



# OSMD Audio Player

Endbericht | Call 14 | Projekt ID 4605

Lizenz: CC-BY-SA

# Inhalt

<b>Einleitung</b>	<b>3</b>
<b>Projektbeschreibung</b>	<b>3</b>
<b>Verlauf der Arbeitspakete</b>	<b>4</b>
Arbeitspaket 1 - Detailplanung und Formales am Projektstart	4
Arbeitspaket 2 - Projektmanagement	4
Arbeitspaket 3 - Multithreading	4
Arbeitspaket 4 - Selektierbare Noten	4
Arbeitspaket 5 - Automatisches Scrolling	5
Arbeitspaket 6 - Wiederholungszeichen	5
Arbeitspaket 7 - Implementierung eines Playback Threads	6
Arbeitspaket 8 - Implementierung eines Metronoms mit Einzähler	6
Arbeitspaket 9 - Dokumentation und Formales am Projektende	6
<b>Umsetzung Förderauflagen</b>	<b>7</b>
<b>Liste der Projektergebnisse</b>	<b>7</b>
<b>Verwertung der Projektergebnisse in der Praxis</b>	<b>8</b>
<b>Öffentlichkeitsarbeit/ Vernetzung</b>	<b>8</b>
<b>Eigene Projektwebsite</b>	<b>8</b>
<b>Geplante Aktivitäten nach netidee-Projektende</b>	<b>9</b>
<b>Anregungen für Weiterentwicklungen durch Dritte</b>	<b>9</b>

## 1 Einleitung

Bei digitalen Noten ist es möglich ein Playback zu generieren, wie es auch bereits in vielen Musikprogrammen und Apps gezeigt wird. Durch das Vorspielen fällt es vielen Menschen deutlich leichter den Rhythmus der Melodie bzw. der Akkordbegleitung zu verstehen und aufzunehmen. Auch die "Gefühlswelt" und Stimmung, die ein Stück vermittelt kann so auch für Anfänger leicht erfasst werden.

Für das Üben eines Stücks ist es ebenso ein großer Vorteil zur vorgespielten eigenen Stimme dazu zu spielen, da man jederzeit akustisch darauf aufmerksam gemacht wird, welcher Ton gerade zu spielen wäre. Umgekehrt kann es auch sehr motivierend sein alle anderen Stimmen (bzw. die Begleit-Akkorde) während dem Üben zu hören (Üben mit Play-Along Begleitung), da man so die Harmonien des Stücks mit dem Gehör erfasst und einlernt und so auch viel mehr Freude am Üben hat.

## 2 Projektbeschreibung

Der Audio-Player setzt als eigenständiges Produkt im Rahmen unserer interaktiven, erweiterbaren Web-Komplettlösung (OpenSheetMusicDisplay; OSMD) für die Darstellung von Musiknoten im Webbrowser auf. Im Rahmen dieses Projekts soll OSMD um ein interaktives Playback mit Kontrolloptionen wie z.B. Geschwindigkeit, Lautstärke der einzelnen Stimmen, Wahl der Startposition, automatische Mitscrollen usw. erweitert werden.

Die Zielgruppe für den OSMD Audio-Player sind Unternehmen, die abspielbare Notenblätter im Webbrowser (und in weiterer Folge auch für native mobile Apps) als wirtschaftliche Grundlage für ihre Produkte benötigen. Dazu zählen vor allem Musik-Lernportale und -Lernspiele, Übungs-Apps, Notenverlage und Online-Shops.

## 3 Verlauf der Arbeitspakete

### 1.1 Arbeitspaket 1 - Detailplanung und Formales am Projektstart

Das AP wurde planmäßig durchgeführt.

### 1.2 Arbeitspaket 2 - Projektmanagement

Durchführung von internen Meetings, Protokollen, Projektcontrolling.

AP 2 wurde planmäßig zu 100% durchgeführt.

### 1.3 Arbeitspaket 3 - Multithreading

Implementierung von Multithreading der Anzeige und des Playbacks mittels Webworker Threads.

Mittels Webworker werden im Hintergrund des Browsers Thread-ähnliche Strukturen zur Verarbeitung von parallel durchzuführenden, rechenintensiven Aufgaben ermöglicht, dh die komplexen Berechnungen für die Tonwiedergabe beeinträchtigen nicht die Usability und halten die Webseite responsive.

Das Herausforderung hierbei ist der Austausch der Daten zwischen dem Browser-Main-Thread und den Webworker-Threads, da immer nur ein Thread auf die Daten zugreifen darf, um gleichzeitiges Verändern der Daten zu verhindern. Für die Berechnung der vertikalen Position im Notenblatt wurde hier ein Messaging zwischen Main-Thread und Worker verwendet, wo nur die notwendigen Daten übergeben werden, die für die Berechnung der Position gebraucht werden.

AP 3 wurde vollständig durchgeführt. Es wurden sogenannte Webworker geschaffen, die im Browser die rechenintensiven Aufgaben wie zB. der Tonwiedergabe parallel durchführen.

### 1.4 Arbeitspaket 4 - Selektierbare Noten

Implementierung von selektierbaren Noten. Modellierung von Bildschirmkoordinaten und Koordinaten am Notenblatt sowie BoundingBoxes der Notenobjekte.

Die Klickposition am Bildschirm wurde auf die Notenblattposition umgerechnet. Abspielen der selektierten Noten.Über einen Klick am Schirm kann die dort befindliche Notenposition

bestimmt werden und somit der Cursor an diese Stelle gesetzt bzw. die zugehörigen Noten abgespielt werden.

AP 4 wurde zu 100% fertiggestellt. Noten können am Bildschirm angeklickt werden. Dadurch wird der Cursor gesetzt und die geklickte Note abgespielt. Es folgt Integration in das Playback.

Das Optimierungsproblem hierbei betrifft die Performance: Um nicht für eine angeklickte Bildschirmposition das gesamte Datenmodell der grafischen Objekte durchsuchen zu müssen, kann hier durch Optimierungen wie Filter auf Objekttypen (z.B. eine Note oder ein Notenschlüssel) und deren Eigenschaften die Performance verbessert werden. Dadurch ist es auch möglich bei bestimmten Modi nur bestimmte Objekttypen anklicken zu können.

### **1.5 Arbeitspaket 5 - Automatisches Scrolling**

Berechnung der Position für das automatische Scrolling.

Automatisches Scrollen der Anzeige, abhängig von der Playback-Position, inklusive Rücksprung bei Wiederholungen.

AP 5 wurde zu 100% fertig gestellt. In diesem AP wird Code generiert, der sicherstellt, dass das Notenblatt auf dem Bildschirm automatisch mit der Playbackposition mitgescrolled wird inklusive Rücksprung bei Wiederholungen (siehe auch AP 6).

Hierbei wird über die aktuelle Cursorposition, welche über eine hochzählende Zeitvariable bestimmt wird, das aktuelle System im Notenblatt bestimmt. Durch den vorher durchgeführten Layout-Prozess des gesamten Notenblattes hat dieses nun eine definierte vertikale Position. Das Display kann nun bei jedem Cursor-Update auf die berechnete System-Position geschoben werden. Beim Sprung zum nächsten System soll ein weicher Übergang der Display-Position berechnet werden.

### **1.6 Arbeitspaket 6 - Wiederholungszeichen**

Einlesen und Interpretieren von Wiederholungszeichen (spezielle Taktlinien, da capo, dal segno, al fine). Anzeige und Playback reagiert auf Wiederholungszeichen.

AP 6 wurde zu 100% durchgeführt. In diesem AP wird Code generiert, der auf die verschiedenen Wiederholungszeichen reagiert, Notenblatt und Cursor an die korrekte Stelle "schiebt" und an dieser Position wieder die Audiowiedergabe startet.

Wiederholungszeichen werden bereits korrekt eingelesen und im internen Musik-Objektmodell verarbeitet, um letztendlich Wiederholungen korrekt zu erkennen und interpretieren zu können. Der Iterator, der die Folge der Noten erstellt, kann somit die Wiederholungszeichen korrekt verarbeiten, und erstellt demnach Notensequenzen inklusive der Wiederholungen für das Playback.

### **1.7 Arbeitspaket 7 - Implementierung eines Playback Threads**

Wartezeiten in Millisekunden zwischen aufeinanderfolgenden Noten werden korrekt berechnet und abgewartet. Dann werden Samples abgespielt und die Position im Notenblatt gesetzt. Die Töne werden gleichzeitig zu den gezeigten Noten abgespielt.

AP 7 wurde zu 100% durchgeführt. In diesem AP wird Code generiert der die Zeitspanne der Noten korrekt berechnet um dann die Audio Samples abzuspielen. Dadurch wird der Effekt generiert, dass gleichzeitig zu den Noten die Töne abgespielt werden.

Die Planung und Tests hierzu haben ergeben, dass die Berechnung der Zeitspannen bis zur nächsten zu spielenden Note auf eine unabhängig laufende hochzählende Zeitvariable bezogen werden muss, um keine Schwankungen und Verzögerungen durch die Berechnungszeiten zu bekommen.

### **1.8 Arbeitspaket 8 - Implementierung eines Metronoms mit Einzähler**

AP 8 wurde zu 100% durchgeführt und wie geplant abgeschlossen. In diesem AP wird Code generiert, der einerseits spezielle Audiodateien als Metronom abspielen kann und auf Basis des gegebenen Taktes des Musikstückes einen (1) Takt vor den sichtbaren Beginn des Musikstückes setzt. Beim Abspielen des Musikstückes wird nun zuerst der vorangestellte Takt abgespielt.

Sowohl das Metronom als auch der Einzähler kann vom Entwickler optional verwendet werden.

### **1.9 Arbeitspaket 9 - Dokumentation und Formales am Projektende**

AP 9 wurde mit Verspätung begonnen und abgeschlossen. Grund dafür waren fehlende Ressourcen des Projektteams, da parallel noch weitere Entwicklungsprojekte durchgeführt wurden, wie zB ein OSMD Wordpress Plugin.

Wie vorgegeben wurden Endbericht, sonstige Dokumentationen und der eigentliche Audio Player als Projektergebnisse auf die netidee Projektsite hochgeladen.

## 4 Umsetzung Förderauflagen

Es gab keine speziellen Förderauflagen.

## 5 Liste der Projektergebnisse

1	Projektzwischenbericht	CC-BY-SA	<a href="https://www.netidee.at/sites/default/files/2020-07/prj4605_call14_Zwischenbericht_Projekte_V02.pdf">https://www.netidee.at/sites/default/files/2020-07/prj4605_call14_Zwischenbericht_Projekte_V02.pdf</a>
2	Projektendbericht	CC-BY-SA	<a href="https://www.netidee.at/osmd-audioplayer">https://www.netidee.at/osmd-audioplayer</a>
3	Entwickler_innen Dokumentation	CC-BY-SA	<a href="https://github.com/opensheetmusicdisplay/opensheetmusicdisplay/wiki">https://github.com/opensheetmusicdisplay/opensheetmusicdisplay/wiki</a>
4	Anwender_innen Dokumentation	CC-BY-SA	<a href="https://wordpress.org/plugins/opensheetmusicdisplay/">https://wordpress.org/plugins/opensheetmusicdisplay/</a>  Anwender_innen sind in unserem Fall WordPress Plugin User.
5	Veröffentlichungsfähiger Einseiter	CC-BY-SA	<a href="https://www.netidee.at/osmd-audioplayer">https://www.netidee.at/osmd-audioplayer</a>
6	Dokumentation Externkommunikation	CC-BY-SA	<a href="https://www.netidee.at/osmd-audioplayer">https://www.netidee.at/osmd-audioplayer</a>
7	OSMD Audio Player	BSD-3-clause	<a href="https://github.com/opensheetmusicdisplay/opensheetmusicdisplay">https://github.com/opensheetmusicdisplay/opensheetmusicdisplay</a>

## 6 Verwertung der Projektergebnisse in der Praxis

Der Audio Player für OSMD kann für verschiedenste Anwendungen verwendet werden, bei denen das Anhören von Musiknoten sinnvoll ist:

- Digitale Anwendungen im Musikunterricht
- Anwendungen zum Lernen von Instrumenten
- Websites von Verlage bzw. Notenplattformen

## 7 Öffentlichkeitsarbeit/ Vernetzung

Die Vernetzung vor allem mit Entwicklern, die OSMD für die Umsetzung ihrer Projekte verwenden oder in Betracht ziehen, erfolgt größtenteils über unsere Organisationseite GitHub.com (<https://github.com/opensheetmusicdisplay>). Vereinzelt kommen Anfragen über unsere Facebook Seite oder direkt per Email. Diese werden aber dann gebeten, ihre Anliegen ebenfalls öffentlich auf Github zu formulieren, damit auch die Community davon lernt.

Seit Sommer 2021 bieten wir den Entwicklern auf der Kommunikationsplattform "Discord" einen direkten Zugang zu OSMD, dh. es findet ein unkompliziert, direkter Austausch statt. Unter folgendem Link kann man beitreten: <https://discord.gg/>

Der Audio Player wurde natürlich auch im Blog unserer Website (<https://opensheetmusicdisplay.org/osmd-blog/>) sowie in unseren sozialen Kanälen kommuniziert:

Instagram: [https://www.instagram.com/open\\_sheet\\_music\\_display/](https://www.instagram.com/open_sheet_music_display/)

Facebook: <https://www.facebook.com/opensheetmusicdisplay/>

Twitter: <https://twitter.com/osmdengine>

Youtube: [https://www.youtube.com/channel/UHCMvITdC1dX2esPXDe\\_7Hw](https://www.youtube.com/channel/UHCMvITdC1dX2esPXDe_7Hw)

## 8 Eigene Projektwebsite

Die eigene Projektwebsite ist unter <https://opensheetmusicdisplay.org> zu finden und wurde im Sommer 2021 einem Relaunch unterzogen.



## 9 Geplante Aktivitäten nach netidee-Projektende

Nach Projektende ist geplant, den Audio Player weiter zu verbessern und um Features zu erweitern. Welche das genau sein werden, richtet sich nach den Anforderungen der Entwicklercommunity.

Zusätzlich sollen auch verstärkt Nicht-Entwickler (No Code!) angesprochen werden. Dazu wurde bereits ein Plugin für WordPress entwickelt (siehe <https://opensheetmusicdisplay.org/wp-plugin/>).

Für sämtliche geplanten Aktivitäten ist jedoch eine entsprechende Finanzierung nötig. Als Geschäftsmodell testen wir derzeit gerade drei Ansätze parallel:

- Individuelle Anpassung der Notenanzeige bzw. des Audio Players gegen Bezahlung. Dies ist vor allem für kommerzielle Projekte interessant
- Freemium Modell bei WordPress Plugin: Bestimmte Features sind nur gegen Bezahlabo verfügbar
- [Sponsorware](#) Modell für TypeScript Bibliothek: Bestimmte Features der Entwickler Library sind nur mit einem aktiven [GitHub Sponsorship](#) zugänglich.

Erste Erfahrungen mit diesen Ansätzen waren sehr vielversprechend. Wir sind zuversichtlich den OSMD Audio Player in Zukunft aus eigener Kraft weiterentwickeln zu können.

Zusätzlich werden die OSMD Social-Media Kanäle bedient.

## 10 Anregungen für Weiterentwicklungen durch Dritte

Die am meisten nachgefragte Weiterentwicklung ist eine native mobile Lösung für OSMD inklusive Audio Player. Bevorzugt mit React Native umgesetzt.