

# Instructions for editing AMT Human Intelligence Screening and Experimental Ratings Tasks

In this document, you will learn how to make minor modifications to the screening and experimental rating tasks. You may make these modifications directly in the AMT interface or using an editor such as Atom.

## 1. How to open the file for editing?

Double-clicking the HTML file will open a preview of the task in a web browser. You can also open the file using a text editor, or using a code editor such as Atom (<https://atom.io/>). We'll use Atom, but these instructions will also help you edit the code directly in AMT.

## 2. What is the general structure of the file?

Atom can help you see the general structure of the file by hiding the details. Anywhere you see a > sign before a line of code indicates that that portion of the code can be expanded by clicking on the symbol, as in lines 18 and 22 of the following screenshot. When that symbol is pointing down, toward the bottom of the screen, as in lines 11, 13, and 16, that means that you can collapse that portion of the code. Collapsing and expanding blocks of code will allow you to focus on precisely what you want to modify.

A screenshot of the Atom code editor interface. The editor shows a file with HTML code. Line numbers 11 through 57 are visible on the left. Lines 11, 13, and 16 have a downward-pointing chevron (collapse) symbol. Lines 18 and 22 have an upward-pointing chevron (expand) symbol. The code includes a <crowd-form> tag, a <div name=SlideShow> tag, a <div id=TabQues1> tag, a <div style=background: #428bca; color: #ffffff; padding: 5px> tag, a <div class=panel-body id=consentBody> tag, and an <a href=# class=w3-button> tag. The editor has a dark theme.

At the beginning of the file, you will find the following:

- Line 2 adds a java script file that we need for using AMT.
- Line 3 sets up an element that is needed for the appearance of the page.
- Line 5 allows the use of Spanish symbols.

- Line 11 creates a <crowd-form> element, which is closed at the end of the code. This element is necessary for collecting data. It acts like a container that all the other elements such as textbox, check box, etc. are placed into.

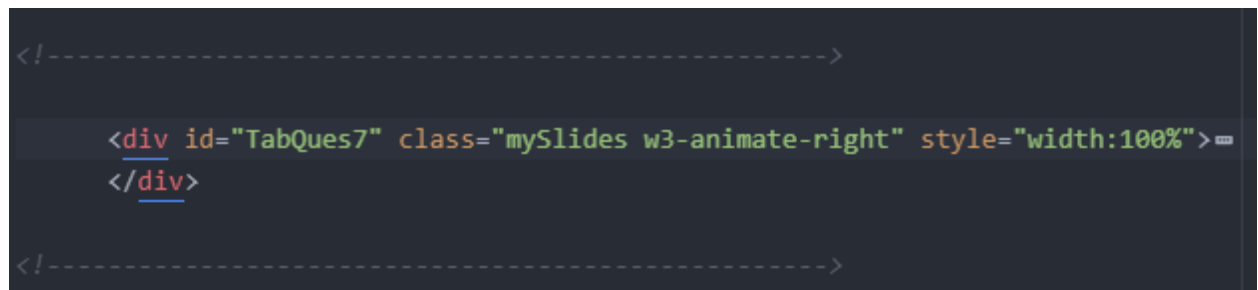
The following sections outline how to make minor modifications to the task, such as adding or removing audio files, changing the scales, etc.

### 3. How to add or remove a questionnaire tab

To remove a questionnaire tab from the screening task, you can delete the <div> ... </div> section. To add a new tab, the easiest way is to copy, paste, and edit an existing tab. Each <div> tab contains a different section of the questionnaire, as follows:

- TabQues1 = Informed consent
- TabQues2 = Demographic questions
- TabQues3 = Questions on Spanish experience
- TabQues4 = Questions on English experience
- TabQues5 = Questions on language use
- TabQues6 = Instructions for the ratings (Note: This section contains a very long string of code that embeds the screen captured images that participants saw; the code to embed that image begins with "<img alt="

You can copy, paste, and edit these sections, but you will need to update the number of the section. So, if we wanted to add another section to the questionnaire, we could copy and paste one of the existing sections (the section whose format most closely mirrors what we want in the new section) and update the id tag (to 7) as in the following screenshot:



```

<!------->

<div id="TabQues7" class="mySlides w3-animate-right" style="width:100%">
  <div>
<!------->

```

If you add a new section between existing sections, you will need to update the ids of all tabs (TabQuesX, where X is a consecutive number). The id of two elements cannot be the same, so please check the numbering carefully.

Once you have copied the tab, you can expand it and begin editing it. I do not recommend making drastic changes to the structure of the questions because that could break the code. You can modify the questions and labels to collect the data you want.

### 4. How to change the response time

You may want to give workers more or less time to respond depending on your research goals. To modify the amount of time they have to log their response after the audio file has finished playing, navigate to the "Configurations" portion of the code and change the value that appears after "var

timeout.” As shown in the following screenshot, the current interface contains a 30-second timeout function.

```
// Configurations -----  
var timeout = 30;  
var audiofiles = [  
    ['file_name', 'file_path'],  
    ['file_name', 'file_path'],  
    ['file_name', 'file_path'],  
    ['file_name', 'file_path'],  
    ['file_name', 'file_path']];
```

## 5. How to manage and change the audios

You can easily change the audio files included in the experiment, but you will need to edit multiple portions of the code. First, after you decide on how many audio files you would like to have in your experiment, you will need to generate a rating page for each audio file. In the following screenshot, you can see that we are generating 24 pages for 24 audio files.

```
<!-- Generate 24 pages for 24 audio files (can be adjusted depending on the number of files)

<div id="TabAudio1" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio2" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio3" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio4" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio5" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio6" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio7" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio8" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio9" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio10" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio11" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio12" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio13" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio14" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio15" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio16" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio17" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio18" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio19" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio20" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio21" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio22" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio23" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio24" class="mySlides w3-animate-right" style="width:100%"></div>
```

Next, you need to generate an audio tab for each file. You can find the number of audio tabs to be generated under the “Generate tabs” portion of the code, as in the screenshot below. Please note that the number of audio pages and tabs must match one another (24 in this example).

```
// Generate tabs -----
document.getElementById('TabAudio1').innerHTML=gentab(1);
document.getElementById('TabAudio2').innerHTML=gentab(2);
document.getElementById('TabAudio3').innerHTML=gentab(3);
document.getElementById('TabAudio4').innerHTML=gentab(4);
document.getElementById('TabAudio5').innerHTML=gentab(5);
document.getElementById('TabAudio6').innerHTML=gentab(6);
document.getElementById('TabAudio7').innerHTML=gentab(7);
document.getElementById('TabAudio8').innerHTML=gentab(8);
document.getElementById('TabAudio9').innerHTML=gentab(9);
document.getElementById('TabAudio10').innerHTML=gentab(10);
document.getElementById('TabAudio11').innerHTML=gentab(11);
document.getElementById('TabAudio12').innerHTML=gentab(12);
document.getElementById('TabAudio13').innerHTML=gentab(13);
document.getElementById('TabAudio14').innerHTML=gentab(14);
document.getElementById('TabAudio15').innerHTML=gentab(15);
document.getElementById('TabAudio16').innerHTML=gentab(16);
document.getElementById('TabAudio17').innerHTML=gentab(17);
document.getElementById('TabAudio18').innerHTML=gentab(18);
document.getElementById('TabAudio19').innerHTML=gentab(19);
document.getElementById('TabAudio20').innerHTML=gentab(20);
document.getElementById('TabAudio21').innerHTML=gentab(21);
document.getElementById('TabAudio22').innerHTML=gentab(22);
document.getElementById('TabAudio23').innerHTML=gentab(23);
document.getElementById('TabAudio24').innerHTML=gentab(24);
```

You can add or remove audio pages and tabs, but it is very important that you do not modify other elements of the code. The code must follow the naming convention TabAudioX where X is a consecutive number that refers to the number of the audio file. For instance, if we wanted to add another audio file for rating, we would need to create a new page and a new tab for that file and update the numbering (to 25). The easiest way to do this is to copy, paste, and update the existing code, as in the last line of the following screenshots:

```
<div id="TabAudio22" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio23" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio24" class="mySlides w3-animate-right" style="width:100%"></div>
<div id="TabAudio25" class="mySlides w3-animate-right" style="width:100%"></div>
```

```
document.getElementById('TabAudio22').innerHTML=gentab(22);
document.getElementById('TabAudio23').innerHTML=gentab(23);
document.getElementById('TabAudio24').innerHTML=gentab(24);
document.getElementById('TabAudio25').innerHTML=gentab(25);
```

To pull audio files into the interface, you have to tell AMT where to find them. You specify this information in the “var audiofiles” portion of the code. In the following screenshot, there are 24 lines, one per file that we would like to pull into the interface. The file name in the AMT code must match the name of the audio clip exactly, and the file path should be a publicly accessible link (i.e., the address of the audio file). The naming convention for this section is:

```
['id', 'path/address of the audio file'],
```

where 'id' can be any identifier/name that you would like.

[illegible]

Again, if we were to add another file for rating, we would need to add a line to var audiofiles so that AMT would know where to locate the file to pull it into the interface. Below, you can see a screenshot of what this section looks like when it is filled in with file names and paths.

```
var audiofiles = [
['_H600a_1', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H600a_1.WAV'],
['_H600a_2', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H600a_2.WAV'],
['_H603a_1', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H603a_1.WAV'],
['_H603a_2', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H603a_2.WAV'],
['_H605a_1', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H605a_1.WAV'],
['_H605a_2', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H605a_2.WAV'],
['_H606a_1', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H606a_1.WAV'],
['_H606a_2', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H606a_2.WAV'],
['_H607a_1', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H607a_1.WAV'],
['_H607a_2', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H607a_2.WAV'],
['_H608a_1', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H608a_1.WAV'],
['_H608a_2', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H608a_2.WAV'],
['_H610a_1', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H610a_1.WAV'],
['_H610a_2', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H610a_2.WAV'],
['_H611a_1', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H611a_1.WAV'],
['_H611a_2', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H611a_2.WAV'],
['_H612a_1', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H612a_1.WAV'],
['_H612a_2', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H612a_2.WAV'],
['_H613a_1', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H613a_1.WAV'],
['_H613a_2', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H613a_2.WAV'],
['_H614a_1', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H614a_1.WAV'],
['_H614a_2', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H614a_2.WAV'],
['_H615a_1', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H615a_1.WAV'],
['_H615a_2', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H615a_2.WAV'],
['_H616a_1', 'https://cnagleamt.s3.amazonaws.com/5+AMT+2019+Intelligibility/H616a_1.WAV'],
```

## 6. How to alter the rating scales

Altering the rating scales involves editing multiple parts of the code. If you want to modify the rating scales for the audio files, you need to locate the appropriate tab generator. The beginning of the code looks like this and is signaled by "`<!--Audio`"

```
function gentab(tabcntr){  
    gentb = '<!--Audio-----><div  
id=daudio' + tabcntr + ' width=100%><br><p id=chkbx-msg-01 class="text-center  
style=font-size:10pt;color:#777> <tr> <td><i>extremely<br>disfluent</i></td> <  
disabled=true></crowd-slider> <table width=90% style=font-size:10pt;color:#777  
min="1" max="7" step="1" pin="true" value="1" style=width:90% disabled=true><  
onclick="plusDivs(1); window.scrollTo(0,0);"><b>&nbsp;&nbsp;&nbsp;&nbsp;&Continue &nbsp;&nbsp;&&  
return gentb;
```

Note that this is a very long string of code that generates the entire rating interface, including the scales. To modify the number of points on the scale, find the portion of the code that refers to “min”, “max”, and “step,” as on the second line in the following screenshot:

```
tatic/1po7j9ojuq9d9t9ik9zb74hzvf8j8vw5.png width=55px> (Play)</a> <div id=t
' name=Fluency' + tabcntr + ' min="1" max="7" step="1" pin="true" value="1"
wd-slider id=Comprehensibility' + tabcntr + ' name=Comprehensibility' + tabc
order-radius: 10px;"> <br><label><b>Foreign Accent</b></label><br> <crowd-sl
style=background:#eee; onclick="plusDivs(-1); window.scrollTo(0,0);">< Back
```

As shown, the fluency scale is a 7-point scale where each step is one point and the slider is pinned, or begins, at 1. We could change this to a 9-point scale by updating max to 9 (max="9"), and we could pin the cursor to the midpoint, 5, by updating the value to 5 (value="5"). Please note that you would need to do this for each of the three scales, and if you are using a screening task or including sample files for rating, that portion of the interface would need to be updated as well.

## 7. Additional information

There are many other portions of the interface that could be changed. However, unless you are very familiar with code, I do not recommend changing them. If they are modified incorrectly, the task may not work properly or at all. This code uses some Amazon crowd objects such as crowd-form, crowd-slider, etc. You can find documentation on crowd objects at the following link:

[https://docs.aws.amazon.com/AWSMechTurk/latest/AWSMturkAPI/ApiReference\\_HTMLCustomElementsArticle.html](https://docs.aws.amazon.com/AWSMechTurk/latest/AWSMturkAPI/ApiReference_HTMLCustomElementsArticle.html)