

TOWARD INTERACTIVE MUSIC NOTATION FOR LEARNING MANAGEMENT SYSTEMS

Matthias Nowakowski,* Aristotelis Hadjakos, & André Stärk

University of Music Detmold, Detmold, Germany

*Address all correspondence to: Matthias Nowakowski, University of Music Detmold, Hornsche Strasse 44, 32756 Detmold, Germany; Tel.: +49 (0)5231 975-874, E-mail: Matthias.nowakowski@hfm-detmold.de

Learning management systems (LMSs) are currently lacking when it comes to music education. Music-specific features, especially those based on music notation and audio, do not meet the necessary requirements. This should be remedied by integrating the now world-leading open-source platform for learning tools H5P into several LMSs, which will then create significantly more application possibilities. An integrated music notation with audio output is indispensable in these systems to be able to create typical practice and presentation scenarios for music education. Here the open-source notation environment Verovio is to be extended by graphical input and annotation capabilities and to prepare it for H5P integration. Finally, H5P applications are to be programmed based on this notation environment. Typical forms of presentation and practice patterns in music education will be covered, including creation of automatically evaluable notation tasks and analysis tasks, as well as synchronization possibilities.

KEY WORDS: music theory, LMS, music notation, interactive exercises

1. MOTIVATION

Over the past several years, there has been a great interest in online music education. Many systems and approaches have dealt with instrumental and singing pedagogy. This continues to be a main focus of research and technical innovation (Bowman, 2014; Blackburn & Hewitt, 2020; Johnson, 2020; Merrick, 2020; Nikolsky, 2020; Osborne, 2020; Pike, 2020). In general, most research covers instrumental and singing practice. This is, of course, a central topic of music education and, in the context of online learning, one-to-one music tuition bears its own challenges: the quality of the equipment, sound and video quality, network delays, etc.

In contrast to that, in our project we focus on another central part of higher music education, which is music theory. The subject covers a broad range of topics, including techniques of tonal composition, harmony, counterpoint, music notation, music analysis, and ear training. To learn the concepts, music theory students often solve written musical tasks. For example, they name musical phenomena and structures, analyze chord progressions, identify musical

themes, or harmonize a melody for four parts in the style of Bach. Such learning scenarios are not well supported by current learning management systems (LMSs). Usually, the teacher provides feedback in a face-to-face learning environment.

Web-based LMSs or learning platforms have been used with great success in a wide area of academic subjects for which exists a wide variety of plug-ins. There are some for music theory, such as a plug-in that features various musical question types [see Brisson (2019)]. Among other things, the user can write a note or enter a musical interval. However, the support of musical notation in a LMS is currently very limited. Such existing tools operate mostly with lists or sets of cards to choose from, or lack examples of real use cases and interaction with them, like notation and annotation, which adds an unnecessary level of abstraction to the learning process. There are commercial products, such as NoteFlight [see NoteFlight (2008)] and Flat [see Flat.io (2015)], which offer comprehensive music notation for LMSs, but at the moment there is no free, open-source solution available that limits the impact of the community to design and implement new types of tasks.

In an ongoing project, we are developing such an interface for music notation for LMSs. Based on that, it will be possible to support a breadth of quizzes and music theory exercises: from entering single notes, over harmonizing a melody, to writing (small) polyphonic pieces. With such a comprehensive undertaking, the applications running on the platforms not only have to be functional, but there also needs to be a certain degree of overall conceptualization, which leads to design choices and to support or improved learning outcomes. In this paper, we discuss the requirements, technical basis, and design considerations of our proposed system.

2. REQUIREMENT ANALYSIS

Learning platforms such as Moodle, Drupal, and Canvas are widely used to support blended learning, flipped classrooms, or to realize massive open online courses. These LMSs have been used in professional music education, but only to a limited extent because they are not primarily designed for music pedagogical purposes. Music-specific tools lack requirements especially regarding comprehensive music notation and adequate audio processing, like Musical Instrument Digital Interface (MIDI) or MP3 replay or taking audio input.

Innovative teaching and learning approaches require tools that enhance current teaching practices. For example, one can enhance lecture recordings with interactive elements or present concepts graphically with timelines, image sliders, or other presentations. Furthermore, LMSs can help one to practice more effectively (e.g., with image hotspots, flashcards, or questionnaires) or to practice in a more playful manner (e.g., with memory games or various forms of quizzes). These functionalities are commonly available ways of interaction in a standard LMS but are not yet implemented for music-related tasks, unless specialized and LMS provider exclusive plug-ins are downloaded.

To enable truly sophisticated teaching in the field of music, an LMS should provide options for presentation and practice encompassing different sensory approaches to music, including sound, notation, analysis, playing and singing. These multisensory approaches can be

addressed by integrated tools that evaluate music notation tasks, analyze and assess simple instrument performance tasks, enable automatic synchronization of scores and music recordings, or provide rich listening exercises, among other things.

The creation of such learning content is currently very laborious and time-consuming. To create the required materials, one must use external third-party software to make the scores, audio files, etc., that one then needs to upload and integrate into the LMS. The music notation is then not interactive, which means that a large part of the abovementioned tasks cannot be realized. This makes the creation of learning content unattractive for teachers and is a major obstacle to their willingness to create such materials, which could be widely shared as open educational resources (OER).

To make significant progress, a music notation interface is needed that is integrated into the platform and enables the immediate creation and editing of notations and sound samples. This notation environment should include, in addition to Common Western Music Notation (CWMN), display options for early music and, to a certain extent, new music. Furthermore, the notation environment should provide means for harmonic description and analysis for chords, functions, steps, basso continuo, partimento, form progressions, etc. At the same time, the described functional enhancements form the foundation for future developments, such as applications for composition and improvisation tasks, algorithm-based training tools, integration of sequencer functionalities, or basic audio editing tools.

Another requirement is the support of audio. Audio helps to provide a varied and dynamic experience to help enhance reflection and self-check on the created solutions (Silveira & Gavin, 2016). Also, there should be a low-latency audio rendering during note entry. Online notation should be perceivable and verifiable aurally during input and should sound synchronously with the notation display during playback. For that purpose, standard MIDI sound can be used or, for even higher quality, sounds from dedicated sound libraries.

For demanding aural training or for music analysis, actual music recordings should be supported. This includes uncompressed or low-compressed sound files (WAV, AIFF, or MP3 with 360 kbps). Additionally, it should be possible to synchronize the audio files with a score created with the music editor. For this purpose, we can use established algorithms from music information retrieval, like dynamic time warping (Waloschek & Hadjakos, 2018; Müller, 2015).

3. TECHNICAL BASIS

The technical requirements described in Section 2 were already implemented, in part, in the *detmoldmusictools* learning platform (DET, 2013) for the Detmold University of Music in 2013 (Fig. 1). After five years, however, these must now be brought to a new level of quality and made available to users of commonly used LMSs, such as Moodle or Canvas and others, like the Integrated Learning, Information and Work Cooperation System (ILIAS; German: Integriertes Lern-, Informations- und Arbeitskooperations-System), which is mostly popular in the German speaking area.

Bestimme die Tasten



In der folgenden Übung kannst Du jede Taste entweder als Stammtton, als Erhöhung oder Erniedrigung beschreiben.

(In England und Amerika macht man sich die Benennung der Töne etwas leichter. Dort hängt man kein "is" oder "es" an die Noten sondern schreibt das Kreuz oder b direkt hinter die Stammtöne, z. B. C#, F# oder Ab, Bb. Das spricht man dann C sharp, F sharp, A flat und B flat.)

Benenne die Tasten.

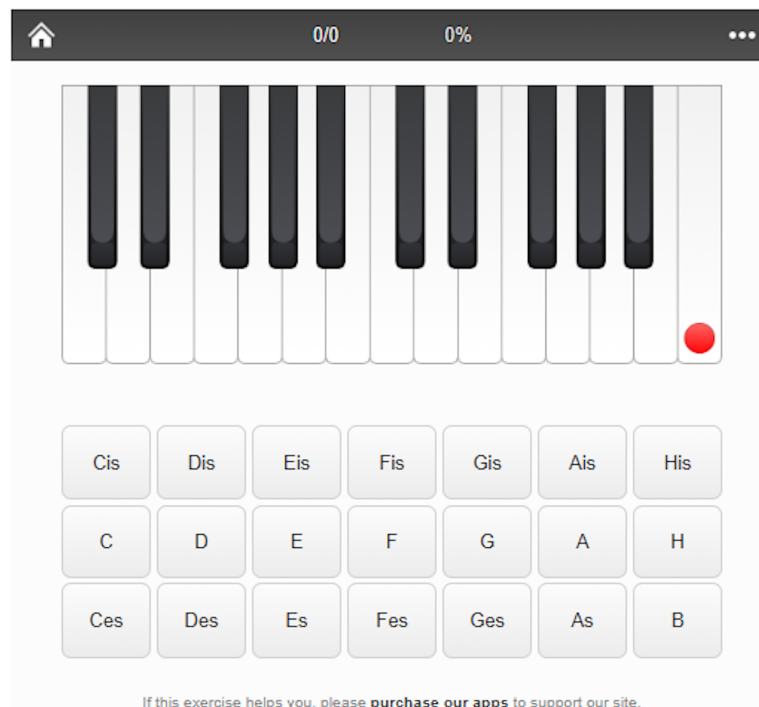


FIG. 1: Example application within the *detmoldmusictools*. In this case, the students must name keys of a piano keyboard. The platform supports further applications, such as learning notes, repertoire knowledge, or preparation for qualification tests at the Detmold University of Music. The current status of these applications should be the baseline for our project.

Qualitative and quantitative expansion of notation options including audio output are needed, as well as support for MusicXML, Music Encoding Initiative (MEI) [see MEI (1999)] and further formats.

The open-source notation engraving library, *Verovio*, is the basis for our notation interface. *Verovio* is used by more and more scientific institutions, maintained by a community of developers, and thus proves to be a sustainable tool. An example for fruitful collaboration between an online editor and the *Verovio* library would be *Neon2*. It is just available for neumes texts, but it helped heavily to improve the editing possibilities in *Verovio*, which remain to be implemented for CWMN (Regimbal et al., 2019; Pugin, 2018).

The benefits in using the open-source MEI format as a pivot point for our editing framework is the underlying XML markup structure, which encodes the music in a declarative way. This means that we have no information about the exact layout within XML nodes but do about the

hierarchical representation of the later rendered score. Although the markup can still be overly complex, this makes it highly human readable, since most numerical parameters are removed. This also simplifies the computational parsing process and, respectively, its implementation. MEI lets the user encode different versions of a musical text (the so-called readings). This feature is commonly used in digital music editions but can also be used to support tasks such as quizzes, in which students must distinguish between predefined score excerpts. MEI is not restricted to CWMN, but it also supports different notation types, such as neumes or mensural notation, and it is generally open for development of new notation types, which could be used for compositional tasks (Pugin, 2018).

To interact with the SVG, we need to map this format to our underlying XML structure. For this, we also must focus on parsers to ensure that every editing step will be also reflected in both formats. Although *Verovio* supports a variety of XML structures, concentrating on implementing parsers only for MEI will be sufficient, since *Verovio* internally converts every supported format, which will reduce development effort (Fig. 2).

Although *Verovio* employs a suitable renderer for MEI, it does not have a graphical user interface. This functionality must be programmed so that users (content creators and learners) can enter and edit notes graphically using a mouse, stylus, visual keyboard, and other buttons. In particular, the following musical parameters must be editable in this way:

- pitch, tone position, and tone duration
- keys and accidentals
- time signatures, bar lines, mensuration signs, and formal progress signs
- staves, clefs, polyphonic representation, and chords
- articulation signs and dynamic indications
- lyrics and free text
- sound source and sound selection
- various chord designation procedures and annotations
- metadata (comments, annotations, etc.)

To achieve interoperability with current LMSs, H5P [see Joubel (2013)] will be used as a plug-in architecture (Fig. 3). H5P is supported by most of the existing LMS platforms. Not only does it already provide multiple applications for interactive content, but it also enables one to extend these applications for given needs. The applications are usable out of the box and do not require any programming skills.

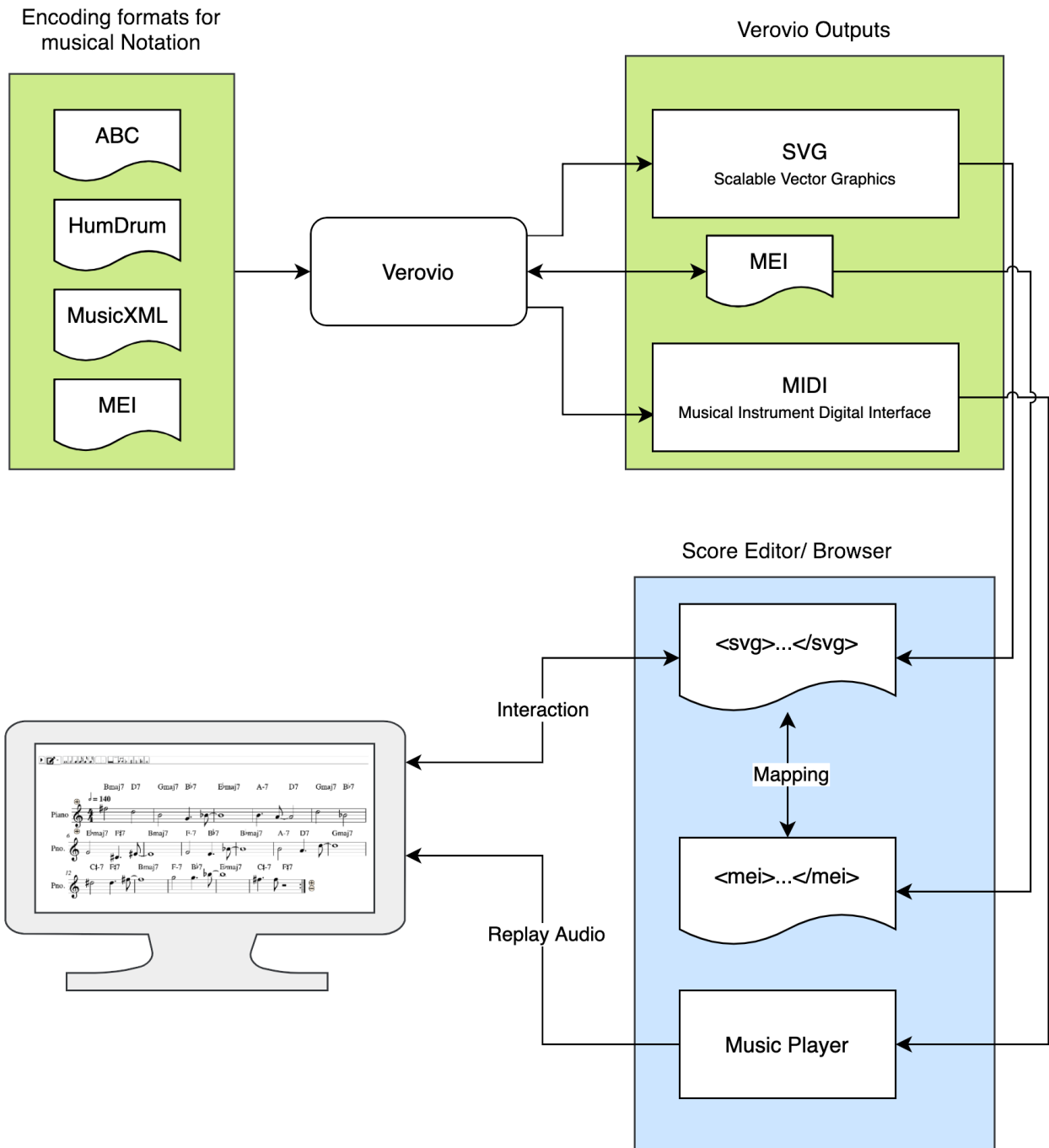


FIG. 2: Procedures and dependencies of *Verovio*. The library takes various data formats as input, which then can be transformed into several, interactable representation in the score editor.

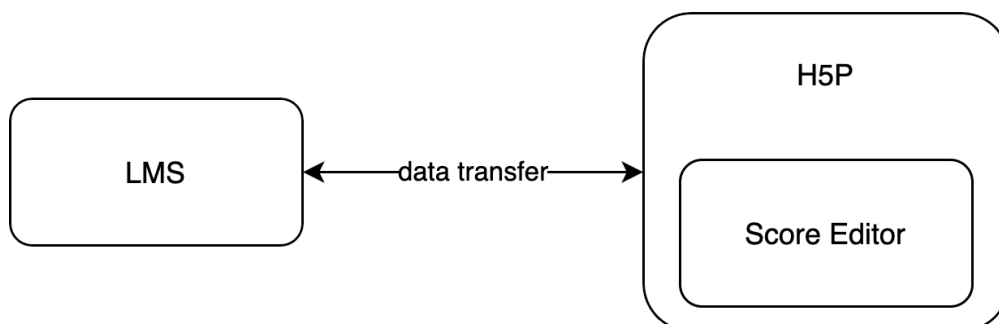


FIG. 3: H5P will encapsulate the Score Editor and provide data transfer and interaction with the LMS, such as for task evaluation and reporting results

H5P also bears the advantage that it is programmable in either plain Javascript or it can employ different frameworks, such as React, Vue.js, or Node.js. In our case, we found it quite useful to develop in Typescript and transpiling it to Javascript afterward, since it gives us better control over outputs and dependencies between the files.

4. DESIGN CONSIDERATIONS

Learning music theory does not include the learning of the underlying technology. In fact, learning a new technology means investing more time to even accomplish easier tasks created by the teacher, which leads to frustration and ultimately nonacceptance of the LMS. Avoiding this is the goal of human centered design, which is the process that ensures that the design matches the needs and capabilities and builds on the experience of the people for whom they are intended (Norman, 2013). Each interaction with previously unknown technologies or interfaces therefore could cause self-attribution for erroneous usage, reducing the feeling of control over the given task, which we also call “frustration.” In a pedagogical context, we see any design consideration as working in favor of the constructivist concept of self-efficacy, which describes ones beliefs about ones capabilities to perform actions that influence their lives (Bandura, 1993). We therefore suspect that designing software in a learner-centered fashion improves learning outcomes (Schmidt et al., 2020).

We take into account that students are familiar with notating tasks using pen, paper, and if necessary, a musical instrument such as a piano keyboard. Many might also have some experience with common score editors, such as *Sibelius*, *Finale*, *Dorico*, or *MuseScore*, which are extensively analyzed [see MartinTheKearyKid (2009)]. These are full-fledged editors that have their own design requirements, such as providing comprehensive notation possibilities for power users, flexible orchestration, navigation in large scores, or voice extraction. For education purposes, they seem overpowered but can represent guidelines for common (inter-)actions, commonly used symbols, and user expectations. Questions in the design process are (but are not limited to) the following:

- How are notes edited?
- When does the user expect to use keyboard or mouse input?
- What happens to the context when something is deleted?

For our tools, it is necessary to isolate and reduce these methods and test certain configurations to make it as user friendly as possible and so acceptance among students increases, as exemplified by the design of a gamification app for ear training (Pesek et al., 2020).

A music or score editor can be viewed as a text editor with special demands, since music and text both are notated as a time-dependent sequence. This stems from the observation that music scores in their prescriptiveness are designed to be translated into action and are

realized in time. But in hearing, reading, and analyzing music, the vertical dimension comes into play which lets us distinguish between harmonies, instruments, and counterpoints. This raises the question, for what amounts vertical or sequential thinking influences the act of solving a task. For our tools, we would therefore like to design the tools in multiple ways:

- Mouse input
- Keyboard input
- MIDI device input
- Combinations of the above

By comparing these input methods, we will filter through the most efficient way to solve the respective tasks. We assume to have a focus on mouse and keyboard inputs, since these are the most widely available input methods and enable users to work from home. Especially, the lack of mouse input and feedback with laptop touchpads could be a motivation to develop more sophisticated keyboard input methods. To collect data for our studies, we can make use of many potential users, as the project is supported by several stakeholders.

5. FUTURE PERSPECTIVES

As we implied earlier, these tools will at first cover one dimension of a student-centered teaching model only through interactive activities, as teacher and students will not interact with each other directly (Johnson, 2020). This does also mean that any collaboration between students is not yet reflected in our project. To overcome this one-sidedness, some solutions, as follows, would be worth thinking about:

- Unsorted (not graded) and anonymous display of solutions of more complex composition and harmony tasks as well as commenting could be a way to enable some sort of interaction between students and between students and the teacher.
- Exploring the subject through wikis or videos by teachers or students integrated into the LMS.
- In a field that is heavily dependent on explicit memory, a certain number of teacher-provided tasks could be insufficient to explore the subject by the student in a self-managed way. To provide an adequate number of tasks, it would be sufficient to create logic to generate and evaluate tasks automatically, where for example machine learning approaches could be employed to generate such sequences. In a creative environment—as is music—this also raises the question of how much of an error margin we can program to label a solution as “correct” or “acceptable.”

We modeled our prototypical user as sitting in front of a desktop PC and using mouse and keyboard. We omitted the use of touch interfaces (such as smartphones, tablet computers, and Wacom-tablets) since developing proper rules for recognizing notation has its own challenges. Although our user most likely will not use a touch device, we would like to cover

most cases to work with the tools at some point. In the case of graphical input, we could already benefit from more than 20 years of research using dynamic programming and machine learning (Forsberg et al., 1998; Miyao & Maruyama, 2004; Lee et al., 2016).

Developing these new tools started in January 2021, so they are in their infancy and should be developed incrementally. By building a framework with Typescript within the H5P-Plugin, we are not only enable technological accessibility for other programmers, since knowledge in developing plugins for an LMS is not necessary, but we will also make this project publicly accessible, maintainable, and sustainable by uploading it to a GitHub and NPM repository. The code for the H5P package as well as installation instructions are available (Nowakowski, 2021a,b). The underlying score editor can also be downloaded and integrated into web projects independent of the H5P package as a NPM module (Nowakowski, 2021c). As for now, we managed to find a fitting development environment with the help of Drupal 7, to dynamically develop and review our outcomes. We started from entering and editing (deleting, dragging) monophonic sequences, because they are the most basic actions. In practice, however, it is necessary to enable this via several systems in parallel.

6. CONCLUSION

The main challenge will be first to implement a functional score editor, which fulfills all requirements. Especially parsing the SVG and mapping the graphical elements to the MEI will be the core technical foundation. Based on that, the graphical interface will be implemented based on a human-centered design methodology. We hope that presenting this project in an early stage will evoke feedback from technically interested people as well as practitioners in music education. As for now, we see the following added values of this project:

- **Contemporaneity** – Through the development of the notation environment *Verovio* and its integration via H5P into the LMS used at the universities, students now also will have a tool available that meets today's requirements for notation and annotation options, which eliminates the need for costly purchases. At the same time, the same tool can now be used in all courses, eliminating the need for cumbersome familiarization with new systems.
- **Compatibility** – Teachers can continue to use their previous notations with *Sibelius*, *Finale*, or *Dorico* through the included file import types MusicXML, HumDrum, MEI, and ABC and now prepare them interactively for teaching.
- **Availability** – By integrating the H5P framework into the LMS, its functionalities are immediately available to all institutions. A high degree of sustainability can be assumed, because H5P is growing worldwide, adapted to new operating systems and browsers at regular intervals, and optimized for mobile devices.
- **Expandability** – Through the measures described earlier, further H5P applications can be developed in the future with significantly less effort and thus adapted to the

state of the art. For this purpose, an application for form analysis, an exercise tool based on algorithmic specifications, an application for controlling instrumental playing, and notation input via pen are planned.

REFERENCES

- Bandura, A. (1993). Perceived self-efficacy in cognitive development and functioning. *Educational Psychologist*, 28(2), 117–148.
- Blackburn, A. & Hewitt, D. (2020). Fostering creativity and collaboration in a fully online tertiary music program. *International Journal on Innovations in Online Education*, 4(2). Retrieved from <https://onlineinnovationsjournal.com/streams/visual-and-performing-arts/63209b3a0ef5fedf.html>
- Bowman, J. (2014). *Online Learning in Music: Foundations, Frameworks, and Practices*. Oxford and New York: Oxford University Press.
- Brisson, E. (2019). Music theory. Moodle Pty Ltd. Retrieved from https://moodle.org/plugins/qtype_musictheory.
- DET. (2013). Detmoldmusictools. DET Mold Music Tools. Retrieved from <https://detmoldmusictools.de>.
- Forsberg, A., Dieterich, M., & Zeleznik, R. (1998). The music notepad. *Proceedings of the 11th annual ACM symposium on user interface software and technology – UIST '98*, New York, ACM Press. 203–210.
- Johnson, C. (2020). A conceptual model for teaching music online. *International Journal on Innovations in Online Education*, 4(2). Retrieved from <https://onlineinnovationsjournal.com/streams/visual-and-performing-arts/4cc75e683a8f5c22.html>.
- Joubel. (2013). H5P Forum. Joubel AS. Retrieved from <https://www.h5p.org>
- Lee, H.-L., Gong, S.-J., & Chen, L.-H. (2016). An online handwritten recognition system of musicscore. *2016 International Conference on Machine Learning and Cybernetics (ICMLC)*, IEEE. 552–557.
- MartinTheKearyKid. (2009). Tantacru YouTube channel. Retrieved from <https://www.youtube.com/user/martinthekearykid>.
- MEI. (1999). Music encoding initiative. Academy of Sciences and Literature, Mainz. Retrieved from <https://music-encoding.org>.
- Merrick, B. (2020). Changing mindset, perceptions, learning, and tradition: an adaptive teaching framework for teaching music online. *International Journal on Innovations in Online Education*, 4(2). Retrieved from <http://onlineinnovationsjournal.com/streams/visual-and-performing-arts/0adeed9b5d34eeae.html>.
- Miyao, H. & Maruyama, M. (2004). An online handwritten music score recognition system. *Proceedings of the 17th International Conference on Pattern Recognition*, IEEE. 461–464.
- Müller, M. (2015). *Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications*. Cham: Springer International Publishing and Imprint and Springer.

- Nikolsky, T. (2020). Challenges and opportunities in teaching VCE music at virtual school Victoria. *International Journal on Innovations in Online Education*, 4(2). Retrieved from <http://onlineinnovationsjournal.com/streams/visual-and-performing-arts/4da1ddd71017cf19.html>.
- Norman, D.A. (2013). *The Design of Everyday Things*. New York: Basic Books, revised and expanded edition.
- NoteFlight. (2008). NoteFlight. Retrieved from <https://www.noteflight.com>.
- Nowakowski, M. (2021a). Score4LMS. GitHub, Inc. Retrieved from <https://github.com/mnowakow/Score4LMS>.
- Nowakowski, M. (2021b). Verovio score editor. GitHub, Inc. Retrieved from <https://www.npmjs.com/package/verovioscoreeditor>.
- Nowakowski, M. (2021c). VerovioScoreEditor. NPM, Inc. Retrieved from <https://www.npmjs.com/package/verovioscoreeditor>.
- Osborne, M.S. (2020). Entering the live-streaming void and emerging victorious: teaching performance psychology under pressure. *International Journal on Innovations in Online Education*, 4(2). Retrieved from <https://onlineinnovationsjournal.com/streams/visual-and-performing-arts/2c5f8445632bbdf2.html>.
- Pesek, M., Vucko, Z., Savli, P., Kavcic, A., & Marolt, A. (2020). Troubadour: a gamified e-learning platform for Ear Training. *IEEE Access*, 8, 97090–97102.
- Pike, P.D. (2020). Preparing an emerging professional to teach piano online: a case study. *International Journal on Innovations in Online Education*, 4(2). Retrieved from <https://onlineinnovationsjournal.com/streams/visual-and-performing-arts/26ae6d4f6cbba1d3.html>.
- Pugin, L. (2018). Interaction perspectives for music notation applications. *Proceedings of the 1st International Workshop on Semantic Applications for Audio and Music*. New York. 54–58.
- Regimbal, J., McLennan, Z., Vigiensoni, G., Tran, A., & Fujinaga I. (2019). Neon2: A Verovio-based square-notation editor. *Music Encoding Conference*. Vienna.
- Schmidt, M., Tawfik, A.A., Jahnke, I., & Earnshaw, Y. (2020). *Learner and User Experience Research: An Introduction for the Field of Learning Design & Technology*. EdTechBooks.
- Silveira, J.M. & Gavin, R. (2016). The effect of audio recording and playback on self-assessment among middle school instrumental music students. *Psychology of Music*, 44(4), 880–892.
- Flat.io. (2015). Tutteo Limited. Retrieved from <https://flat.io/de>.
- Waloschek, S. & Hadjakos, A. (2018). Driftin' down the scale: Dynamic time warping in the presence of pitch drift and transpositions. *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR*, Paris, France. 630–636.