



Quick getting started tutorial – last updated: 01/06/12

The typical use of **ROC me!** involves the following steps:

1. [creating a corpus file](#)
2. [creating a project](#)
3. [creating a user](#)
4. [recording](#)

1) Creating a corpus file

A corpus file is a simple text file containing 2 kinds of information, separated by a tab, on each line:

- a file identifier
- the sentence to be read

The file identifier is followed by a tab which tells **ROC me!** that the characters after the tab should be displayed on the computer screen.

If you want to include instructions, replace the file identifier by //

Please note:

- don't forget the tab after //
- the file identifier will be included in the name of the sound file
- each sentence will be saved in a different sound file

Corpus files must be saved as UTF-8 text files. More information on corpus files can be found [here](#).

Advanced users: **ROC me!** corpus files are HTML-compliant; in other words, you can use (most) standard HTML tags to change the style of the display (colour, font size, etc.). More information and examples can be found [here](#). Sample corpus files are available from the **ROC me!** [website](#).

Example (replace [tab] by a real tab):

```
// [tab] This is an instruction  
01 [tab] This is sentence number 1  
02 [tab] This is sentence number 2
```

2) Creating a project

Once the program has been started, click **New Project**.

Fill in the form on the screen. Most field labels are explicit.

`Order` lets you choose whether the sentence should be displayed in the same order as in the corpus file (`Normal`) or in random order (`Shuffle`).

`Focus`: if `Yes` is checked, a mask with a fixation point appears before each stimulus; recording starts before the stimulus appears. This is useful when:

- you do not want participants to see the stimulus before the recording starts,
- you want to time when the stimulus appears,
- you want to make sure that the recording has already started when participants utter their first word.

Note that with the `Focus` option, sound files are automatically saved to disk when you stop the recording.

`Split`: if `No` is checked, each participant will see all stimuli. If `Yes` is checked each participant will see a predefined subset of the corpus ([more...](#)).

`Repeat` makes it possible to record several sound files for a single stimulus.

`Allow Clipping`: if `No` is checked, the user will not be allowed to save the current recording to disk if at least one sample has been clipped. If `Yes` is checked, saving clipped signals will be possible. Clipping is equivalent to saturation.

`Administrator Password` lets the investigator choose a password ([more...](#)).

The XML `Users` box is for advanced users. You may leave it as it is for the moment ([more...](#)).

`Show Clock`: a timer counts down the remaining time from a preset value. ([more...](#)).

`Background Color`: set the color of the box within which stimuli appear. Default is white.

`User information filled by`: if you choose `Users`, the latter will be prompted to fill in the questionnaire before the recordings.

`Rate Hz` is the sampling rate, `Bits` is the quantization rate (resolution, bit depth), and `Channels` lets you choose between `Mono` and `Stereo`. Unless you have serious reasons to change them, we recommend that you stick to default values for these 3 fields (custom audio settings have been disabled in Version 2.0 but are expected to be back very soon).

Once all fields have been filled in, save the project. Note that `Project Name`, `Working Directory`, `Order`, `Focus`, `Split`, and `Repeat` will no longer be editable.

3) Creating a user

Creating a new user (or loading existing users) presupposes that a project has been loaded. Click the `USERS` tab, and create a new user by clicking `New User`. Enter a suitable name, and click `Done`. The metadata form (if such a form has been linked to the project at project creation) will appear after clicking on the `Edit User` button. User metadata can be provided at any time.

4) Recording


Go to the `PLAYER` tab. This presupposes that a project and a user have been loaded. At the beginning of a recording session, you should make sure that




1. the right audio input is selected
2. the recording level is adequate

Once all hardware connections have been checked, let the user speak through the microphone and examine the behavior of the input gain meter in the top left-hand side of the screen.




In the figure above, the gain meter says “NO INPUT SIGNAL”. The input gain slider should be adjusted and/or the user should try to speak louder or closer to the microphone. If the “NO INPUT SIGNAL” warning remains, then it very probably means that the wrong audio input was selected.


If the corpus contains instructions, they will be displayed on the screen. Click  to move to the 1st sentence.


Click  to start recording, the button then changes to . Click  to stop recording.

The input gain meter continuously monitors the sound level: you should make sure the level meter remains green. If it becomes orange (clipping allowed) or red (clipping not allowed; this parameter can be set at project creation), it means that samples have been clipped (and the signal is therefore saturated). In this case, you might want to re-record the sentence, after doing one of the following:

- increase distance between mouth and microphone
- adjust the gain slider 

 lets users play back what they have just recorded.

 click this button to save the recording to disk and move on to the next sentence.

In some cases, users may want to split the recording into several sessions. ROC me! will automatically locate the last sound file recorded by the current user. If the corpus file starts with instructions,  will allow users to jump directly to the right sentence. This occurs in XP mode only ([learn more about Administrator vs. XP mode](#)).

Note: alternatively, you can press the space bar to start and end recording. Press Enter to save the current recording to disk and move on to the next sentence. The Tab key allows you to play back the current recording.

(Note: demo projects are available from [ROC me!](#) website.)

End of getting-started tutorial

Advanced functions - last updated: 01/06/12

1. [XML Users box](#)
2. [Corpus files](#)
3. [HTML formatting in the corpus file](#)
4. [Split mode](#)
5. [Other features and cool tips](#)

The XML User Input box

Basic principle

When you create a new project, the form contains a text box entitled XML User Input. In this box, a default XML template file has been provided, looking something like this:

```
<user>
  <field id="login"></field>
  <field id="First_name"></field>
  <field id="Surname"></field>
  <field id="sex" type="single">
    <field id="M"></field>
    <field id="F"></field>
  </field>
  <field id="Birthdate" type="date"></field>
  <field id="Language"></field>
  <field id="Comments"></field>
</user>
```

The XML-formatted text is the basis for the User Complementary Information (UCI) form that will be used to collect user metadata. You can remove it with the `Empty` button, load a pre-existing XML file (`File` button), paste XML data directly, and/or edit the XML directly within the box. If you click the `Preview` button, a (non editable) preview of the form will appear on the right of the screen. Note: once a project has been saved, the XML template – whether provided through a separate file or directly through the XML User Input box – will no longer be editable.

The syntax is straightforward, and it follows a tree structure. In the example above, `user` is the root element. Whatever appears between the opening tag `<user>` and the closing tag `</user>` relates to `user`. In *RGE me!* XML files, `<user>` is the most basic element in the structure; everything you add should be a child element of `<user>`.

Now, let's have a look at the following line:

```
<field id="Birthdate" type="date"></field>
```

A new field has been created with the opening tag `<field ...>` and the closing tag `</field>`. Within the opening tag, there is a list of attributes with their assigned value between double quotes. For instance, the `id` attribute has the value `Birthdate`.

Attributes

id

The `id` is the only compulsory attribute for a *ROE me!* field to be valid. The expected value is an alphanumeric string (avoid spaces and special symbols). The value of `id` will be displayed in the user form as a label for the field. If you think an `id` is not explicit enough for the user, you can create tooltips with the `lbl` attribute. In this case, the value of `lbl` will replace that of `id` in the final form as a label for the field.

lbl

The value of this attribute – a string of characters – will appear as the label of the field in question and also as a tooltip when the user hovers the cursor over the corresponding field in the user form. Note that only the first 40 characters or so will be displayed the final form (hence the usefulness of tooltips).

type

The `type` attribute is intended to describe the kind of data the field can contain. Although specifying a type is not compulsory, we strongly recommend that you do it because subsequent versions of *ROE me!* will use types to compute summaries of the metadata. The following are possible values:

age: the field can contain integers (years) or years + months. In the latter case, the convention is that years and months should be separated by a semi-colon; e.g., 3;8 means 3 years and 8 months.

choice: choose one or several items from a set. Example:

```
<field id="Accommodation" lbl="Type of accommodation of the informant"
type="choice">
  <field id="House"/>
  <field id="Flat"/>
  <field id="Residential" lbl="residential area"/>
  <field id="Housing" lbl="housing estate"/>
  <field id="Public" lbl="public housing"/>
  <field id="Block" lbl="block of flats"/>
  <field id="Complex" lbl="apartment complex"/>
  <field id="Etc"/>
</field>
```

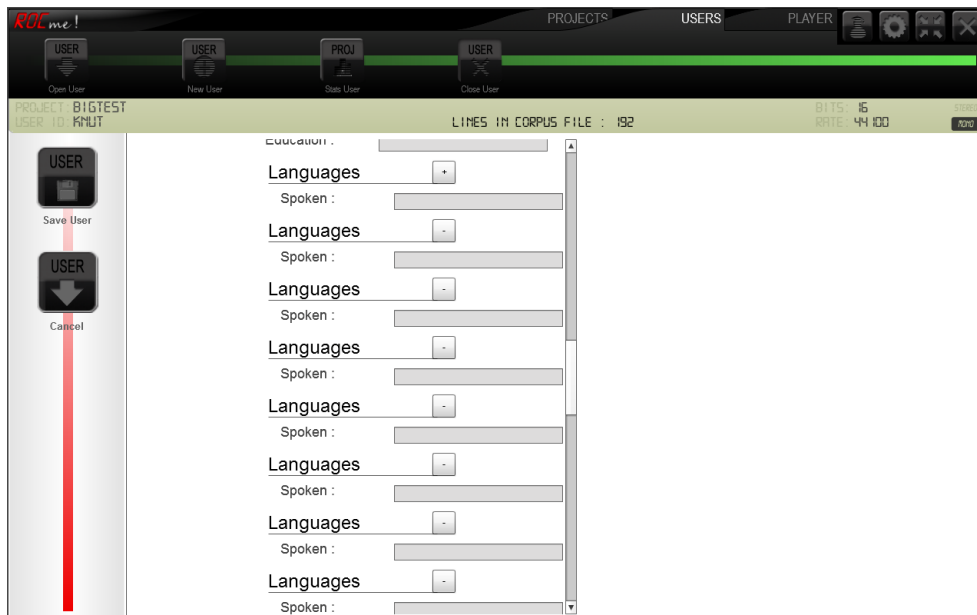
Fields of the choice type appear in the form as check boxes.

date: dates should conform to the following template: DD/MM/YYYY. Example:

```
<field id="Date_of_recording" type="date"/>
```

multiple: indicates that the user can duplicate this field. For instance, in the following example, the speaker may speak more than one language, in which case, the plus sign allows her to create another field (which can be removed by clicking the minus sign); the resulting form is shown in the figure.

```
<field id="Languages" type="multiple">
  <group>
    <field id="Spoken"/>
  </group>
</field>
```



number: self-explanatory.

single: choose one and only one item among a set. Example:

```
<field id="Sex" type="single">
  <field id="M"/>
  <field id="F"/>
</field>
```

The single type appears in the form as a radio button.

required

This optional attribute controls whether the field is required or not. Example:

```
<field id="Sex" type="single" required="required">
  <field id="M"/>
  <field id="F"/>
</field>
```

If a field is required, and the project option User information filled by has the value User, then the participant will not be able to leave the questionnaire and move on to the recording unless she has filled in all required fields.

Loading a pre-existing XML template file

As mentioned earlier, when creating a new project, your XML code for the project template can be entered directly in the XML Users box or prepared in a separate text file. For the latter, use a text editor (not word-processing software) and save the file with the .xml extension. The only important difference between the two methods is that the separate template file should start with an XML declaration

```
<?xml version="1.0" encoding="utf-8"?>
```

specifying the version and the type of encoding. Sample XML files are available from [RDC me! website](http://rdc.me/website).

Corpus files

Corpus files are simple UTF-8 text files containing, on each line, a file identifier, a tab, and the sentence or word to be displayed on the screen. Corpus files should be prepared with a text editor (not a word processing program); standard programs like Notepad (Windows) or TextEdit (Mac) are perfectly adequate. These programs let the user choose the type of character encoding when she saves the file to disk: from the list of options, choose UTF-8.

Corpus files must be loaded at project creation: clicking the File button next to the Corpus field calls a dialog box for retrieving the corpus file on the disk. The corpus file is then copied to the project directory, and this copy then becomes the only version of the corpus that *ROE me!* reads. In other words, if you were to edit the corpus file after project creation, make sure you edit the corpus file within the directory of the current project, not the original version.

In the PLAYER tab, in Administrator mode, you may now (version 2.0) edit the corpus file in your text editor program and reload it in *ROE me!* without – as was previously the case – closing the project. The Reload Corpus function can be found in the right-click menu.

Special tags in corpus files

If you want to include instructions, replace the file identifier by:

```
//
```

You can also display images (instead of text). At project creation, *ROE me!* creates a directory called `_stimuli` within the working directory (i.e. the folder with the name of your project). The procedure is as follows:

1. place your images in the `_stimuli` folder,
2. in the corpus file, link the images to the corpus with the *ROE me!*'s special `` tag (do not forget the file identifier and the tab before):

```
myImage [tab] <img>Image8.png
```

Similarly, *ROE me!* can display videos in .flv format, external html pages, or play mp3 audio files:

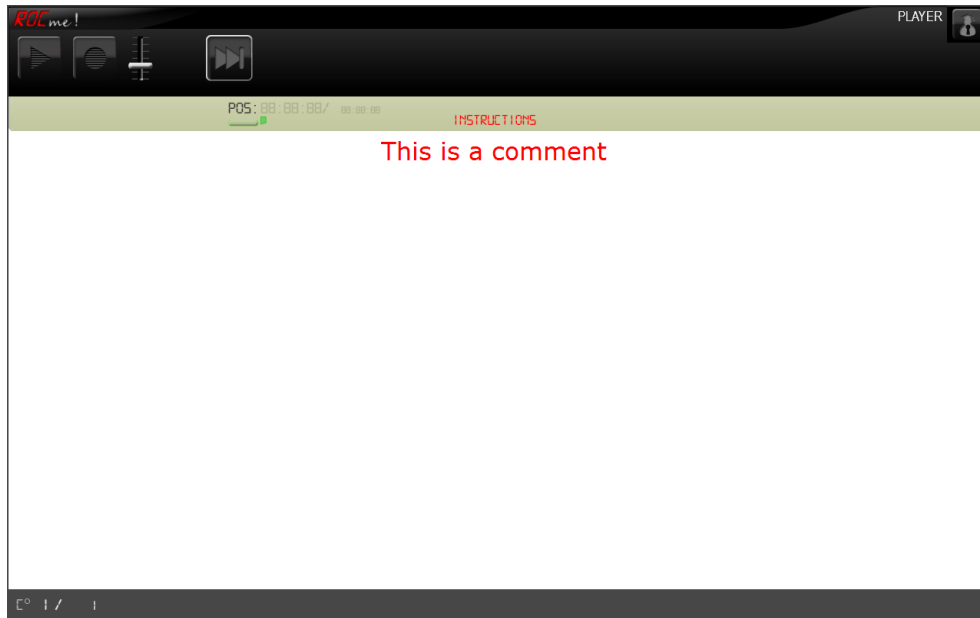
1. place your stimulus (.flv, .htm, .mp3) file in the `_stimuli` folder,
2. link the stimulus file to the corpus file with the appropriate tag. Examples:
 - for a video file: `myVideo [tab] <video>Video1.flv`
 - for an external html page: `myWebPage [tab] <html_ext>htmlStimulus.htm`
 - for an mp3 audio file: `mySound [tab] <audio>Sound1.mp3`

HTML formatting in the corpus file

HTML tags can be used in the corpus file in order to customize the display. HTML tags are available for both the instructions and the corpus itself. The aim of this section is to provide some examples. (Replace `[tab]` by real tab). Sample corpus files are available from *ROE me!* [website](#).

Example 1:

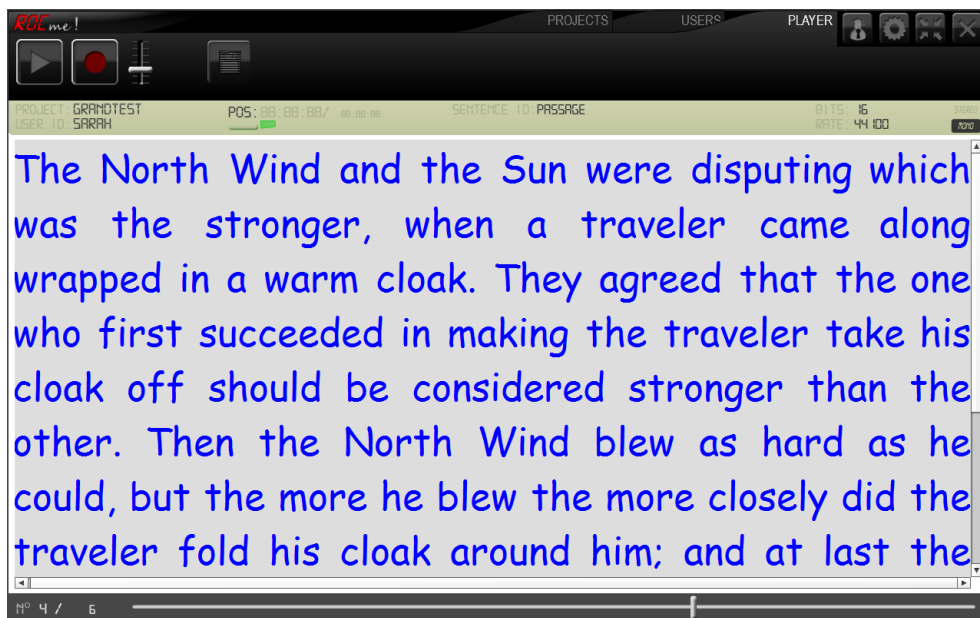
```
// [tab] <html> <font color="#ff0000" size="+3">This is a comment</font>
```



Example 2:

```

passage[tab]<html><p align="justify" style="font-family:Comic Sans MS; font-size:50px; color:#00f; background-color:#ddd;"> The North Wind and the Sun were disputing which was the stronger, when a traveler came along wrapped in a warm cloak. They agreed that the one who first succeeded in making the traveler take his cloak off should be considered stronger than the other. Then the North Wind blew as hard as he could, but the more he blew the more closely did the traveler fold his cloak around him; and at last the North Wind gave up the attempt. Then the Sun shined out warmly, and immediately the traveler took off his cloak. And so the North Wind was obliged to confess that the Sun was the stronger of the two.</p>
  
```



Example 3 :

```

pot[tab]<html><table width="100%" height="100%"><tr><td align="center" valign="middle"><font color="#ff0000" size="+2">4.1. pot</font></td><td align="center" valign="middle"><font color="#00ff00" size="+3">4.2 put</font></td></tr><tr><td align="center" valign="middle"><font color="#0000ff" size="+4">5.1. pat</font></td><td align="center" valign="middle"><span
  
```



```
style="border:5px solid red;font-size:150px;color:#ff00ff:">5.2  
pit</span></td></tr></table>
```



Split

The `Split` function allows participants to record a user-defined subset of sentences from the original corpus. This feature can be useful with very large corpora for applications like automatic speech recognition. At project creation, under the `Split` option, choose `Yes`. Then move to `USERS` tab, create a new user, and click the `Edit User` button. A form is displayed, with a field named `Split`. Here, you must supply the range of stimuli the current user will see, e.g.: 5-15.

Repeat



Normally, one stimulus equals one sound file. The `Repeat` function overrides the constraint and allows speakers to record as many sound files as they want for each stimulus.






Show Clock

This function, in `XP` mode, displays a countdown timer that subtracts the duration of all recorded sound files from a preset value. Suppose for instance that some of your participants receive course credits for one hour of recording with you. Choose the `Show Clock` option in the `PROJECT` tab, create a new user in the `USERS` tab, then click the `Edit User` button. In the form, supply the desired value in minutes. This option is editable after project creation; i.e. within the same project, some speakers may be recorded with, and others without, the `Show Clock` option.

Other features and cool tips

Administrator mode and Administrator Password


The PLAYER tab works in two modes: Administrator and XP. Unless a password has been set, you can freely move from one to the other by clicking the following buttons:  

In XP mode, only recording (stop), playback, saving, and next  are enabled. The user can also adjust the input  and output  gains with the corresponding sliders:  and  respectively.

In Administrator mode, the following (additional) features are available:

- a slider at the bottom of the screen to browse the corpus
- exit **ROE me!**
- go to compact mode
- preferences

Setting an Administrator Password at project creation gives you the possibility to prevent the user from accidentally leaving the XP mode in the PLAYER tab.

The button  in the PLAYER tab indicates that a password has been set. If the user clicks the button, she will be prompted to enter a password.

File names

For optimal management of sound files, file names contain many informative elements separated by underscores. Example:

001_002_toto_G_agathe_deblouse.wav

001 indicates that the file contains a recording of the first sentence that was displayed to the user.

002 means that the sentence can be found on line 2 in the corpus file.

toto is the file identifier.



G (standing for Green) means that no clipping occurred (in other words: the gain meter remained in the Green). If clipping was allowed at project creation, and the signal has been clipped, then G will be replaced by **O** (for Orange).

agathe is the user ID.

deblouse is the project name.

Navigating from the Project mode to the Compact mode, and window size


As of Version 2.0, window size is available in 2 flavors: full-screen or reduced in Project mode. Navigating from one to the other involves the Minimize or Maximize functions available from the right-click menu.

ROE me! also has a Compact mode, which is a simple recorder with basic functions. Moving from Project to Compact mode is accessible through the following buttons:  or 

About

The About window can be accessed by clicking “*RGE me!*” in the main window or using the right click (Windows). It displays the version number and has links to *RGE me!* [Facebook page](#) and [e-mail](#).

Automatic Output File Name

In Compact mode, you may want to record a series of sound files with consistent names, and automatically incremented file IDs. The Automatic Output File Name function in the Preferences lets you do just that. Open Preferences window . Check the Automatic Output File Name box, select an output directory, and create a template. You may use 3 pre-defined variables that let you specify the number of digits for the file ID [dd], the time [time], and the date [date]. The following template:

```
[ddd]_mySound_[date]_[time]
```

means that each file name will start with an automatically incremented ID composed of 3 digits, followed by the string mySound, plus the date and time of recording, all these fields being separated by an underscore. For instance, the first file name might look like this: 001_mySound_2012-4-23_18-11-51.wav

The following one: 002_mySound_2012-4-23_18-11-59.wav

etc.


Check for updates

A simple right click allows the user to check if a new version is available, and to download it. *RGE me!* is expected to evolve to meet the needs of the scientific community involved in language research; we therefore recommend that you periodically check for updates.

Gamepad

Mouse clicks or keyboard strokes may cause unwanted noises. A possible way round the problem is to let participants manage the recording with a gamepad. In order to do this, you need a gamepad and a gamepad mapper (often provided with the driver of your gamepad). The latter is a software program that converts input from the gamepad to key strokes. All you have to do is associate a gamepad button with one of the keyboard shortcuts in *RGE me!*: SPACE BAR (start and stop recording), ENTER (save recording to disk) and TAB (playback).

Preferences window

This window can be called by clicking . In full-screen mode, it allows the Administrator to select the appropriate audio input. In compact mode, sampling rate, quantization size, and number of channels can also be set. Preferences in compact mode also feature a special function that lets the user create template for output file names ([learn more](#)).