Pulsar Audio Labeling Instructions

1. Project Overview

In this project, Alignerrs clean up word-level audio transcriptions in the internal audio labeler. Each task starts when Alignerrs upload or open the assigned audio files in the internal editor. Each uploaded file may contain one or more clips that need to be reviewed and annotated. The editor displays a rough pre-labeled transcript as word boxes. Alignerrs listen to each clip and correct those word boxes so the transcript matches what is actually said.

This may include:

- correcting a wrong word
- deleting a word that was not actually said
- adding a missing word
- adding an allowed sound, style, or speech-feature tag
- marking guessed words with double parentheses when needed
- marking unintelligible speech when needed
- choosing a standalone discard reason when the whole clip qualifies
- adjusting the time boundaries of a word or tag in the editor
- splitting one segment into multiple segments when it contains more than one word
- merging segments when a single word was split into two or more
- removing a segment that contains only silence or background noise with no voice activity
- fixing a segment that has no duration (start time equals end time) by extending it to cover the word

The final output comes from the internal editor and QA flow:

1. Open your assigned task from Labelbox. The audio loads automatically in the internal editor (no manual upload needed).

2. Review and align each clip in the editor. Your work autosaves automatically as you go, so you do not need to export or manually save anything. If you close and reopen the task, your progress is restored.

3. Click Run QA (top of the editor) to check every clip, then fix anything it flags.

4. Click Submit (top of the editor). This is the step that finalizes your work: it uploads your JSON, runs the final QA, and generates a review link.
5. Copy the review link that Submit gives you and paste it into the Labelbox text field.

6. Complete the platform ontology field.
7. Submit the completed data row in Labelbox.

What saves your work: Submit is the action that writes your final JSON. The per-clip Done toggle only marks a clip as finished, and Export JSON is just an optional local download. Neither of those saves your work to us, so do not rely on them as a save.

There will be two training videos:

1. A labeling-process video that explains how to use the internal editor.
2. A platform-norms video that explains QA, upload expectations, Labelbox ontology fields, and final submission.

This document explains the transcription rules, allowed tags, discard reasons, boundary expectations, overlap handling, QA flow, and final submission process.

Finally a disclaimer, please use Chrome for this project. It plays best with our editor.

Time limit: each data row has a 4 hour limit. Open, label, run final QA, paste the QA link, and submit within 4 hours of opening the row.

2. Training Video and Evaluation

Before beginning production work, Alignerrs should review the below required training videos.

2.1 Labeling-Process Video

https://share.descript.com/view/QVp4c2QH2rr

2.2 Platform-Norms Video

https://www.loom.com/share/a296003ce91c42b69e380a27c06d6730

Alignerrs will also complete a training course before production access. The course has four modules, each with a lesson, a quiz, and hands-on boundary exercises in the editor, followed by a final hands-on assessment of several audio clips labeled using the same rules described in this document. The final is graded against gold labels on word accuracy and boundary precision, and Alignerrs must pass it to unlock production access.

The course confirms that the Alignerr can:

- correct pre-labeled transcript words
- adjust word boundaries to the sound in the editor
- add only allowed mid-transcription tags
- use end-of-file tags correctly
- handle guessed, unintelligible, and background speech correctly
- identify overlapping words when applicable

3. Core Labeling Concept

The internal editor displays the audio waveform and a draft word-level transcript.
Each word or allowed tag appears as its own editable segment in the editor. Alignerrs should listen to the audio and make the segments match what is actually heard.
For each segment, Alignerrs should check:

- Is the word or tag correct?
- Is the segment present only when the sound is actually heard?
- Is the segment missing anything that should be labeled?
- Are the segment boundaries aligned to the actual sound?
- Does the segment follow the tokenization and tag rules in this document?

Keep a segment when it contains a non-lexical vocal sound such as a cough or laugh, because that counts as voice activity. Only remove a segment when it has no word and no non-lexical sound.
The editor handles the underlying export format. Alignerrs should focus on producing accurate word/tag segments and boundaries in the editor.

4. Boundary Editing Standard

Alignerrs adjust timing by moving word or tag boundaries in the internal editor.
For words and allowed tags, boundaries should closely match the actual speech or sound signal.
General rule:

- The start of the segment should align to the beginning of the word or tag sound.
- The end of the segment should align to the end of the word or tag sound.
- When possible, boundaries should be within roughly 20 ms of the true beginning or end of the speech signal.
- For single-speaker speech, adjacent word segments should not overlap.
- When speech is fast, adjacent segments may need to be very close together.
- If placing both boundaries within 20 ms would create overlapping segments in single-speaker speech, prioritize preventing overlap while keeping each boundary as

close as possible to the real speech signal.
- Each segment needs correct, non-zero boundaries. Never give a segment identical start and end times, and ensure multi-word segments do not have incorrect timestamps applied.

Because the editor and QA tooling handle the exact export format, the training video should be treated as the source of truth for the mechanics of moving boundaries.

5. Annotation Tags

The editor supports tags for non-lexical sounds, style, uncertainty, unintelligible speech, and task-level discard cases.
Use tags when the corresponding audio feature clearly occurs. Do not invent tags outside the allowed set.

5.1 Non-Lexical Sound Tags

Use these tags when the sound clearly occurs in the clip.

Tag	Meaning
[l]	Laughter
[c]	Cough
[s]	Sigh
[t]	Throat clear
[z]	Sneeze
[p]	Lip smack
[hn]	Human noise catch-all
<hum>	Humming
[fp]	Filled pause

Tag	Meaning
[artifact]	Audio artifact

Use [hn] only when a clear, foreground, non-lexical human noise is present and no more specific tag applies.

A clearly audible breath is a non-lexical human noise: give it its own segment and tag it [hn] (use [s] if it is a clear sigh). Breaths are most often missed at the very start and end of a clip, so check both ends. Do not tag faint or background breathing that is not a clear, foreground sound.

Use [artifact] for technical audio artifacts such as a pop, click, static burst, or other non-human audio defect.

The words "um" and "uh" are lexical words, not [fp].

5.2 Style and Speech-Feature Tags

Use these tags around the affected text or as the editor workflow specifies.

Tag	Meaning
<ws> </ws>	Whisper
<lv> </lv>	Low voice
<sg> </sg>	Singing
<ct>	Cross talk when speech is unintelligible because speakers overlap
<i>	List item separator
[bg]	Background speech

If speech is intelligible even when speakers overlap, keep the intelligible words in the transcript and use the platform overlap fields as described in Section 8.

Examples:

Whisper:

```
<ws>please be quiet</ws>
```

The style tag is attached directly to the affected word token. It is not a standalone token. If only one word is whispered, the opening and closing tags both go on that word:

```
<ws>him</ws>
```

If multiple words are whispered, the opening tag goes on the first affected word and the closing tag goes on the last affected word:

```
<ws>please be quiet</ws>
```

Low voice:

```
<lv>do not want them to hear</lv>
```

If only one word is low voice, the opening and closing tags both go on that word:

```
<lv>him</lv>
```

Singing:

```
<sg>happy birthday to you</sg>
```

If only one word is sung, the opening and closing tags both go on that word:

```
<sg>happy</sg>
```

The workflow / hotkeys will help apply these tags, but the final transcript should still show the tag attached to the word token.
Cross talk where overlapping speech is not intelligible:

```
<ct>
```

Use <ct> only when overlapping speech is present and the words cannot be transcribed clearly. <ct> is different from <ws>, <lv>, and <sg> because it is not wrapping a known word. If the overlapping words are intelligible, transcribe the words and complete the overlap fields in the platform instead.
List item separator:

```
set alarm for monday<i> wednesday and friday mornings
```

Use <i> when there is a list break between items that are not separated by a spoken conjunction like "and" or "or."
The <i> tag is appended directly to the word immediately before the list break. It is not a standalone token.

Background speech:

```
I need to go [bg] now
```

Use [bg] when background speech is present according to the editor/project workflow. Do not use [hn] for background speech.

5.3 Guessed and Unintelligible Speech

There are two cases, with two different markings:

- Guessed word — you can make out the word but aren't fully sure: transcribe the word and wrap it in double parentheses, ((word)).
- Unintelligible — you cannot make out any word: use empty double parentheses, (()), for that portion.

"Do not guess" means: never put a word inside the parentheses that you cannot actually hear. If you can identify the word, use ((word)); if you genuinely cannot, use (()) — for example, do not invent a word for an unintelligible start, use (()). Segment every clearly audible word individually, even if the rest of the clip is unintelligible. (See Example 6.)

5.4 End-of-File Tags

End-of-file tags apply to the whole audio file or clip. They are placed at the end of the relevant audio file, not immediately after the affected speech segment.
A file may contain many words before and after the audio feature. The end-of-file tag still goes at the end of that file's transcript.

Tag	Meaning	When to use
<ms>	Media speech	Use if speech in the file came from a TV, radio, app, speakerphone, recording device, or similar media source.
<pws>	Partial whisper	Use if any part of the file contains whispered speech.

Example for media speech:

```
the radio said turn left soon <ms>
```

Even if only "turn left soon" came from media speech, <ms> belongs at the end of the file's

transcript.
Example for partial whisper:

```
said <ws>please stop</ws> and then walked away <pws>
```

The whispered words are tagged directly in the transcript. If only one word were whispered, it would look like <ws>word</ws>. The <pws> tag is added at the end of the file because the file contains some whispered speech.
If both end-of-file tags apply, include both at the end according to the editor workflow:

```
final words <ms> <pws>
```

Do not place <ms> or <pws> directly after the first affected phrase unless that phrase is also at the end of the file.

5.5 Standalone Discard Tags

Use one of these only when the entire clip qualifies. Do not combine a standalone discard tag with other transcription content.

Tag	Meaning
(())	Entire clip is unintelligible
<ga>	Garbled audio
<na>	No audio
<ns>	No speech
<hum>	Entire clip is only humming
[se]	Entire clip should be discarded or flagged as sensitive according to project policy

6. Tokenization Rules

Transcribe in spoken form, meaning how the word is actually pronounced rather than how it is written. For example, write "kesha", not "Ke$ha".
Tokenization determines what counts as one word segment. Transcribe all words in lowercase. Do not use punctuation in the transcript except:

- apostrophes when required by the word, such as don't
- hyphens in approved standardized tokens, such as uh-huh, mm-mm, and nuh-huh

Do not use commas, periods, question marks, exclamation points, colons, semicolons, quotation marks, or capitalization. Some more specific cases below:

6.1 Hyphenated Words

Hyphenated words are treated as separate lexical items. Since hyphens are typically stripped, each component receives its own segment.
Example:

`"eighty-seven" → "eighty" + "seven"`

Hyphens should remain only when they are part of an approved standardized token.
Examples:

`uh-huh`
`mm-mm`
`nuh-huh`

6.2 Backchannels and Filled Pauses

Some short vocal tokens are real lexical items in this project. If the sound matches one of the standard spellings below, transcribe it as the word itself rather than replacing it with [fp].
Use the dictionary spelling when the token appears in the dictionary of record. For non-dictionary backchannels, use the standard forms below.

Audio / Intent	**Transcription**
Agreeing or listening signal, mouth closed	mhm
Agreeing or listening signal, mouth open	uh-huh
Thinking or considering	hmm
Short hesitation, open vowel	uh
Hesitation with closed lip / nasal	um
Disagreeing signal, mouth closed	mm-mm

Audio / Intent	Transcription
Disagreeing signal, mouth open	nuh-huh
Surprise or realization, rhymes with "owe"	oh
Excitement or impressed, rhymes with "food"	ohh
Other non-standard hesitation or backchannel	[fp]

Example:

```
I um think uh yes
```

Segments:

```
1
um
think
uh
yes
```

Use [fp] only when the vocalization is a filled pause or backchannel that does not match one of the standard word forms above and is not better captured by another allowed tag.

6.3 Letter Sequences and Acronyms

Each individually pronounced letter in a letter sequence is a separate word and gets its own segment.
Example:

```
abc
```

Segments:

```
a
b
c
```

Periods after individually pronounced letters are stripped. For example, a. b. c. should be treated as abc.
Acronyms pronounced as words are single lexical items and receive one segment.

Example:

```
opec
```

The word "okay" should be transcribed as ok and kept within one segment. It is not treated as a letter sequence.

6.4 Numbers

Numbers transcribed as words follow standard tokenization. Since hyphens are stripped, each spoken component is a separate segment.
Examples:

```
"eighty-seven" → "eighty" + "seven"
"three zero six" → "three" + "zero" + "six"
```

Transcribe numbers as the speaker pronounces them:

- "three zero six" or "three oh six", depending on whether they said "zero" or "oh"
- "306" said as a whole number becomes "three hundred and six"
- ordinals are spelled as said, for example "5th" becomes "fifth"

6.5 Mispronunciations

If a speaker clearly intends a recognizable word but mispronounces it, transcribe the intended word rather than spelling the mispronunciation phonetically.
Example:

```
"vitis" intended as "visit" → visit
```

Do not create a new token for the mispronounced form if the intended word is clear from the audio and context.

6.6 Contractions and Informal Contractions

Keep standard contractions as one word with the apostrophe: that's, it's, don't, what's.
Transcribe these informal contractions as written: gonna, wanna, gotta, gimme, lemme, whatcha.
If an informal contraction is in the dictionary of record, use the dictionary spelling, for example c'mon and 'cause.
If a contraction is not in that list or the dictionary, spell out the words it is made of:

- "there're" becomes "there are"
- "i'd've" becomes "i'd have"
- "mustn't've" becomes "mustn't have"

- "whaddya" becomes "what do you"

If a contraction sounds just like the separate words said individually, transcribe the separate words.

6.7 Truncations and Self-Corrections

When a speaker cuts off a word, transcribe as much of the word as you can hear and use a hyphen to mark where it cuts off. Hyphens are optional, so both forms are acceptable:

- "set alarm for elev-" or "set alarm for elev"
- "-ey stop" or "ey stop"

Do not add a hyphen before or after a complete word. If a full word is only slightly clipped but still fully audible, transcribe the whole word, for example "turn down the television".

For a self-correction that cuts a word off, use a hyphen at the cutoff. If the self-correction does not cut a word off, transcribe it normally with no hyphen, for example "call janet wait i mean janine" and "get directions to the to the hotel".

If the speaker is cut off while spelling letters, transcribe only the letters they said, for example "who founded h. and m.".

If a non-lexical sound interrupts a word, place the tag inside the word at the break, for example "ama- [hn] zing".

If the entire clip is only one extremely truncated word, discard the task.

Stutters and repetitions: transcribe every audible repetition, not just one. If the speaker restarts a cut-off word, each cut-off attempt takes a hyphen and the final complete word does not, for example "el- el- elevator" becomes "el-" + "el-" + "elevator". If the speaker repeats a whole word, transcribe each occurrence with no hyphen, for example "the the the" becomes three "the" segments. Do not collapse a stutter or repeated word into a single segment.

6.8 Homophones

When a word could be spelled more than one way, use context and best judgment to pick the intended word. For a word heard in isolation, choose the more likely one, for example transcribe an isolated affirmation as "right" rather than "write". When both spellings are about equally likely in isolation, for example "waist" and "waste", either spelling is acceptable.

7. Unclear, Guessed, Background, and Non-Target

Speech

If speech is unclear but a word is still reasonably guessable, mark the guessed word with double parentheses.
Example:

```
((word))
```

If speech is unintelligible and no word can be guessed, use (()) for that unintelligible portion.
If the entire clip is unintelligible, use (()) alone.
Use [bg] for background speech when the project/editor requires it. Do not use [hn] for background speech.
Use [hn] only for clear, foreground, non-lexical human noises that are not covered by a more specific tag.

Non-Target Languages

If a non-target language appears in the speech and you can transcribe it, transcribe as much as you can using plain letters with no accents or diacritics. If a letter carries an accent that is not part of the target language, use the closest plain letter.
Do not transcribe non-target language that is only in the background.
If any non-target language segment cannot be transcribed, even a small portion of the clip, discard the entire task.

8. Overlap and Multiple Speakers

For single-speaker audio, adjacent word segments should not overlap.
If the audio contains multiple speakers and their words overlap, both intelligible words should remain in the transcript.
The platform asks three things: whether there is more than one speaker, whether there is overlapping speech (yes or no), and which words overlap. Answer the multiple speaker question and the overlapping speech question, and if there is overlap, list the overlapping words. For a single speaker, segments must not overlap. Across different speakers, segment times are allowed to overlap.
The ontology will ask whether there were multiple speakers. If there were multiple speakers, Alignerrs should list any overlapping words if they exist.
Example ontology response:

```
Multiple speakers: Yes
```

```
Overlapping words: hello, howdy, yes, yeah
```

If there are no overlapping words, answer according to the platform instructions and leave the overlapping-words field blank or mark it as not applicable, depending on the field requirements.

9. End-of-File Tags

End-of-file tags are not mid-transcription tags. They are placed at the end of the relevant audio file in the editor according to the labeling-process video.

9.1 Media Speech <ms>

Use <ms> as a global flag if speech in that audio file originated from a TV, radio, app, speakerphone, recording device, or similar media source.
<ms> belongs at the end of the relevant audio file, not after the affected speech segment.

9.2 Partial Whisper <pws>

Use <pws> as a global flag if any portion of the relevant audio file contains whispered words.
<pws> belongs at the end of the relevant audio file, not after the whispered segment.

9.3 Multiple End-of-File Tags

If more than one end-of-file tag applies to the same audio file, place each applicable tag according to the labeling-process video.

10. Standalone Discard Tags

When an audio clip hits a critical failure condition, use one of the following discard tags according to the labeling-process video.
Do not add punctuation, capitalization, or any other content when the full clip qualifies for a standalone discard tag.

Tag	Meaning	Usage
(())	Entire clip is unintelligible	Use when the full clip cannot be understood.
<ga>	Garbled Audio	Use when technical recording issues such as

Tag	Meaning	Usage
		digital drops, skips, or microphone damage make speech evaluation impossible.
<na>	No Audio	Use when the file contains a completely blank audio signal.
<ns>	No Speech	Use when there is white noise or background environment sound, but zero human speech or taggable human vocal features. If the clip has no real speech (only mechanical sounds, background noise, audio glitches, or fully unintelligible), discard it. Do not create segments or transcribe noise.
<hum>	Humming only	Use when the entire clip is only humming and contains no other transcribable speech.
[se]	Sensitive audio	Use when the whole clip qualifies for the sensitive-audio workflow according to project policy.

11. QA and Submission Workflow

There are two QA steps.

11.1 Live Session QA

Live session QA is used while the Alignerr is working and it can be run inside the editor at any time.

Live session QA may flag:

- typos
- invalid tags
- boundary issues
- overlapping words
- other editor/export issues

11.2 Final QA

Final QA is used when the task is complete.
Alignerrs upload the required final files to the Vercel QA tool according to the platform-norms video. Final QA generates the QA link that must be pasted into Labelbox.
Final QA may also surface overlapping words. Alignerrs should use that output to complete the ontology fields.

11.3 Platform Submission

After final QA:

1. Copy the final QA link generated by Vercel.
2. Paste the QA link into the Labelbox ontology field.
3. Answer the multiple-speaker ontology question.
4. If multiple speakers are present, list any overlapping words surfaced by QA.
5. Submit the completed data row.

12. Common Mistakes to Avoid

Do not:

- Treat the draft transcript as automatically correct.
- Use tags that are not supported by the editor or project rules.
- Mark "um," "uh," or standard backchannels such as mhm, uh-huh, hmm, mm-mm, nuh-huh, oh, or ohh as `[fp]` instead of as lexical words. Listen closely and match the exact sound; do not substitute similar sounds (for example, flagging "umm" as "hmm", writing "mhm" as "mm", or substituting "uh" for a throat-clear).
- Miss segments, especially at the start and end of clips (for example, missing a breath at the start, missing a truncated "-s", or missing an "mhm" at the very end). Every audible word and non-lexical sound needs its own segment. Tag an audible breath as [hn].
- Insert or guess words that are not actually audible (for example, writing "i don't know" when the audio only says "yeah", or transcribing "aren't" when there is no matching audio).

- Merge separate words into a single segment (for example, combining "I am" into one segment, or writing "you know" as "y'know").
- Use annotation tags incorrectly or outside the context described in Section 5.
- Collapse hyphenated words into one segment.
- Mark "um," "uh," or standard backchannels such as mhm, uh-huh, hmm, mm-mm, nuh-huh, oh, or ohh as [fp] instead of as lexical words.
- Split ok into letters or transcribe "okay" instead of ok.
- Merge individually pronounced letters into one segment.
- Split acronyms that are pronounced as words.
- Transcribe clear mispronunciations phonetically instead of using the intended word.
- Confuse the two unclear-speech marks: a word you can make out but aren't sure of is ((word)); speech you can't make out at all is (()).
- Use [hn] for guessed or unclear speech.
- Use [hn] for background speech.
- Use [hn] when a more specific tag applies.
- Forget that full-clip unintelligible audio should be (()) alone.
- Combine a standalone discard tag with other transcription content.
- Place <ms> or <pws> immediately after a speech segment instead of at the end of the relevant audio file.
- Use a standalone discard tag alongside other transcription content.
- Skip live session QA when checking the JSON would catch issues early.
- Forget to paste the generated final QA link into Labelbox
- Collapse a stutter or repeated word into one segment. Transcribe every audible repetition (for example "the the the" is three segments), and keep a hyphen on each cut-off attempt in a restart like "el- el- elevator".

13. Prominent Quick Examples

Example 1: Normal Words

Transcript:

```
Hello there
```

Segments:

```
Hello
there
```

Example 2: Mid-Transcription Cough

Transcript:

```
I [c] agree
```

Segments:

```
1
[c]
agree
```

Example 3: Laughter Between Words

Transcript:

```
That was [l] funny
```

Segments:

```
That
was
[l]
funny
```

Example 4: Filled Pause

Transcript concept:

```
I [fp] think so
```

Segments:

```
[fp]
think
SO
```

Example 5: Tokenization Examples

Hyphenated number:

```
eighty
seven
```

Letter sequence:

```
a
b
C
```

Acronym pronounced as a word:

```
opec
```

"okay" transcribed as ok:

```
ok
```

Backchannels and filled pauses transcribed as lexical words when they match the standard forms:

```
mhm
uh-huh
hmm
uh
um
mm-mm
nuh-huh
oh
ohh
```

Clear mispronunciation transcribed as intended word:

```
visit
```

Example 6: Guessed and Unintelligible Speech

Guessed word:

```
((word))
```

Partially unintelligible clip:

```
I want to see (())
```

Fully unintelligible clip:

```
(())
```

Example 7: End-of-File Flags

<ms> and <pws> are placed at the end of the relevant audio file according to the labeling-process video.

Media speech:

```
the radio said turn left soon <ms>
```

Partial whisper:

```
said
<ws>please stop</ws>
and then walked away
<pws>
```

Single-word whisper:

```
<ws>him</ws>
```

Both apply:

```
final words <ms> <pws>
```

Example 8: Multi-Speaker Overlap

If one speaker says "hello" and another speaker says "howdy," and the words partially overlap in time, both words should remain in the transcription.

Both live session QA and final QA may surface overlapping words. Use the QA output to fill the Labelbox ontology fields.

Example ontology response:

```
Multiple speakers: Yes
Overlapping words: hello, howdy, yes, yeah
```

Do not add an inline overlap tag.

Example 9: Completely Blank Audio

If the audio file contains a completely blank audio signal, use the standalone discard tag according to the labeling-process video:

```
<na>
```

No other content should be entered.