the transcript begins with a student id inside square brackets [ ]. Try automatically assigning text to ID1 by selecting case ID1 then clicking the *Auto assign file text* button. You will be asked which file or files to assign the case to. Select *transcript.txt* from the list. Next you need to enter the start and end marks. The start mark will be '[id1]' and the end mark will be '[', note this is case sensitive so that is why you use the lower case here. Now select the case ID1 and see the assigned text.

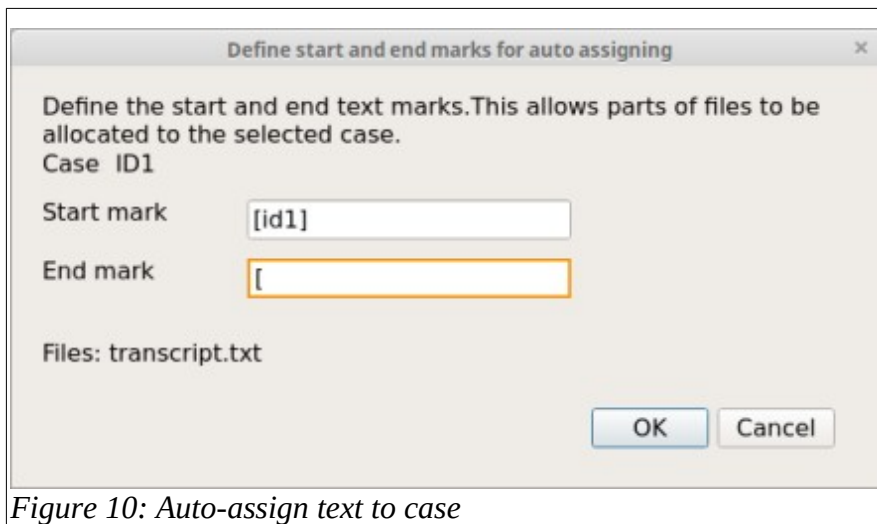


Figure 10: Auto-assign text to case

## Case attributes

Importing attributes for the cases. You can import attributes from a csv or Microsoft excel file. The first row must contain the attribute headings. The first column must contain matching case names (for this example: ID1, ID2, ID3). Note that this is case sensitive. Open the Examples folder has a cases.csv and cases.xlsx file.

You will then need to link any files and file text to each case using the Case-File manager..

### 4.3.6 Attributes

Attributes are variables associated with files or cases. They can be useful to add context to the text analysis. Open the *Manage Attributes* dialog. You can add, delete, rename and add memo notes to attributes. Attributes are stored as text, but numeric attributes are generally interpreted as numbers. Right click on a numeric variable and you can change it to character.

However, if you are running SQL queries you may need to cast the numeric attribute to an Integer or Real: `cast(my_var as integer)`.

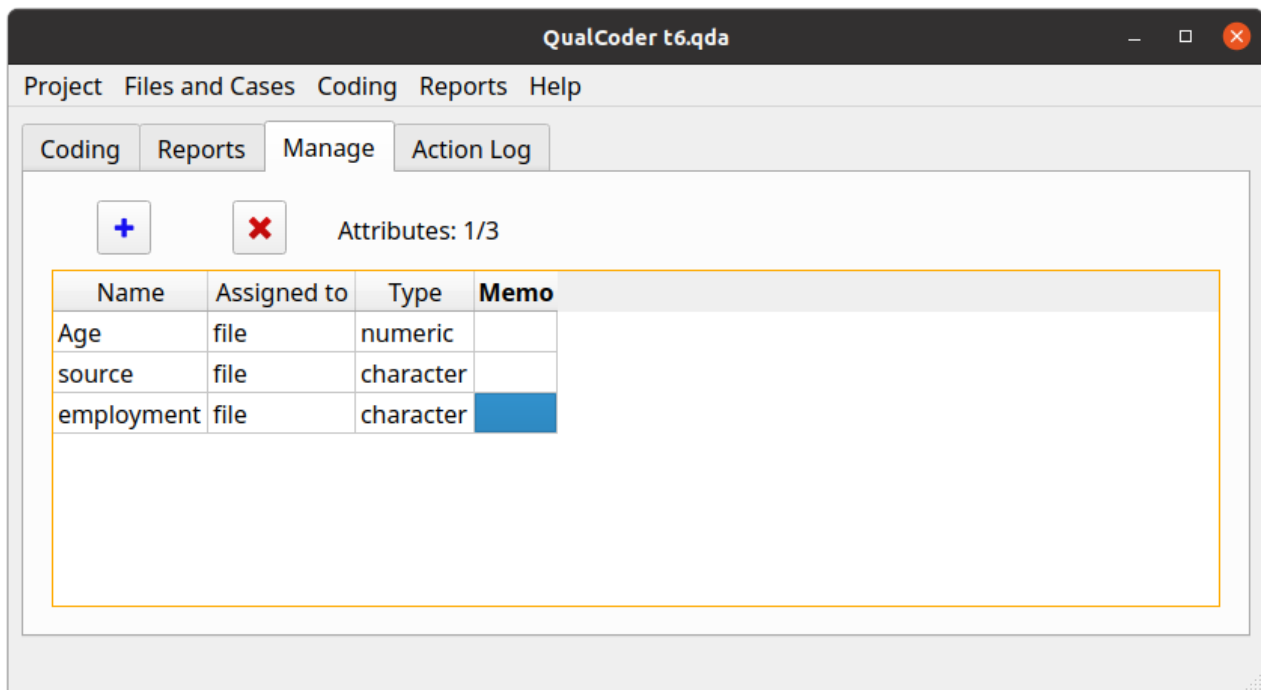


Figure 11: Attribute management

It is not the intention for QualCoder to perform statistical analyses of the attributes. It would be preferable to use dedicated statistical software such as R (<https://www.r-project.org/>) or other such software.

#### 4.3.7 Journals

You can record your thoughts when coding your data in journals. Make use of these to develop ideas and themes.

Journals can be created, deleted, exported and re-named.

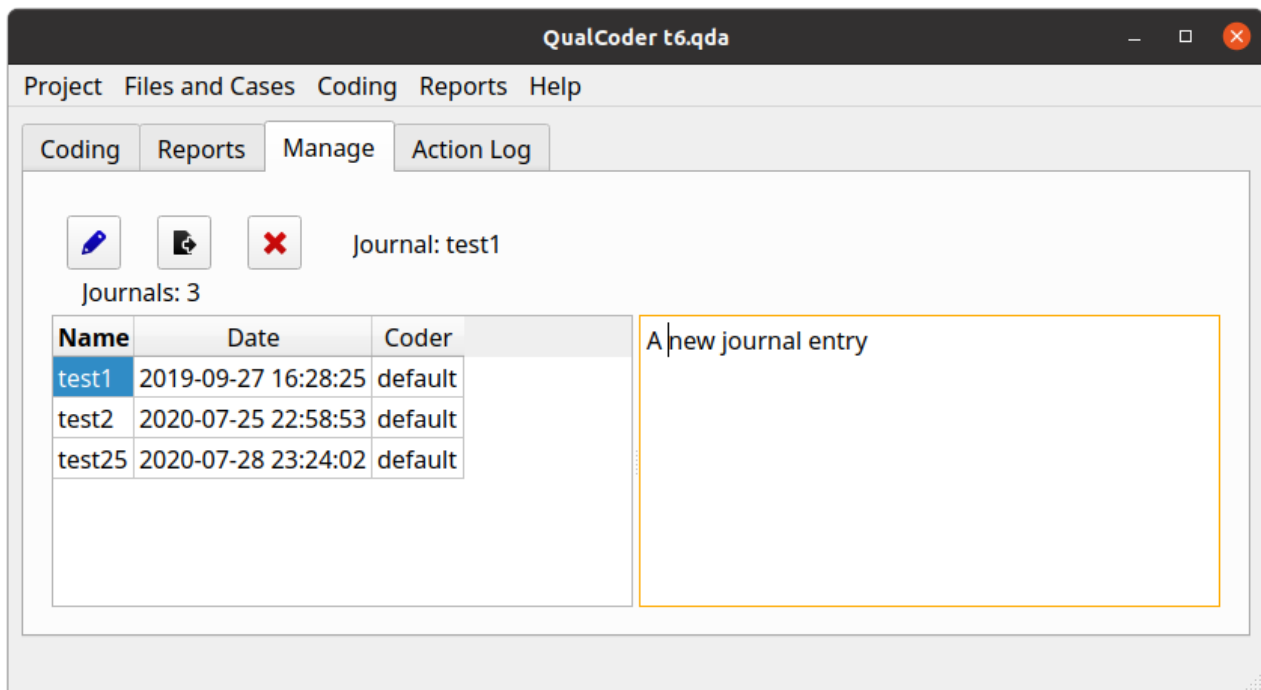


Figure 12: Journal management

### 4.3.8 Manage bad links to files

If your project contains externally linked files, these files can be moved, renamed, or deleted. The Mange bad links window allows you to edit the existing link and replace it with a new one, by finding the correct file and it's location. Bad links do affect images, audio and video files. Text is not impacted unless you need to review the original text document. This is because plain text is imported into the QualCoder database.

If you are importing an REFI-QDA project with external links, you will need to update the links to files using this function.

## 4.4 Coding menu

### 4.4.1 Categories and codes

Categories are used to organise codes. Categories are organised hierarchically in a tree structure. You can move codes into categories and move categories into larger categories. You can move categories and codes out of their current position. Codes and categories can be merged by dropping a code onto a code or a category onto a category. Categories and codes can be assigned memos. Right-click on a category or code to rename. Right click on a code to change the color or to move the code to another category.

To reduce the number of codes shown in the code tree. In the coding windows (code text, code, a/v or code image), right-click on the code tree and select *Show codes like* from the menu. Then enter text in the text box. This will then only show codes that contain that text. Enter nothing into the text box and press OK to show all codes again.

In all coding panes, you can select a code and view the coded portions that match. The right-click menu option ‘Show coded files’ is to show coded text and media for the selected code. Clicking on the coded portion heading will open another window to show that coding in context of the original file.

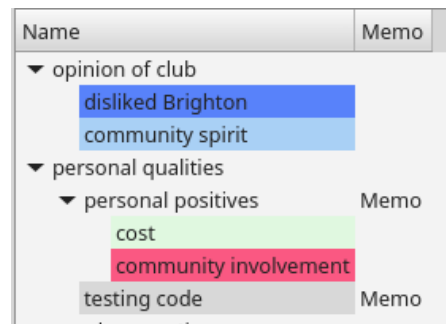


Figure 13: Categories and codes

#### 4.4.2 Coding text

Select *Code text* from the *Coding* menu. This is the central dialog for assigning codes to text. Once text segments are coded, hovering the mouse over the coding shows the code name as a tooltip. Clicking on the coded segment also shows the code name. Select a file in the list to open for coding. The file selection list contains buttons that allow you to move to the next file, to move to the most recently coded file and to go to a bookmarked position in a file. There is also a button to view the file memo.

Create a new code by right-clicking in the left hand window. Codes can be assigned a colour by right-clicking on the code and selecting the *change code colour* option. Other options from the right-click menu include adding a memo to the code, deleting the code, renaming, adding a new code, and adding a new category.

The easiest way to code text, is to select some text, then left-click with the mouse on a code. A second way is to select a code, then select some text. Right-click and mark the text to assign it to the selected code. Hover the mouse pointer over coded text to see a tooltip of the code. Coded text can be uncoded by clicking on the text segment and pressing the *Unmark* button. If a section of text is highlighted pressing the shortcut key ‘q’ will quickly code the selected text to the selected code. You can also pick from a recent code list after selecting text then pressing the shortcut key ‘r’.

#### Automatic coding

The top right has icons for auto-coding text. You can auto-code exact text, auto-code sentences based on a text fragment in the current file, and auto-code sentences based on a text fragment for all files. Automatic coding is case sensitive. You must also define the end of a sentence, there is a default setting with a period and space. For auto-coding exact text matches, multiple sections of text can be assigned by auto code using the pipe ‘|’ symbol. For example, *politics|politicians* can be

assigned to the same code at the same time (for exact auto code text matches only). There is an undo option to undo recently performed auto-coding. Although if the project is closed and reopened, the undo option will be lost.

## Searching the text

The coding text dialog contains a search for text function at the top middle, with tick boxes for searching case sensitive and for searching through all text files. When in the text area, selecting some text and pressing the shortcut key 's' will fill the search text box and focus on the 'next' arrow button for quickly looking through the document for the selected text. The search requires a minimum of 3 characters. The Case sensitive check box will limit to case sensitive searching. The All files check box will continue the search through other text files.

The search uses Regex functions.

- A dot '.' is used as a wild card, e.g. '.ears' will match 'bears' and 'years'.
- A '?' after a character will match one or none times that character, e.g. 'bears?' will match 'bear' and 'bears'
- A '\*' after a character will match zero or more times.
- '\.' will match the dot symbol, '\?' will match the question mark. '\n' will match the line ending symbol. This Regex cheatsheet might assist: [www.rexegg.com/regex-quickstart.html](http://www.rexegg.com/regex-quickstart.html)
- '\b' word boundary, '\bbound\b' will search for the full word 'bound', but not 'boundary'

## Displaying and moving through the text for only one selected code

Underneath the codes tree are several buttons.



*Figure 14: Search buttons for a selected code*

Select a code and then click on the forward and back arrows. These now traverse through the text document showing the selected coded text. All other coded text is hidden. To restore and show all the codes, click on the now coloured grid button.

## Overlapping codes

Overlapping codes can be difficult to view clearly. Overlaps have an overline above the text to show overlapping sections. Mouse hover will show coded text, including overlaps. Clicking in an overlap shows a selection box at the top of the screen where you can select to view one or the other



highlighted coded text. You can toggle through the overlapped codes using the shortcut key 'o'.

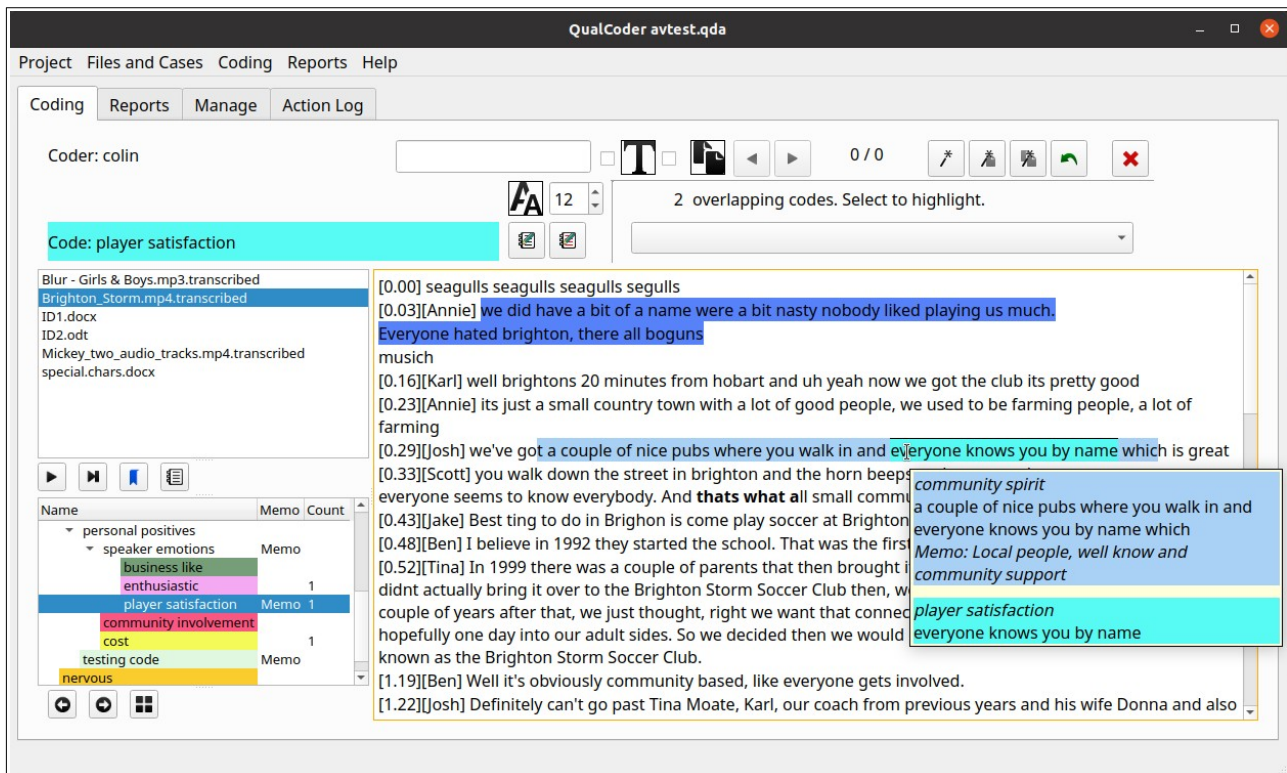


Figure 15: Coding text dialog showing overlapping codes identified with an overline

## Coded text memos

Coded text sections can contain their own memos. Either right-click on the area and select memo from the context menu, or click in a coded text section and click the pencil and notepad icon on the right, or press the 'm' shortcut key. Memos will be displayed in the tooltips.

## Annotations

Add an annotation (like a memo for a text segment) to a text selection. The text will become **bold** to mark the position of the annotation. To re-open an annotation, select some of the bolded-text and right-click to get the Annotate option. Alternatively, click the notepad and pencil icon on the left. If a section of text is highlighted pressing the shortcut key 'a' will annotate.

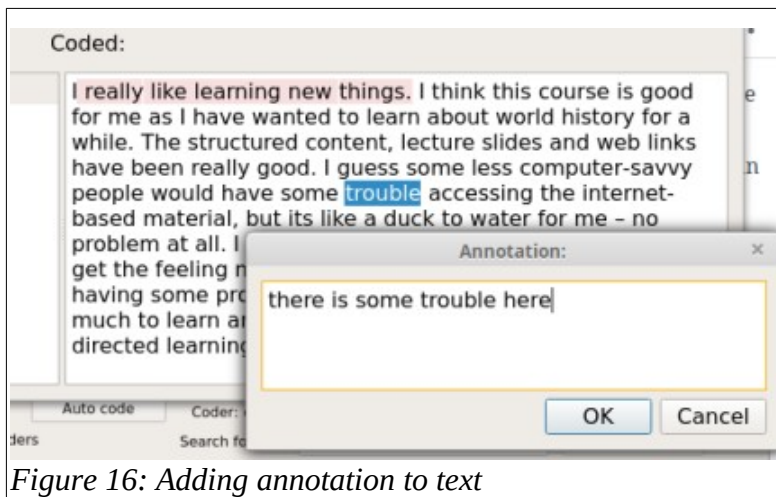


Figure 16: Adding annotation to text

## Modifying code positions

When in the text area, click on a code with the mouse (Note the code must not be overlapping with another code at that position). Press the following key combinations to extend or shrink the coded text segment.

Shift + left arrow	Extends coded text to the left
Shift + right arrow	Extends coded text to the right
Alt + left arrow	Shrinks coded text from the right hand side towards the left
Alt + right arrow	Shrinks coded text from the left hand side towards the right

You can also right-click on a code and select *change start position* or *change end position* by a number of characters.

## Bookmark

You can book mark a position in the current text and current file, by using the right click menu or by pressing the shortcut key 'b'.

## Maximise screen area

To hide the top control panel, click in the text area and press the 'h' key. To show the top panel, press 'h' again.

### 4.4.3 Coding images

Images can be coded in a similar way to text coding. Select a code. Left-click and drag to highlight the area you want to assign to the code. Right-click will open a menu where you can remove the coding or add a memo. At the bottom of the screen there is a slider control to re-size the image. Coded rectangles are coloured to match the code colour.

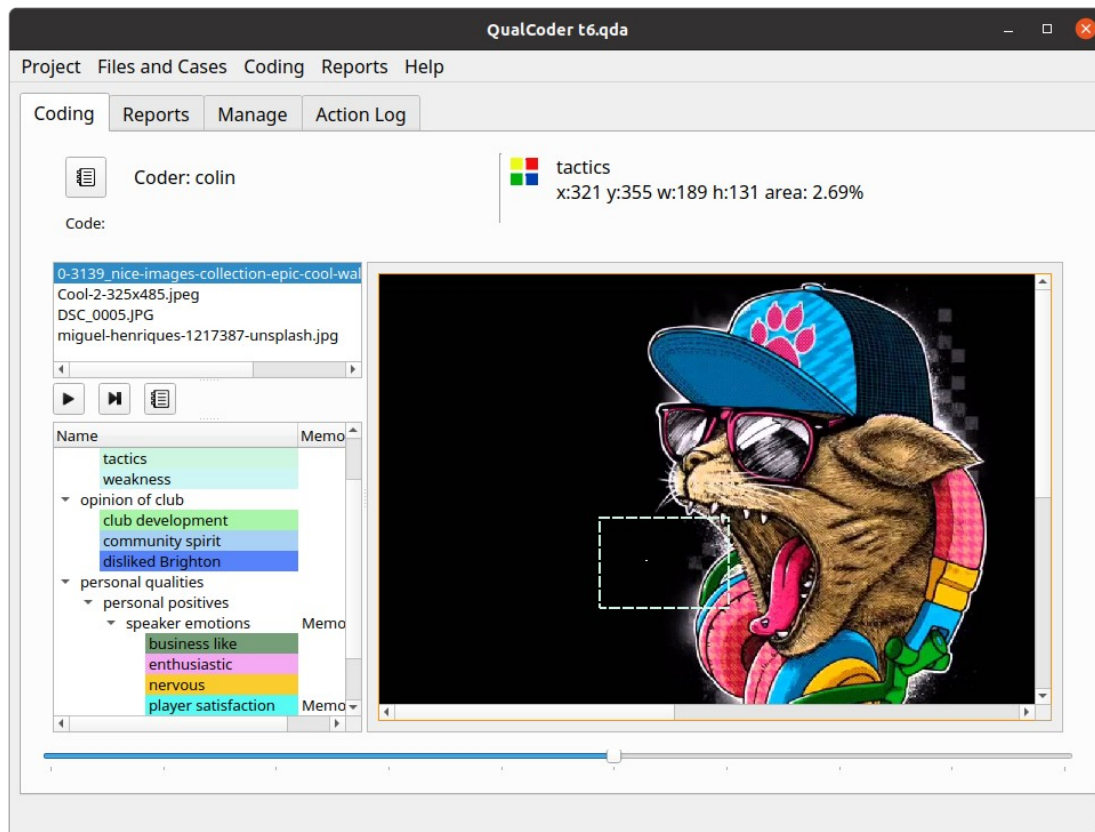


Figure 17: Image coding screen

## Maximise screen area

To hide the top control panel, click in the image area and press the 'h' key. To show the top panel, press 'h' again.

### 4.4.4 Coding audio and video

When coding audio or video, two windows are displayed, shown below. One window has the audio or video playing. The other window has the controls and a button to begin and end a coded segment. The controls window shows the codes in the bottom left pane and the .transcribed file text is shown in the bottom right pane. The transcription text can also be coded and annotated in this window. Right-clicking on a timestamp will give you a menu option to go to that section of the video. If a video has multiple audio tracks, you can change the audio track too. Checking the scroll checkbox allows the transcript to scroll in time with the video, based on detected timestamps, however, you need to uncheck the scroll function to be able to code the transcript.

- Ctrl R and Alt R are shortcuts to rewind 5 or 30 seconds.
- Ctrl F will forward 30 seconds.
- Ctrl S and Ctrl P will stop/start, play/pause.

- Ctrl Shift < will slow play rate down to a minimum of 0.1.
- Ctrl Shift > will increase play rate up to two times.

There are icons that replicate the above functions also.

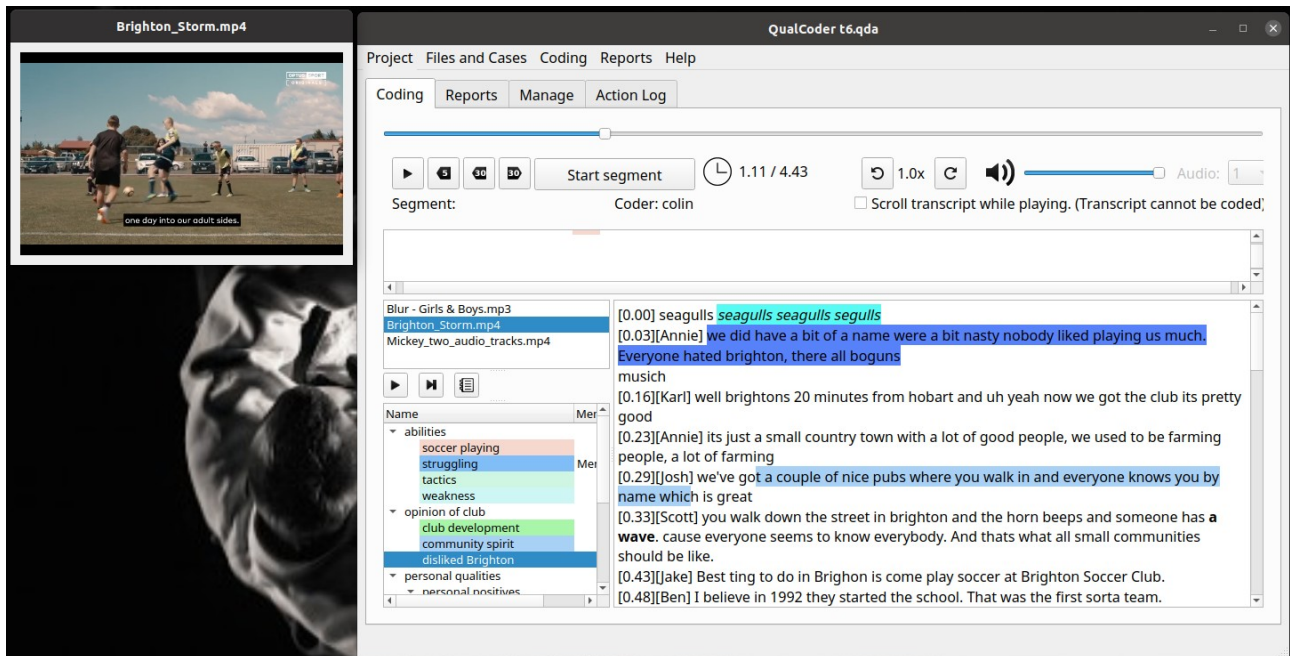


Figure 18: Audio video coding screen, showing coded transcript and one coded segment

## Audio Video segments

Once a segment is made using the *Start segment/Stop segment* button this can be assigned to a code by right clicking on the relevant code and assigning the segment.

Coding stripes for the coded segments are shown in the upper pane. Hovering the mouse over each stripe shows the code name, time segment and any memo attached to that coded segment. Right clicking on a coded segment stripe shows a menu that can be used to edit the memo or delete the coded segment or play from this point. Coded stripes are shown on various lines so that they do not overlap.

The transcript text can also be coded and annotated in this dialog. However, when the Scroll transcript check box is checked, this cannot be performed. Playing the video when this is check will scroll the transcript using the timestamps in time with the video.

## Linking text to coded segments

There are two ways to link text and coded segments.

One way is to select text (it can be coded or preferably uncoded) then assign it to a segment. This is done in two steps shown in Figure 19. First select the text and right-click *Prepare text link to*

segment. Then right-click on the receiving segment, choose *Link text to segment*, to assign the text.

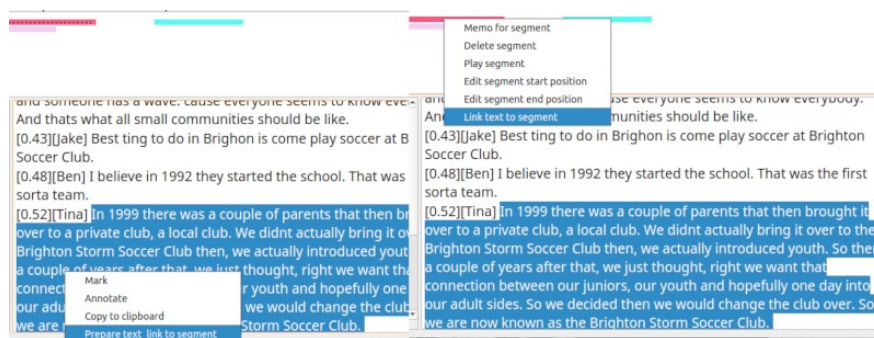


Figure 19: Assign a text selection to a segment

The second way is to select a segment and link it to a text selection via a similar process.

### Modifying code positions in the text area

When in the text area, click on a code with the mouse (Note the code must not be overlapping with another code at that position). Press the following key combinations to extend or shrink the coded text segment.

- |                     |                                                              |
|---------------------|--------------------------------------------------------------|
| Shift + left arrow  | Extends coded text to the left                               |
| Shift + right arrow | Extends coded text to the right                              |
| Alt + left arrow    | Shrinks coded text from the right hand side towards the left |
| Alt + right arrow   | Shrinks coded text from the left hand side towards the right |

You can also right-click on a code and select *change start position* or *change end position* by a number of characters.

### Other key shortcuts in the text area

- A annotate - for current text selection
- M memo code - at clicked position
- O Cycle through overlapping codes - at clicked position
- Q Quick mark with selected code - for current text selection
- R opens a context menu for recently used codes for marking text – for current text selection

### The video window

On the video window you can right-click and have options to change the window size, in pixels. You can also export a screenshot. The screenshot will be save as *Frame\_yyyymmdd\_hh\_mm\_ss.jpg* in the directory listed in the Settings.



## 4.5 Reports

### 4.5.1 Coding Reports

This dialog gives a list of coded text based on your selections. One or more codes need to be selected and one or both of *coder 1* and *coder 2* need to be selected. The *Search* button will present the results. If a category is selected all codes in that category are also selected. Multiple categories can be selected using the mouse and the Shift or Ctrl button. Codes and categories can be sorted to help find the codes of interest.

Codings can be narrowed down by using the *File selection* *Case selection* or *Attribute selection* buttons. Only codings within the selected files or cases will be presented. If text is entered into the *Search text* field, only codings which contain the matching text will be presented. Codings can also be narrowed down to case selections using the *Case selection* button.

The *Search Text* box also limits the reported codings. The search text looks for any codings in text that contain the search text. The search text also looks for matching text in memos for coded image areas and coded audio/video segments. The search text function must be used in combination with *File selection*, *Case selection* or *Attribute selection*.

The *Attribute selection* button opens a dialog window where you can select attributes for files and cases, for example: only interview files for people aged > 60.

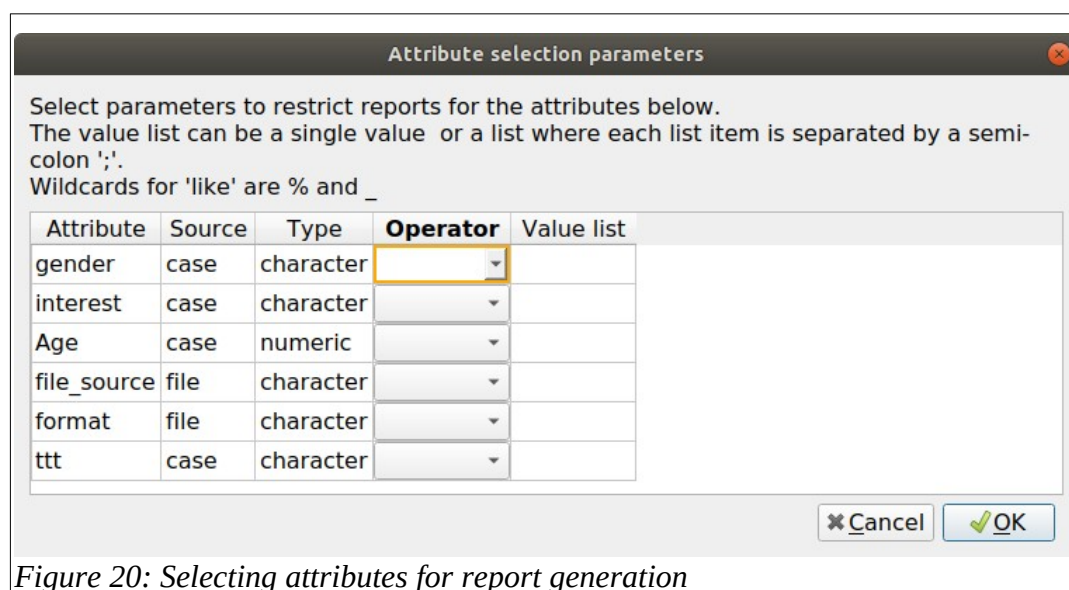


Figure 20: Selecting attributes for report generation

If you are creating a report based on Case selection, then three panes are shown. The middle pane shows the codings in a list, the right pane shows a matrix with one case on each row, see below image. The columns are either: top-level categories, categories with the immediate codes, or codes. These options are selected in the case matrix options combobox, shown as below:

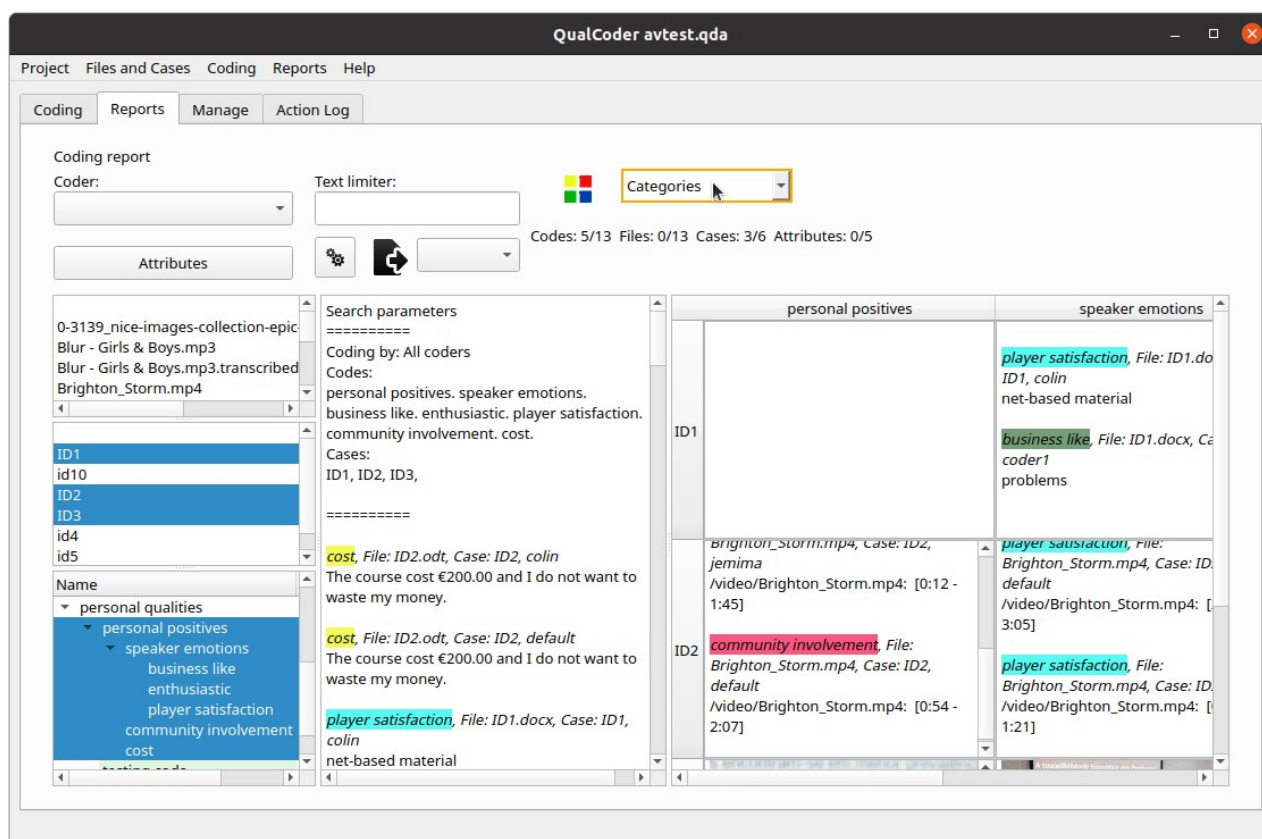


Figure 21: Case based search showing the categories to cases matrix.

Each of the three panes can be widened or narrowed by dragging with the mouse on the bar that splits each section.

## Exporting reports

Reports can be exported to text, open document or html files. HTML files are provided as the main html file and a supporting folder which provides, images and audio or video media. Reports can also be exported as a csv file, where each column is a code.

Currently case matrices are not able to be exported.

### 4.5.2 Node graph

A graph of categories and codes is displayed. There are several options to change what is shown – such as Black and White, All or selected categories. There are two display styles – list view which is ordered with categories and codes or circular view. The circular view fans the codes and categories out, but you may need to move some around as they can overlap if you have many.

Each code or category can be moved around by clicking near the edge of the box and dragging it around. Clicking on the text allows you to temporarily change the code or category wording. Right-clicking in a code gives a menu of options such as displaying the memo, or displaying all the case or file text that has been coded with the selected code.

Right-clicking on a line allows you to change the thickness, change to dotted line style or change the colour to red. This might be useful for emphasis.

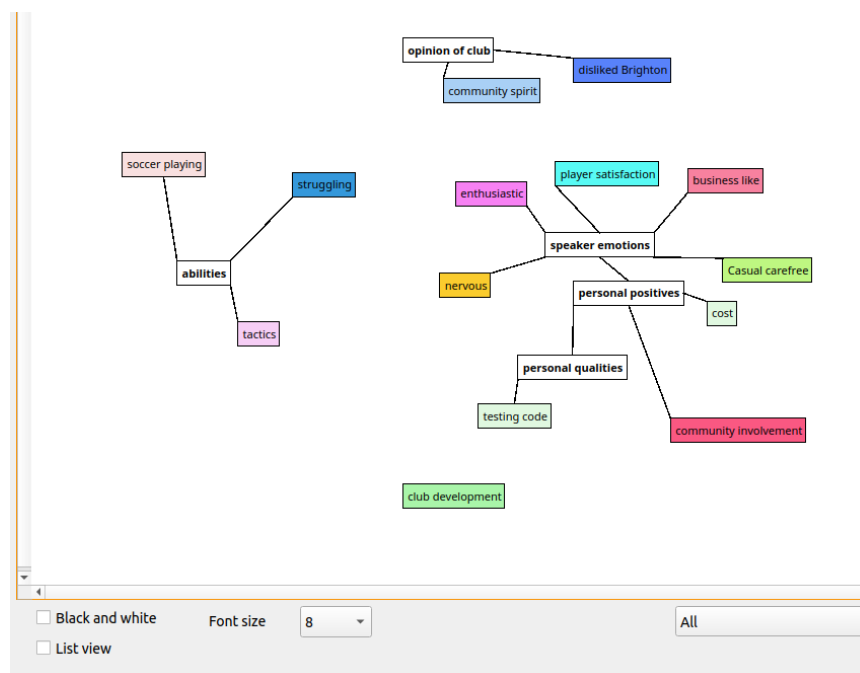


Figure 22: Circular view of categories and codes, after adjusting positions

### 4.5.3 Coding Comparison

This option shows the similarities and differences between two coders. Select two coders and click the *run comparisons* button. Coder comparison is only available for coded text, not coded media files. For each code:

Agreement % shows agreement for a combination of coded and non-coded text characters.

A and B % shows agreement for the only the coded text characters divided by the total characters in the text.

Not A and Not B % shows the total of the uncoded text divided by the total characters in the text.

Disagree % shows the percentage of all the coded and non-coded text that did not match between coders. It is the same as  $100 - \text{the Agree \%}$ .

Cohen's Kappa is calculated based on the information in Wikipedia

[https://en.wikipedia.org/wiki/Cohen%27s\\_kappa](https://en.wikipedia.org/wiki/Cohen%27s_kappa)



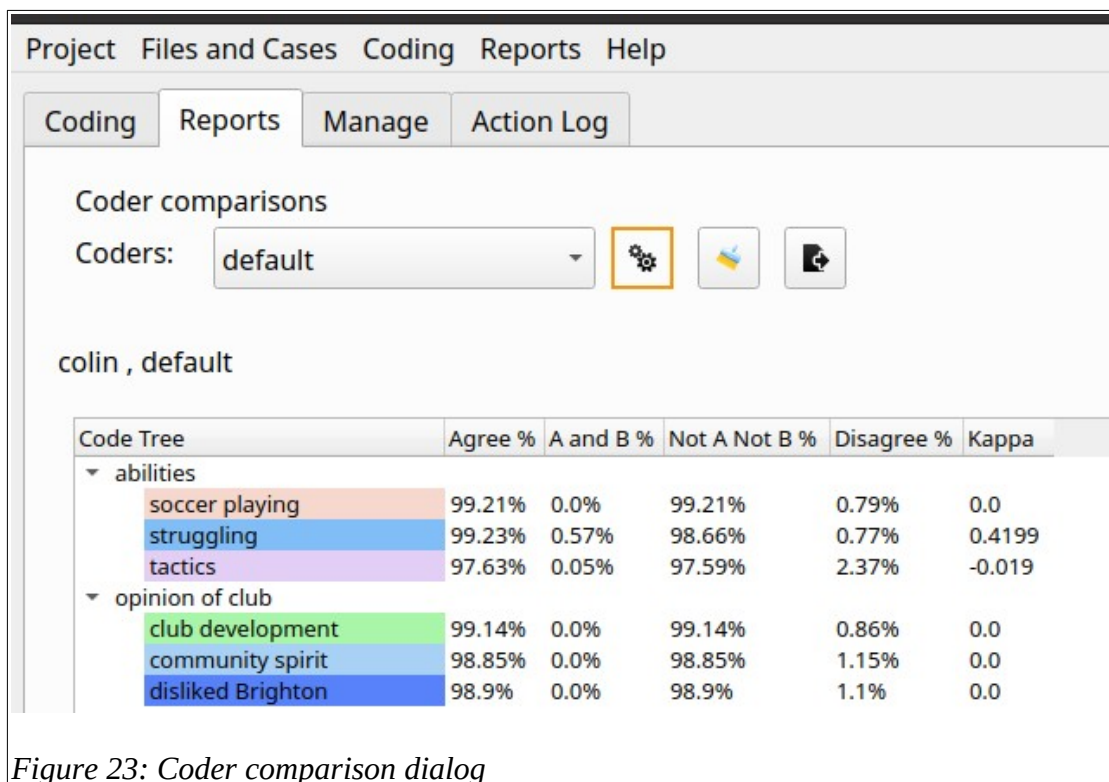


Figure 23: Coder comparison dialog

#### 4.5.4 Code relations

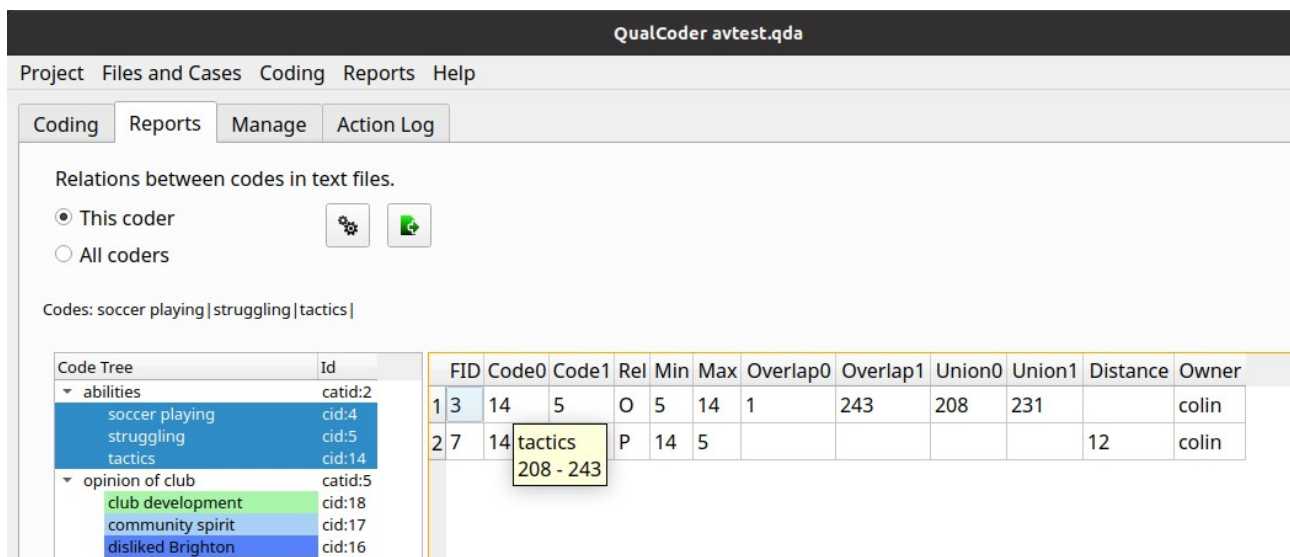
This dialog show the relations between two or more codes. This can only be applied to text files. A csv file can be exported.

The relations are:

- Proximity – Two codes do not overlap. The distance in number of characters is shown.
- Overlap – Two codes partially overlap. The lowest and highest character positions of the combination are shown. The union of the overlapping section is shown in character positions.
- Inclusion – One code is included within another code. The lowest and highest character positions of the combination are shown. The union of the overlapping section is shown in character positions.
- Exact – Both codes match in their start and end positions. The lowest and highest character positions of the combination are shown. The union of the overlapping section is shown in character positions.

Hover the mouse pointer over the code id (Code0 and Code1) (in the example code id 5 is tactics) to get the code name and its start and end positions. FID is the file ID, hover pointer for details. Rel is the relationship between the codes. Min is the earlier positioned code. Max is the higher positioned code. Overlap0 is the lowest character position of the combined overlapping codes. Overlap1 is the highest character position of the combined overlapping codes. Union0 is the character position

where the overlap begins, and Union1 is the highest character position where the overlap ends.



The screenshot shows the QualCoder interface with the title 'QualCoder avtest.qda'. The 'Coding' tab is active. Under 'Relations between codes in text files.', the radio button 'This coder' is selected. The codes 'soccer playing | struggling | tactics' are entered. On the left, a 'Code Tree' shows a hierarchy: 'abilities' (catid:2) containing 'soccer playing' (cid:4), 'struggling' (cid:5), and 'tactics' (cid:14); and 'opinion of club' (catid:5) containing 'club development' (cid:18), 'community spirit' (cid:17), and 'disliked Brighton' (cid:16). The 'tactics' code is highlighted. On the right, a table displays the relationship between 'soccer playing' (Code0) and 'tactics' (Code1).

	FID	Code0	Code1	Rel	Min	Max	Overlap0	Overlap1	Union0	Union1	Distance	Owner
1	3	14	5	O	5	14	1	243	208	231		colin
2	7	14	tactics	P	14	5					12	colin

A tooltip for the 'tactics' code in row 2 shows the range '208 - 243'.

Figure 24: Code relations

#### 4.5.5 Code frequencies

Code frequencies can be generated for all or selected files. The frequencies are shown by coder and as totals. In this example there were multiple coders.

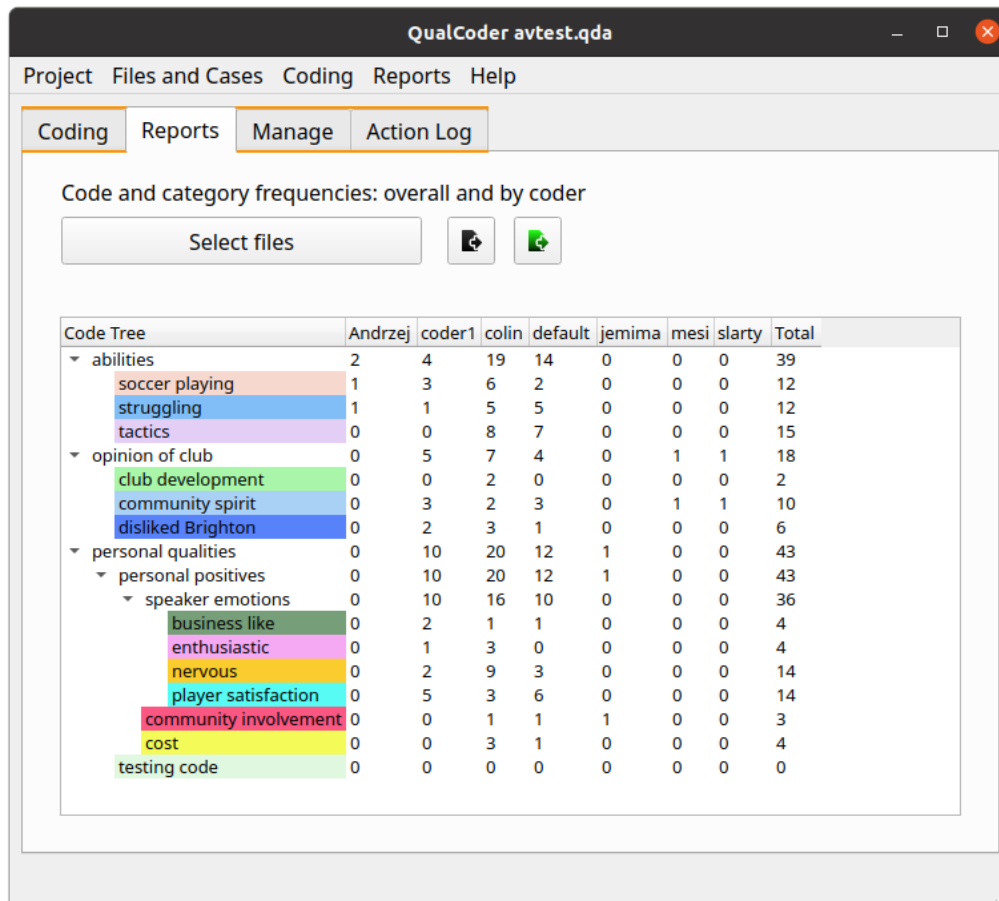


Figure 25: Code frequencies

#### 4.5.6 SQL Statements

This dialog contains three panes. The top pane is where SQL statements are entered and the bottom pane contains the results of queries. The left pane contains tables and field names. Double-clicking on a field name adds it to the SQL statement.

Results can be exported to a delimited file.

If you are not familiar with SQL take care as you will be able to update and delete the data as well as select data. Note: Some Unicode symbols are not converted to plain text and are ignored.

Most table fields are text. The following fields are integer: *anid*, *avid*, *attrid*, *caseid*, *catid*, *cid*, *fid*, *id*, *imid*, *jid*, *pos0*, *pos1*, *x1*, *y1*, *width*, *height*.

There are several prepared join statements listed:

- Case text
- Codes fileid and coded text
- Coded text with each case

- Get coding table – an implementation of the RQDA function of the same name

Most table fields are text. The following fields are integer: anid, avid, attrid, caseid,catid, cid, fid, id, imid, jid, pos0,pos1, x1, y1, width, height.

Right-click in the results table for filtering options.

Right click in the SQL window give you various options such as select all, copy, paste.

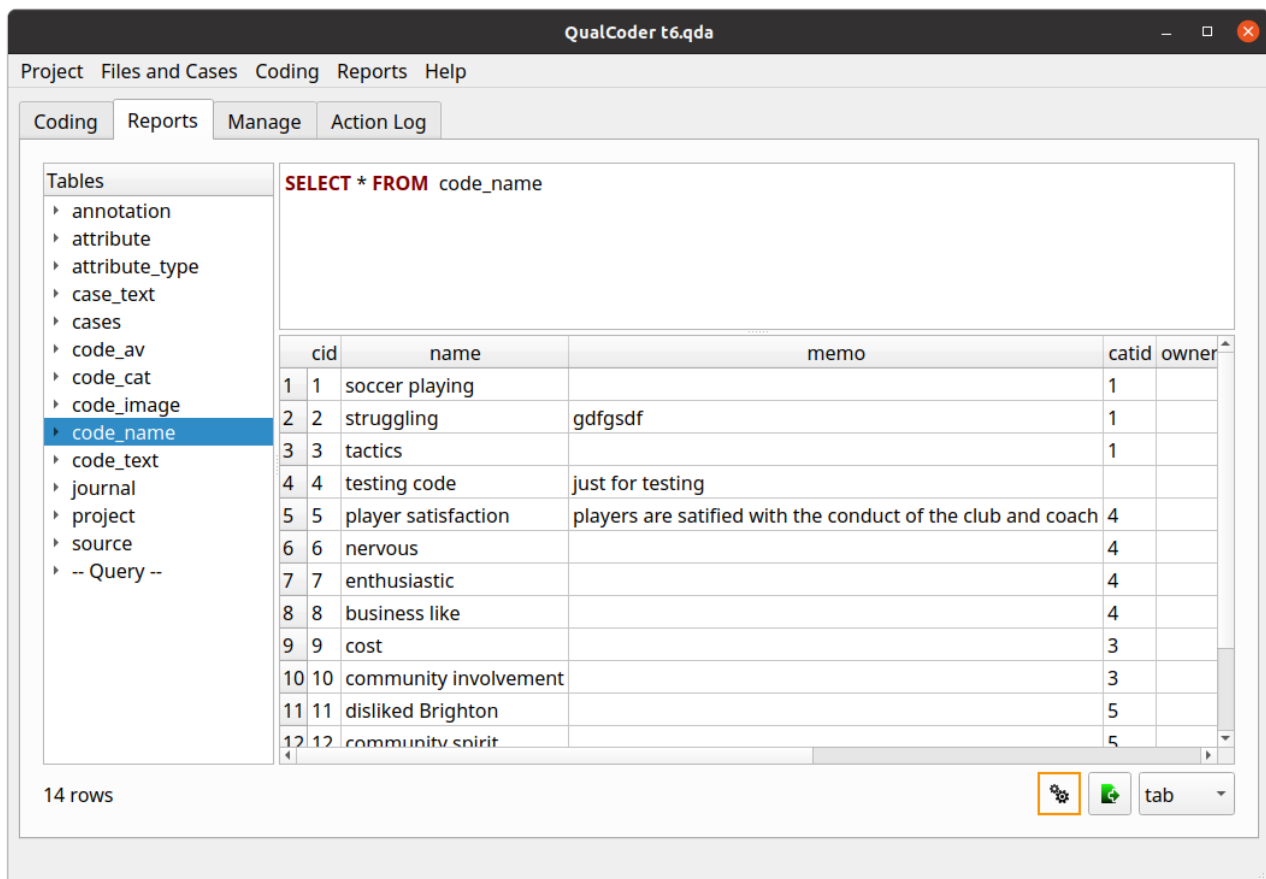


Figure 26: SQL dialog showing database tables and fields , SQL entry and results

## 5 Imports and exports

### 5.1 Import survey

This requires a CSV or Excel (xlsx format) file. You can try importing the survey.csv file in the Examples folder.

Survey files in a CSV format, are comma delimited format. Another delimiter can be chosen. For tab-delimited files type *ta*, *tb*, or *tab* in the Delimiter box so that QualCoder knows the csv file is tab delimited. The first row must contain the headings for attributes. The first column must contain the unique identifiers for each survey respondent.

QualCoder will determine if the other columns (attributes) are Numeric or Character. QualCoder cannot determine if an attribute is qualitative data. You must right-click and change the field type from *character* to *qualitative* for those fields that need to be qualitative.

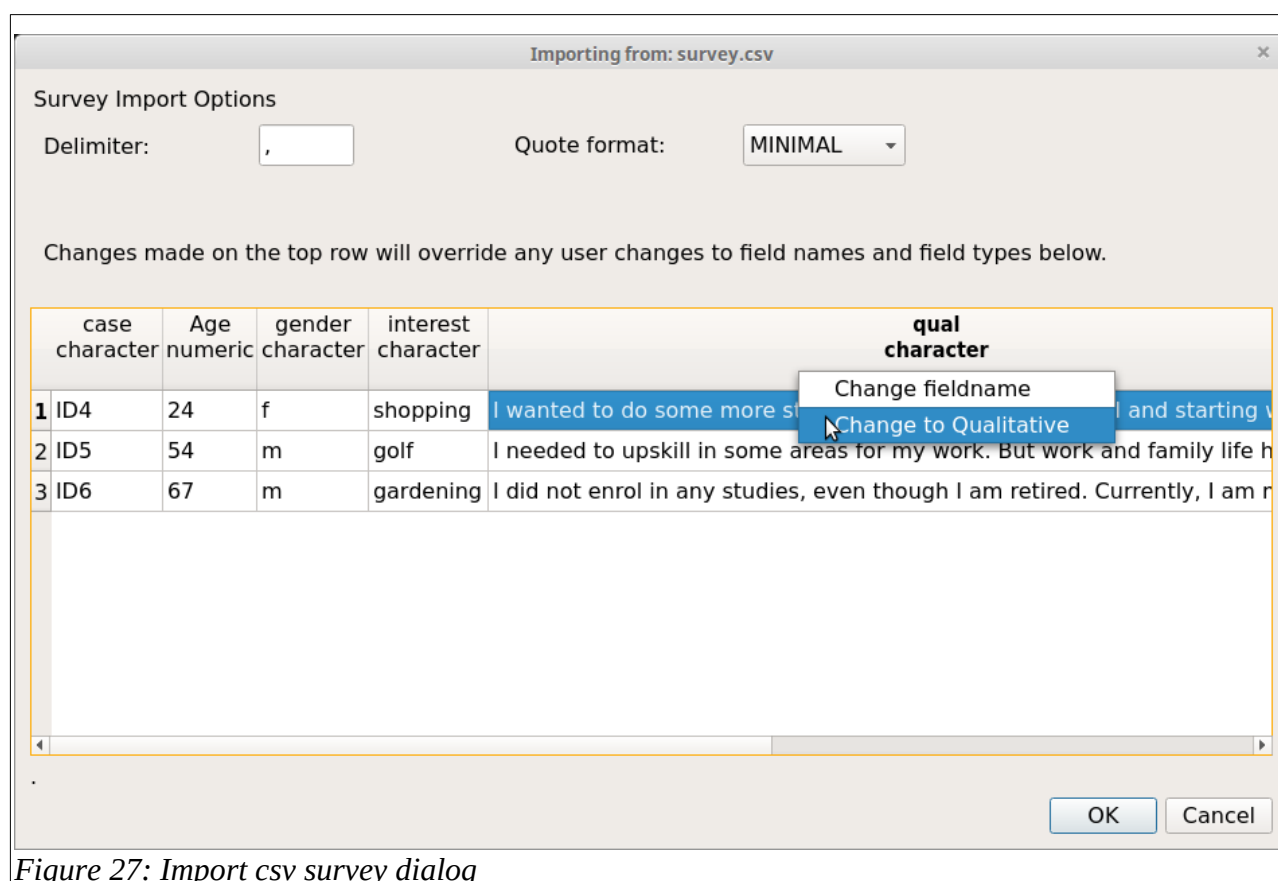


Figure 27: Import csv survey dialog

When you click on the OK button, the cases and their attributes will be added. The qualitative column will be converted to a file which will be named with the column name plus the current date and time. Each respondent's row will be prepended with [the unique id] so that you can identify each respondent. Also, the text for each respondent will be automatically linked to the corresponding case.

## 5.2 REFI-QDA

The Rotterdam Exchange Format Initiative (REFI) is an open standard for exporting and importing codebooks and projects from and to different computer-aided qualitative data analysis software. The website for the exchange initiative is [www.qdasoftware.org](http://www.qdasoftware.org)

Currently, QualCoder is able to export and import a codebook.

Exportation of a project and importation of a project are experimental and do not meet the full REFI-QDA standard. Some experimentation has been performed with Nvivo, Quirkos, Atlas.ti and MAXQDA qdpx files. Importation of project.qdpx files mostly works. For audio and video, importation of transcriptions and syncpoints is untested. Importation/linking of relative linked files has not been tested.

Export from QualCoder and import into some other software can change code positions due to differences in how line endings are managed by the importing software (MAXQDA, Atlas.ti).

Features such as sets and graphs cannot be imported as this functionality is not within QualCoder. Other data formats are not preserved in the original format, for example attributes with date, integer, float or boolean would be converted to text data.

To import, close any currently opened project first. Then choose a new QualCoder project name, then select the qdpx import file.

If no coded data is visible, you may need to change the current coder's name in Settings. Try the drop down box.

## 5.3 RQDA

Projects made with RQDA (<http://rqda.r-forge.r-project.org/>) can be directly imported. All data except for file categories are imported.

Close any opened project. Then in the Main Menu under Project click on RQDA Project import. You will be asked to create a new project - so enter a new project name. Then you will be asked to select the RQDA project file. QualCoder will then import the data.

If no coded data is visible, you may need to change the current coder's name in Settings.

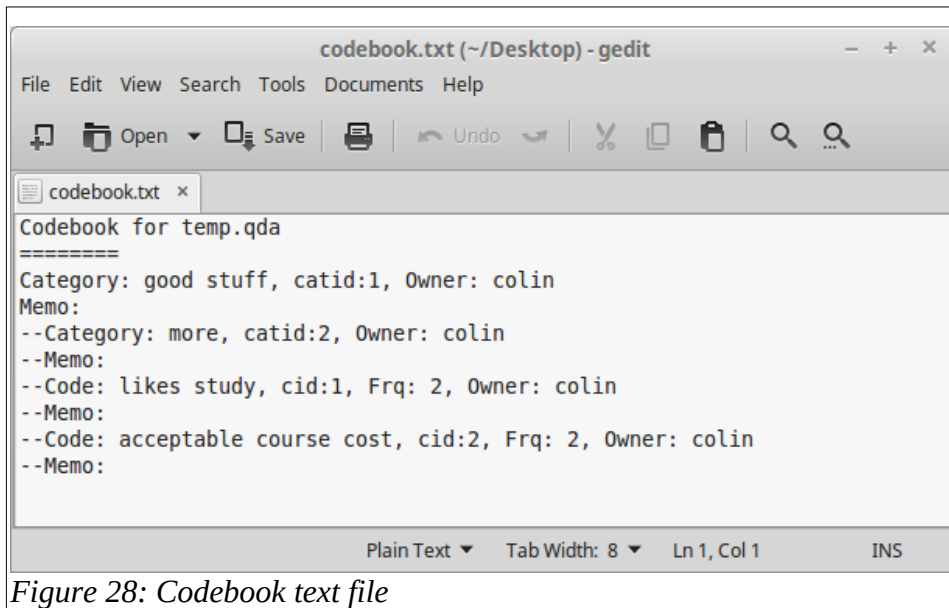
## 5.4 Using the project other operating systems

There are differences in line endings between MacOS, Windows and Linux operating systems. QualCoder will open the same project on the above different operating systems, differences in how python interprets line endings may affect text coding positions by one character.

## 5.5 Codebook

A codebook is a list of your codes. Each code memo should detail the reasons for the purpose of

each code. The codebook can be exported to a text file. The codebook also shows the frequency of the codes used (from all coders). Example codebook output is shown below. As codes and categories are put in a tree like structure the double minus '--' indicates the subordinate codes and categories within a category.



```
codebook.txt (~/Desktop) - gedit
File Edit View Search Tools Documents Help
[Icons] Open Save [Icons] Undo [Icons] [Icons] [Icons]
codebook.txt x
Codebook for temp.qda
=====
Category: good stuff, catid:1, Owner: colin
Memo:
--Category: more, catid:2, Owner: colin
--Memo:
--Code: likes study, cid:1, Frq: 2, Owner: colin
--Memo:
--Code: acceptable course cost, cid:2, Frq: 2, Owner: colin
--Memo:
Plain Text Tab Width: 8 Ln 1, Col 1 INS
```

Figure 28: Codebook text file

## 6 Other details about QualCoder

---

The qda data folder contains folders for imported documents, images, audio and video. It also contains the sqlite database, named data.qda, to store coding data.

QualCoder creates QualCoder.log and config.ini and recent\_projects.txt files inside a .qualcoder folder in your home directory. The config.ini file contains the name of the current coder, a default working directory, font choice and language. The log file records program errors and some user actions. There may be several rolling log files.

QualCoder is written in python 3 using Qt5 for the graphical interface. The minimum python version is 3.6. The minimum recommended screen size is 1024x600 pixels.

YouTube videos in Spanish are available for QualCoder 1.9, thanks to Omar Bautista:

<https://www.youtube.com/watch?v=D2vks2n9d1g>

[https://www.youtube.com/watch?v=cqAy\\_RJkhvY](https://www.youtube.com/watch?v=cqAy_RJkhvY)

### 6.1 Acknowledgements

Ronggui Huang and Zhang Gehao for creating RQDA, which inspired this software. Mike MacCana for the source code for the docx module. User: bit4 on stackoverflow who presented the source code to convert html to text. Pdfminer3k: Copyright (c) 2004-2010 Yusuke Shinyama <yusuke at cs dot nyu dot edu> ebooklib: Aleksandar Erkalović (<https://github.com/aerkalov>). The VideoLAN team for the bindings to VLC. To various members on github for supporting this project. Omar Bautista for creating YouTube videos.

### 6.2 Publications citing QualCoder

Local–global linkages: Challenges in organizing functional communities for ecosocial justice. Joel Izlar, Journal of Community Practice 27(3-4) 2019

Barriers to Health: Understanding the Barriers Faced by Community Intervention Projects. Vera Landrum, The University of Southern Mississippi 2020, Available from: [https://aquila.usm.edu/cgi/viewcontent.cgi?article=1772&context=masters\\_theses](https://aquila.usm.edu/cgi/viewcontent.cgi?article=1772&context=masters_theses)

Framing food geographies. S Ramsay, Masters Thesis, Stockholms Universitet 2020

Seeking research software. A qualitative study of humanities scholars' information practices. Ronny Gey, Masters Thesis, Humboldt University of Berlin 2020

Traditional and biomedical care pathways for mental well-being in rural Nepal. T Pham, R Koirala, B Kohrt, International Journal of Mental Health Systems volume 15 2021



## 7 Future plans

---

Currently, the primary testing environment is Ubuntu 20.04 and secondly Windows 10. QualCoder has been used on Ubuntu 19.04, Linux Mint 19.04, Lubuntu 18.04, Windows 10, Raspberry Pi and MacOS.

In Windows, reports exported in ODT format are okay, but some images may overlap when the ODT file opened with Microsoft Word rather than opened with LibreOffice. There are occasional issues in Windows with the software finding the VLC libvlc.dll file for audio/video work.

Some potential plans for the future are to add the following functionality:

General:

- Provide software in other languages for widgets and documentation. This is underway and some language options are available. Greek, Japanese and Spanish require the most revision work.
- Improve the REFI-QDA project import and export to make it standards compliant. This is gradually improving.
- Improve packaging for Windows 10, Mac OS.

Reports:

- Possibly look at text mining functionality, word clouds, word visualisations
- Look again at the python Igraph package. (This did not load on older OS's or older python versions)
- Possibly word counts
- Alternative ways to visualise codes

## 8 About the author

---

Hello, my name is Colin Curtain and I am a pharmacist and lecturer from Australia. I have many interests including clinical pharmacy, computer programming, research, statistics and clinical decision support. I completed a PhD evaluating computerised clinical decision support in 2014. When doing my PhD I used [R](#) as the statistics program of choice. This is where my interest in qualitative data analysis and the use of [RQDA](#) came from, which ultimately led to this project.

Originally when doing my PhD I analysed qualitative survey data via a thematic approach using RQDA. I then thought this could be reproduced in Python, so I scripted an earlier version called PyQDA which worked OK at the time.

I thought I would share QualCoder in the hope that it may help others. Bugs are possible and functionality could be further extended. I only work on the programming for this in my spare time.

If and when you use QualCoder and publish your results, I would really appreciate it if you let me know the bibliographic information of your work.