

Latex Template for WAC 2016

Nicholas Jillings
n.g.r.jillings@se14.qmul.ac.uk

Brecht De Man
b.deman@qmul.ac.uk

David Moffat
d.j.moffat@qmul.ac.uk

Joshua D. Reiss
joshua.reiss@qmul.ac.uk

Centre for Digital Music
School of Electronic Engineering and Computer Science
Queen Mary University of London
Mile End Road, London E1 4NS
United Kingdom

ABSTRACT

Here comes the abstract.

1. INTRODUCTION

Introducing the paper. Referring to [1]. Talking about what we do in the various sections of this paper. Pointing out that the header of the paper kind of looks like the Bat-sign.

2. ARCHITECTURE

A slightly technical overview of the system. Talk about XML, JavaScript, Web Audio API, HTML5.

3. INTERFACES

We could add more interfaces, such as:

- Hedonic Scale
- Multi attribute ratings
- MUSHRA
- Interval Scale
- Rank Scale
- 2D Plane rating - e.g. Valence vs. Arousal
- Likert scale

There are also the following interfaces, which would require a slightly different ‘engine’ underneath, e.g. loading a different page for every possible pair.

- AB Test
- ABX Test
- JND

A screenshot would be nice.

4. ANALYSIS AND DIAGNOSTICS

It would be great to have easy-to-use analysis tools to visualise the collected data and even do science with it. Even better would be to have all this in the browser. Complete perfection would be achieved if and when only limited setup, installation time, and expertise are required for the average non-CS researcher to use this.

The following could be nice:

- Web page showing all audioholder IDs, file names, subject IDs, audio element IDs, ... in the collected XMLs so far (`saves/*.xml`)
- Check/uncheck each of the above for analysis (e.g. zoom in on a certain song, or exclude a subset of subjects)
- Click a mix to hear it (follow path in XML setup file, which is also embedded in the XML result file)
- Box plot, confidence plot, scatter plot of values (for a given audioholder)
- Timeline for a specific subject (see Python scripts), perhaps re-playing the experiment in X times realtime. (If actual realtime, you could replay the audio...)
- Distribution plots of any radio button and number questions (drop-down menu with ‘pretest’, ‘posttest’, ...; then drop-down menu with question ‘IDs’ like ‘gender’, ‘age’, ...; make pie chart/histogram of these values over selected range of XMLs)
- All ‘comments’ on a specific audioelement
- A ‘download’ button for a nice CSV of various things (values, survey responses, comments) people might want to use for analysis, e.g. when XML scares them

A subset of the above would already be nice for this paper. Some pictures here please.

5. CONCLUDING REMARKS

Perhaps an ‘engineering brief’ such as this one doesn’t really have a lot of conclusion, except ‘We made this’.

You can check it out at code.soundsoftware.ac.uk/projects/webaudioevaluationtool.



Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). Attribution: owner/author(s).

Web Audio Conference WAC-2016, April 4–6, 2016, Atlanta, USA

© 2016 Copyright held by the owner/author(s).

6. FUTURE WORK

Perhaps here, perhaps not. Talking a little bit about what else might happen. Unless we really want to wrap this up.

7. REFERENCES

- [1] N. Jillings, D. Moffat, B. De Man, and J. D. Reiss. Web Audio Evaluation Tool: A browser-based listening test environment. In *12th Sound and Music Computing Conference*, July 2015.