

Archivist



Project Documentation

1. Semesterprojekt SS 2017

Autoren
Boris Fuchs
Matr.-Nr.: S1610237004

Paul Schmutz
Matr.-Nr.: S1610237028

Inhaltsverzeichnis

Inhaltsverzeichnis	2
1 Kurzbeschreibung	4
2 Benutzeranleitung	5
2.1 Installationsanleitung	5
2.1.1 Installation	5
2.1.2 Deinstallation.....	5
2.2 Tutorial	6
2.2.1 Eingabe des Benutzernamens.....	6
2.2.2 Anlegen eines neuen Musikstücks/Noten	6
2.2.3 Suchen und Bearbeiten eines Musikstücks	9
2.2.4 Herunterladen von Noten	11
2.2.5 Löschen von Musikstücken und Noten	11
2.2.6 Sortierte Ansicht	13
3 Poster	14
4 Pflichtenheft	15
4.1 Einleitung	15
4.1.1 Zweck des Dokuments	15
4.1.2 Gültigkeit des Dokuments.....	15
4.1.3 Begriffsbestimmungen und Abkürzungen	15
4.1.4 Zusammenhang mit anderen Dokumenten.....	16
4.1.5 Überblick über das Dokument	16
4.2 Allgemeine Beschreibung des Produkts	17
4.2.1 Projektumgebung	17
4.2.2 Zweck des Produkts	18
4.2.3 Überblick über die geforderte Funktionalität	18
4.2.4 Allgemeine Einschränkungen	18
4.2.5 Vorgaben zu Hardware und Software	19
4.2.6 Benutzer des Produkts	19
4.3 Detaillierte Beschreibung der geforderten Produktmerkmale	19
4.3.1 Lieferumfang	19
4.3.2 Abläufe (Szenarien) von Interaktionen mit der Umgebung.....	20
4.3.3 Externe Schnittstellen des Produkts	21
4.4 Vorgaben an die Projektabwicklung.....	26
4.4.1 Anforderungen an die Realisierung	27
4.4.2 Fertige und zugekaufte Komponenten	29
4.4.3 Unterauftragnehmer	29
4.4.4 Lieferbedingungen.....	30

4.5	Verpflichtungen des Auftraggebers.....	30
5	Präsentation	31
6	Zeitaufzeichnung.....	37
7	Meetings.....	38
7.1	1. Protokoll	38
7.2	2. Protokoll	41
7.3	3. Protokoll	43
7.4	4. Protokoll	45
7.5	5. Protokoll	47
7.6	6. Protokoll	49
8	Detaillierte Systemdokumentation	51
8.1	Systemschnittstellen.....	51
8.2	Datenbankmodell	52
8.2.1	ERD	52
8.2.2	Entitäten	52
8.2.3	Beziehungen	54

1 Kurzbeschreibung

Das Ziel von Archivist ist, ein einfach zu verwendendes digitales Notenarchiv für Musikvereine bereitzustellen. Dazu kann der Archivar eine Datenbank von Musikstücken verwalten und entsprechende Noten abspeichern. Dieser Ansatz ermöglicht eine einfache Übersicht über Musikstücke, eine Suchfunktion für das Auffinden von Stücken sowie das einfache Ausdrucken von Noten.

Nachdem viele Musikvereine keine digitale Absicherung ihrer Musikstücke besitzen, ist die Verwaltung eines Notenarchivs mit viel Aufwand verbunden, vor allem, wenn das Archiv aus einer Vielzahl von Musikstücken besteht.

Aus diesem Grund bietet Archivist ein digitales Archiv für Musikstücken an, um die Organisation dieser einfach zu halten. Durch die Verwendung von Archivist kann nicht nur eine Liste von Musikstücken mit Archivnummern geführt werden, sondern auch das Speichern und Kategorisieren in von Noten selbst in digitaler Form ist möglich.

Eine simple aber sehr hilfreiche Suchfunktion ermöglicht es dem Benutzer, ein bestimmtes Musikstück oder mehrere Musikstücke bezüglich Musiktitel, Komponist, Genre oder Archivnummer sehr effizient aufzufinden. Einige vordefinierte Filter für das Sortieren von Einträgen anhand bestimmter Kriterien unterstützen die Suchfunktion, um noch schneller Suchergebnisse zu erhalten.

Weiters können jederzeit bereits existente Daten bearbeitet sowie neue Einträge für Musikstücke erstellt werden. Die Möglichkeit, Einzelstimmen für die Musiknoten zu speichern, erleichtert die Organisation der Noten für die einzelnen Musiker erheblich. Musiknoten werden dabei als PDF-Datei für jede Einzelstimme abgespeichert. Wird beispielsweise eine zweite Trompetenstimme für ein bestimmtes Stück benötigt, so kann die dazugehörige PDF-Datei des Musikstücks rasch ausgedruckt bzw. auch auf einem Tablet angezeigt werden.

Zusammenfassend bietet das moderne Konzept von Archivist für Musikvereine ein nützliches Werkzeug, um deren Musikstücke in einer einfachen aber effizienten Art zu digitalisieren, was eine Vielzahl an Vorteilen bietet, um die ansonsten aufwändige Organisation eines Notenarchivs unkompliziert und für jedermann verwendbar zu gestalten.

2 Benutzeranleitung

2.1 Installationsanleitung

2.1.1 Installation

Die notwendigen Daten befinden sich auf dem gelieferten USB-Stick. Zur Installation folgen Sie den Schritten in der Installationsanleitung:

- Stellen Sie unter <https://java.com/de/download/> sicher das eine aktuelle JAVA Version auf ihrem PC installiert ist
- Wenn nicht, installieren Sie JAVA auf ihrem PC
- Sie können jetzt den Archivist Ordner auf ihrem USB-Stick in ein von ihnen gewünschtes Verzeichnis kopieren
- Öffnen Sie nun den Archivist Ordner
- Klicken Sie auf Archivist.jar
- Nun wird eine Desktopverknüpfung erstellt, Archivist wird installiert und gestartet
- Archivist ist nun in vollem Umfang nutzbar
- Für weitere Nutzung reicht es auf die Desktopverknüpfung zu klicken

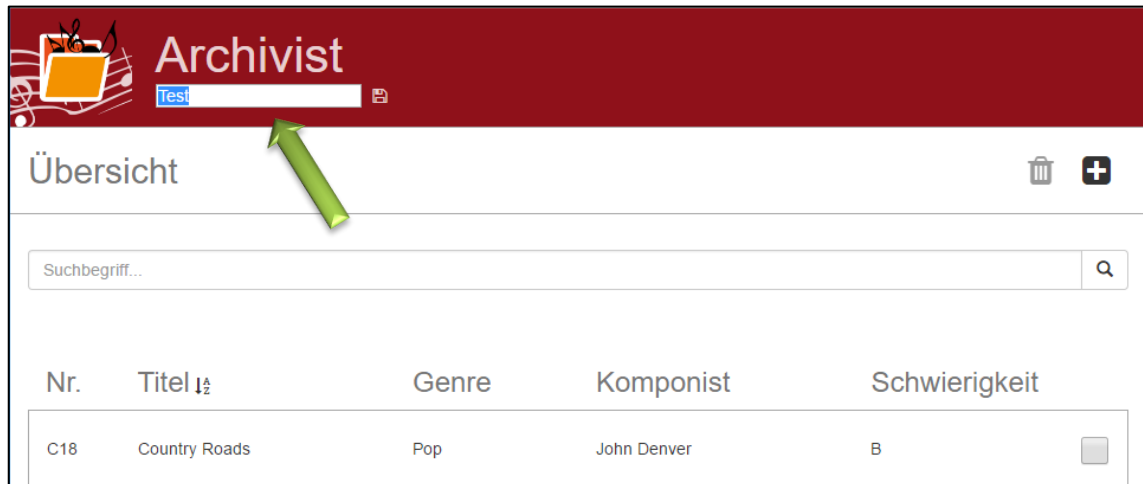
2.1.2 Deinstallation

- Für eine Deinstallation müssen Sie ihren PC vorab neu starten
- Haben Sie ihren PC neugestartet, können Sie nun den kompletten Archivist Ordner und die Desktopverknüpfung löschen
- Sie haben Archivist erfolgreich deinstalliert

2.2 Tutorial

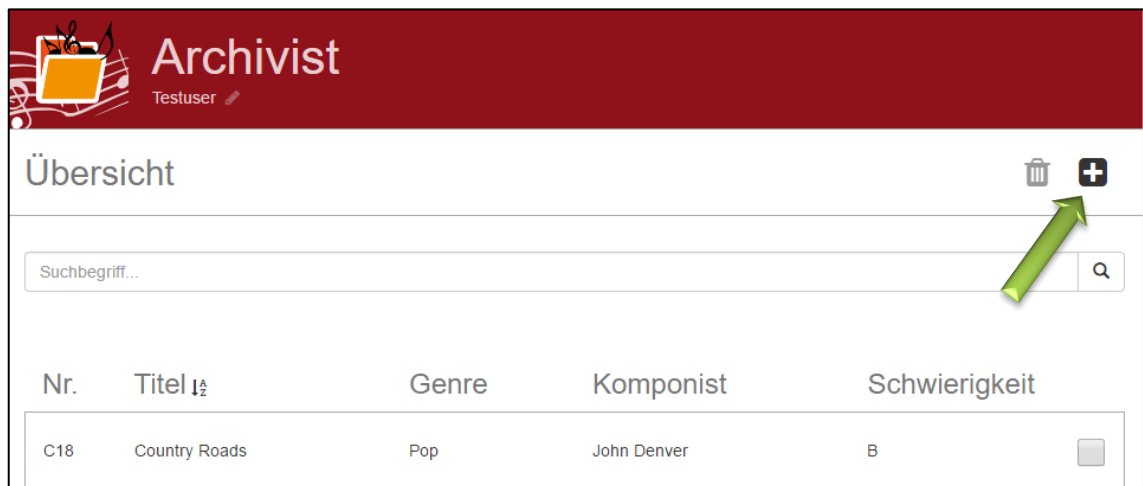
2.2.1 Eingabe des Benutzernamens

Zum eingeben des Benutzernamens klicken Sie entweder direkt in den Text oder auf „Bearbeiten“ rechts davon und bestätigen dies mit „Enter“.

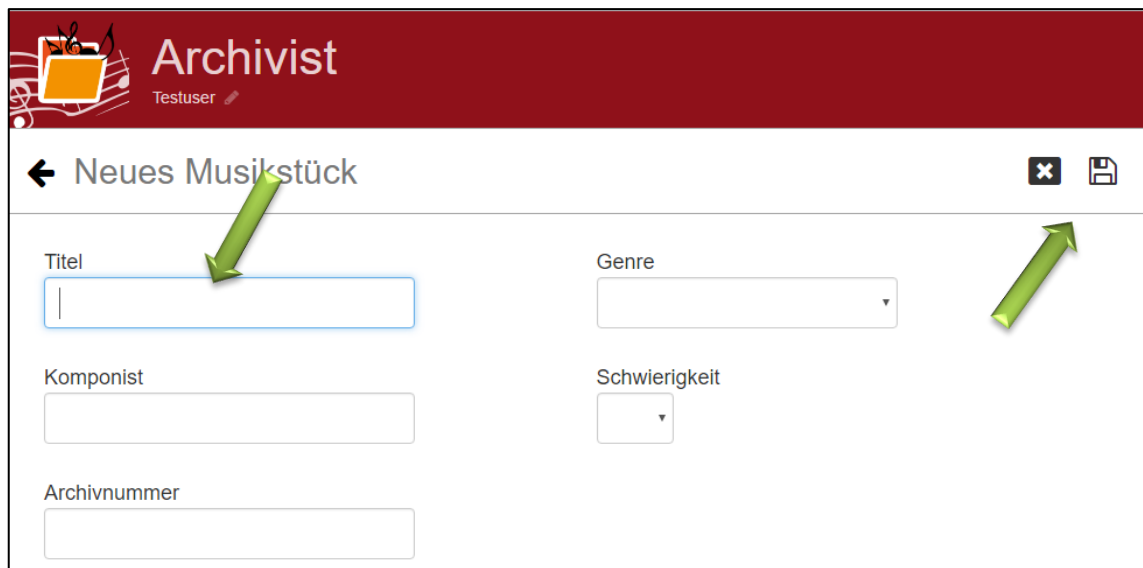


2.2.2 Anlegen eines neuen Musikstücks/Noten

Schritt 1: Zum Anlegen eines neuen Musikstücks drücken Sie „Plus“ auf der rechten Seite. Sie werden dann zu einem Eingabeformular weitergeleitet.



Schritt 2: Geben Sie nun die gewünschten Parameter ihres neuen Musikstücks ein und klicken Sie anschließend entweder auf „Speichern“ oder auf „Abbrechen“ („Abbrechen“ leitet Sie umgehend zurück zum Hauptmenü **OHNE** etwas zu speichern) in der rechten oberen Ecke.



Schritt 3: Wenn Sie zuvor auf „Speichern“ geklickt haben werden Sie nun auf die unten angeführte Seite weitergeleitet. Diese Ansicht verrät Ihnen, dass noch keine Noten in Ihrem Musikstück gespeichert wurden. Sie können nun durch einen Klick auf das „Plus“ Symbol in der Mitte der Ansicht sowie im rechten Eck neue Noten anlegen.



Schritt 4: Haben Sie zuvor wiederum „Speichern“ geklickt können Sie in dieser Ansicht eine neue Stimme für Ihr Musikstück definieren. Geben Sie die gewünschten Parameter ein und ziehen Sie die von Ihnen gewählte PDF-Datei in den strichlierten Bereich. Anschließend klicken Sie wieder auf „Speichern“ oder „Abbrechen“.

Tipp: Durch einen direkten Klick in den strichlierten Bereich öffnet sich der Windows Explorer und Sie können nach ihrem gewünschten Noten-PDF suchen.



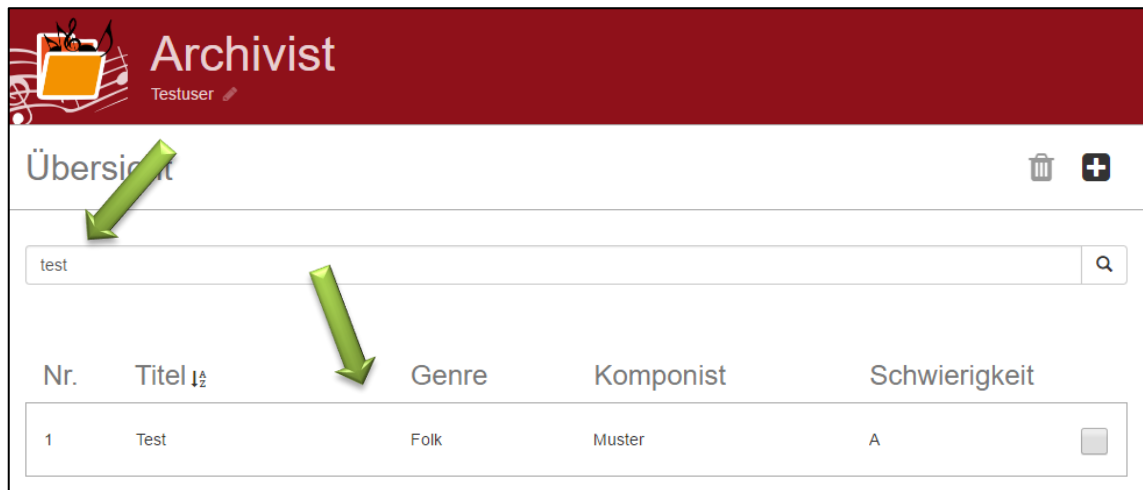
Schritt 5: Sie haben erfolgreich ein neues Musikstück sowie eine dazugehörige Stimme angelegt.

Gehen Sie nun zurück in die Hauptansicht. Klicken Sie dazu „Zurück“ in der linken Ecke der Ansicht.



2.2.3 Suchen und Bearbeiten eines Musikstücks

Schritt 1: Sie klicken in das Suchen-Textfeld im Hauptmenü und geben ihr gesuchtes Musikstück ein. (Sie können hier natürlich auch nach allen anderen Parametern suchen -> z.B. Genre) Das gesuchte Musikstück wird nun unten angezeigt. Durch einen Klick auf das angezeigte Musikstück wechseln Sie in die Musikstück-Ansicht. (Siehe Schritt 2)



Schritt 2: Sie können nun ihr Musikstück bearbeiten oder Löschen und neue Noten hinzufügen oder bestehende Noten bearbeiten.

Zum Bearbeiten des Musikstücks klicken Sie auf das Stift Symbol in der rechten Ecke. (Sie können dann die Parameter ihres Musikstücks ändern -> Siehe „Anlegen eines neuen Musikstücks/Noten“ -> Schritt 2)

Zum Bearbeiten ihrer Noten klicken Sie auf Bearbeiten über den angezeigten Noten. (Siehe „Löschen von Musikstücken und Noten“)

Zum Hinzufügen neuer Noten klicken Sie auf das Plus-Symbol neben den bestehenden Noten. Sie wechseln dann in die ihnen bereits bekannte Ansicht zum Noten hinzufügen. (Siehe „Anlegen eines neuen Musikstücks/Noten“ -> Schritt 4)

Tipp: Mit einem Klick auf Abbrechen gelangen Sie immer in die vorherige Ansicht.

The screenshot shows the 'Archivist' web application interface. At the top, there is a red header bar with the 'Archivist' logo and the text 'Testuser'. Below the header, there is a navigation bar with a back arrow and the word 'Test'. In the top right corner of the main content area, there are icons for deleting (trash) and editing (pencil), with a green arrow pointing to the edit icon. The main content area displays the following information:

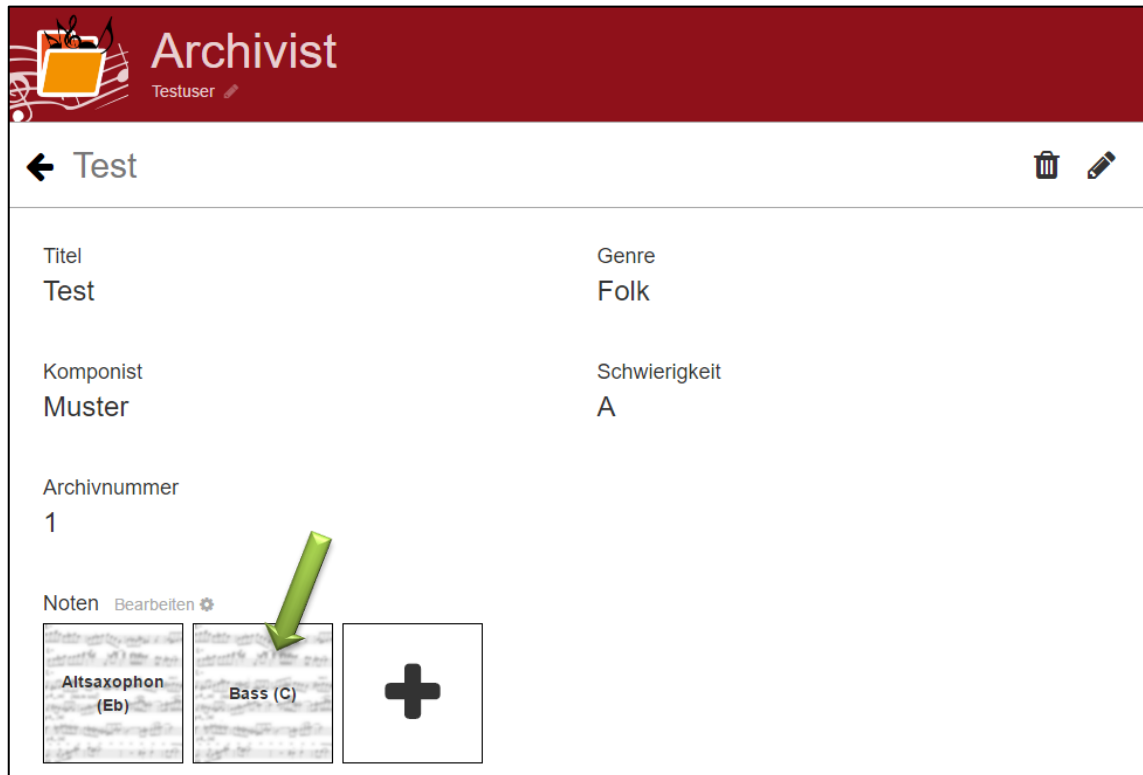
Titel Test	Genre Folk
Komponist Muster	Schwierigkeit A
Archivnummer 1	

Below this information, there is a section titled 'Noten' with a 'Bearbeiten' link and a gear icon. Under 'Noten', there are two existing notes: 'Altsaxophon (Eb)' and 'Bass (C)'. To the right of these notes is a large plus sign (+) in a square box, with a green arrow pointing to it. The interface is clean and modern, with a white background and red accents.

2.2.4 Herunterladen von Noten

Suchen Sie ihr gewünschtes Musikstück in der Hauptansicht und klicken Sie darauf.

Für das Herunterladen der gewünschten Noten genügt ein Klick darauf. Die Noten werden nun in ihren "Downloads"-Ordner gespeichert.



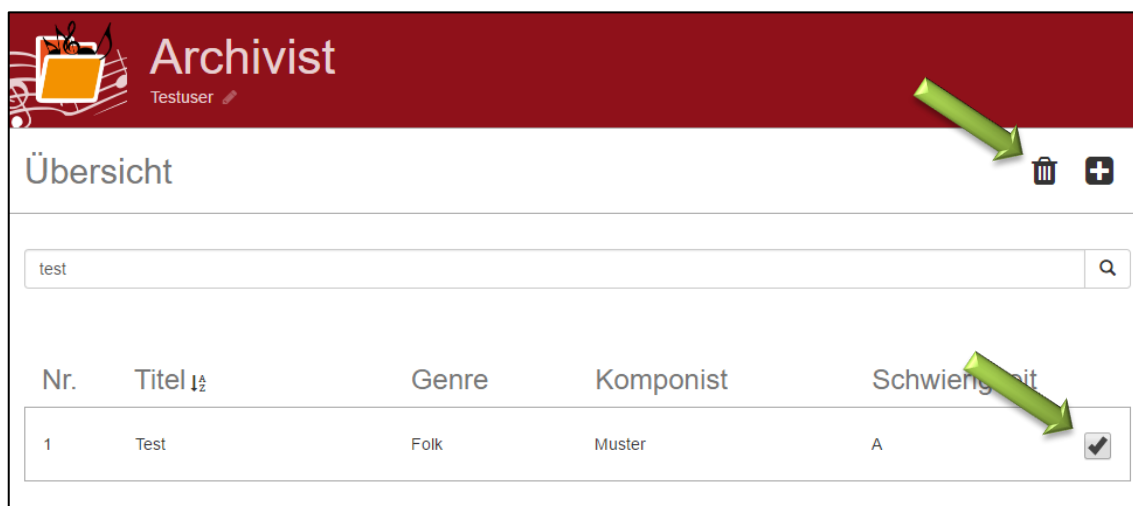
2.2.5 Löschen von Musikstücken und Noten

Wenn Sie zuvor auf Noten Bearbeiten geklickt haben wechseln Sie in die nachfolgende Ansicht. Zum Löschen markieren Sie die Noten am rechten Rand und klicken dann auf das Löschen-Symbol im rechten Eck. („Plus“ zum Hinzufügen)



Wollen Sie nun ein komplettes Musikstück (**Vorsicht:** Alle an diesem Musikstück angefügten Noten gehen hierbei verloren) löschen, haben Sie dazu zwei Möglichkeiten.

Möglichkeit 1: Suchen Sie das zu löschende Musikstück über das „Suchen“-Feld. Danach markieren Sie das Musikstück am rechten Rand und klicken anschließend auf das „Löschen“-Symbol in der rechten Ecke.

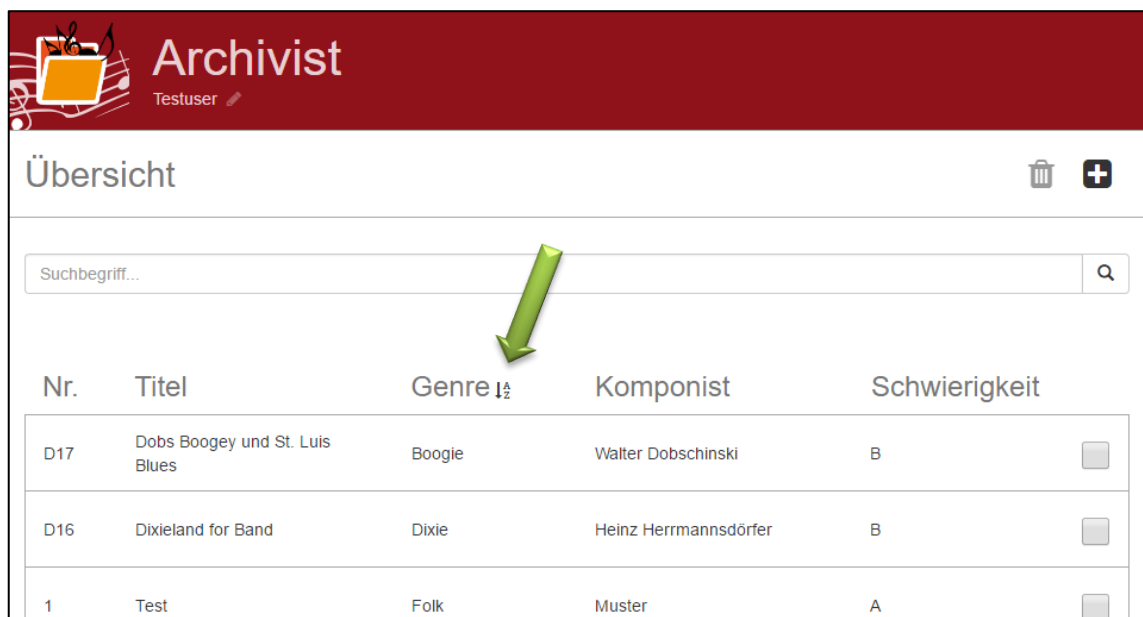


Möglichkeit 2: Sie befinden sich bereits in dem ausgewählten Musikstück. Zum Löschen klicken Sie auf das Löschen-Symbol in der rechten Ecke. (**Vorsicht:** Alle an diesem Musikstück angefügten Noten gehen hierbei verloren)



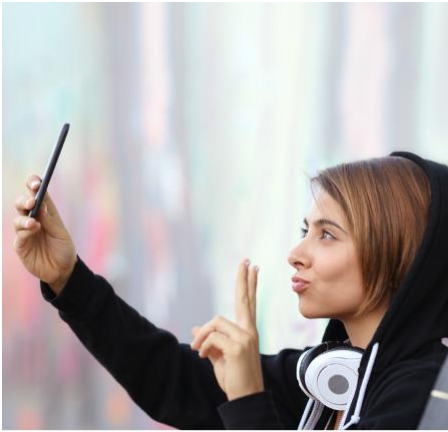
2.2.6 Sortierte Ansicht

Sie können ihre vorhandenen Musikstücke nach allen eingegebenen Parametern auf- oder absteigend sortieren. Gehen Sie dazu auf die Hauptansicht und klicken Sie auf den gewünschten Parameter. Nun erscheint neben dem Parameter eine Anzeige für das aufsteigende Sortieren. Ein weiterer Klick sortiert den Parameter absteigend. Standardmäßig ist es aufsteigend nach Musiktitel sortiert.



The screenshot shows the 'Archivist' application interface. At the top, there is a red header bar with the 'Archivist' logo and the text 'Testuser'. Below the header, the main area is titled 'Übersicht'. There is a search bar with the placeholder text 'Suchbegriff...' and a magnifying glass icon. Below the search bar is a table with the following columns: 'Nr.', 'Titel', 'Genre', 'Komponist', and 'Schwierigkeit'. A green arrow points to the 'Genre' column header, which has a small '1/2' icon next to it, indicating ascending sort order. The table contains three rows of data:

Nr.	Titel	Genre	Komponist	Schwierigkeit
D17	Dobs Boogey und St. Luis Blues	Boogie	Walter Dobschinski	B
D16	Dixieland for Band	Dixie	Heinz Herrmannsdörfer	B
1	Test	Folk	Muster	A



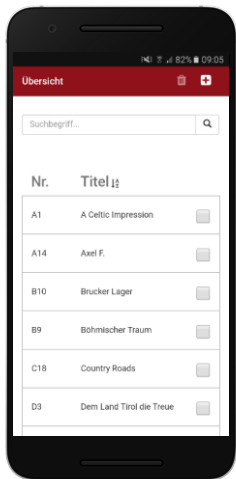
Mobile Computing



Archivist

Speichern, Verwalten, Suchen

Mit Archivist steht Ihnen eine Möglichkeit zur Verwaltung, Speicherung und Suche von Noten-PDFs zur Verfügung. Musikkapellen diverser Größe haben damit einen erheblichen Vorteil in der Verwaltung und Digitalisierung ihrer Notenarchive. Archivist wird derzeit als lokaler Webservice angeboten und kann problemlos für Desktop-Geräte und Smartphones abgerufen werden.



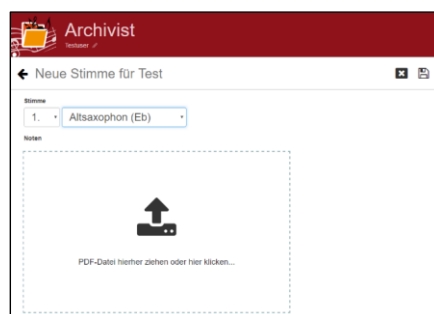
Eckdaten

Musikstücke mit verschiedensten Attribute anlegen. (Titel, Genre, Schwierigkeit, ...)

Zu jedem Musikstück können beliebig viele Noten-PDFs gespeichert werden. Diese gliedern sich nach Stimme und Instrument.

Speichern der PDFs mittels „drag & drop“. Anzeige der Dateien mit jedem Gerät mit PDF-Reader möglich.

Einfaches Verwalten der Musikstücke sowie umfangreiche Such- und Gliederungsoptionen.



4 Pflichtenheft

4.1 Einleitung

4.1.1 Zweck des Dokuments

4.1.2 Gültigkeit des Dokuments

Dieses Pflichtenheft ist für das gesamte Projekt Archivist (Version 1) gültig (eventuell wird das Projekt in späteren Semester erweitert, bzw. fortgeführt). Es setzt auf kein bestehendes Pflichtenheft auf. Für Änderungen ist das Projektteam (Paul Schmutz, Boris Fuchs) zuständig.

4.1.3 Begriffsbestimmungen und Abkürzungen

App	Softwareanwendung; meist gebräuchlich für Anwendungen auf mobilen Endgeräten
ASCII	American Standard Code for Information Interchange; 7-Bit Zeichencodierung, Grundlage für spätere Codierungen mit mehr Bits
Backend	Bezeichnet eine Server-Anwendung, welche Daten verwaltet (Gegenstück: Frontend)
Base64	Verfahren zur Codierung von Binärdateien, Base64-codierte Dateien bestehen aus einer lesbaren Folge von ASCII-Zeichen
Browser	Anwendung zur Darstellung von Webinhalten (Websites)
Client	Endgerät oder Anwendung eines Netzwerks, welches mit einem Server kommuniziert
Datenbank	Logische Einheit zusammengehöriger Daten; wird von einem DBMS verwaltet
DBMS	Datenbankmanagementsystem; stellt Werkzeuge bereit, mehrere Datenbanken zu erstellen und zu verwalten sowie strukturiert auf deren Daten zuzugreifen
Frontend	Bezeichnet eine Client-Anwendung, welche Daten vom Backend empfängt und darstellt (Gegenstück: Backend)
HTML	Hypertext Markup Language; Beschreibungssprache zur

	Strukturierung von Elementen wie Text, Bildern usw., Grundlage für das World Wide Web, HTML-Dokumente werden in einem Browser grafisch dargestellt
HTTP	Hypertext Transfer Protocol; Protokoll zur Übertragung von Anwendungsdaten über ein Netzwerk
Interface	Schnittstelle; User-Interface=für den Benutzer sichtbare grafische Oberfläche einer Anwendung
JPA	Java Persistence API; Schnittstelle für Java-Anwendungen zur Übertragung von Datenbankeinträgen; dient zum ORM
JSON	JavaScript Object Notation; Datenformat als (einfach lesbaren) Text zum Datenaustausch zwischen zwei Anwendungen
ORM	Object Relational Mapping; Konzept zur Abbildung von Objekten objektorientierter Programmiersprachen in relationale Datenbanken
PDF	Portable Document Format; plattformunabhängiges Dateiformat, von Adobe entwickelt
Server	Computer bzw. Anwendung eines Netzwerks, welche Dienste und Ressourcen für Clients bereitstellt und anbietet

4.1.4 Zusammenhang mit anderen Dokumenten

Derzeit liegt weder ein Angebot noch ein Lastenheft vor.

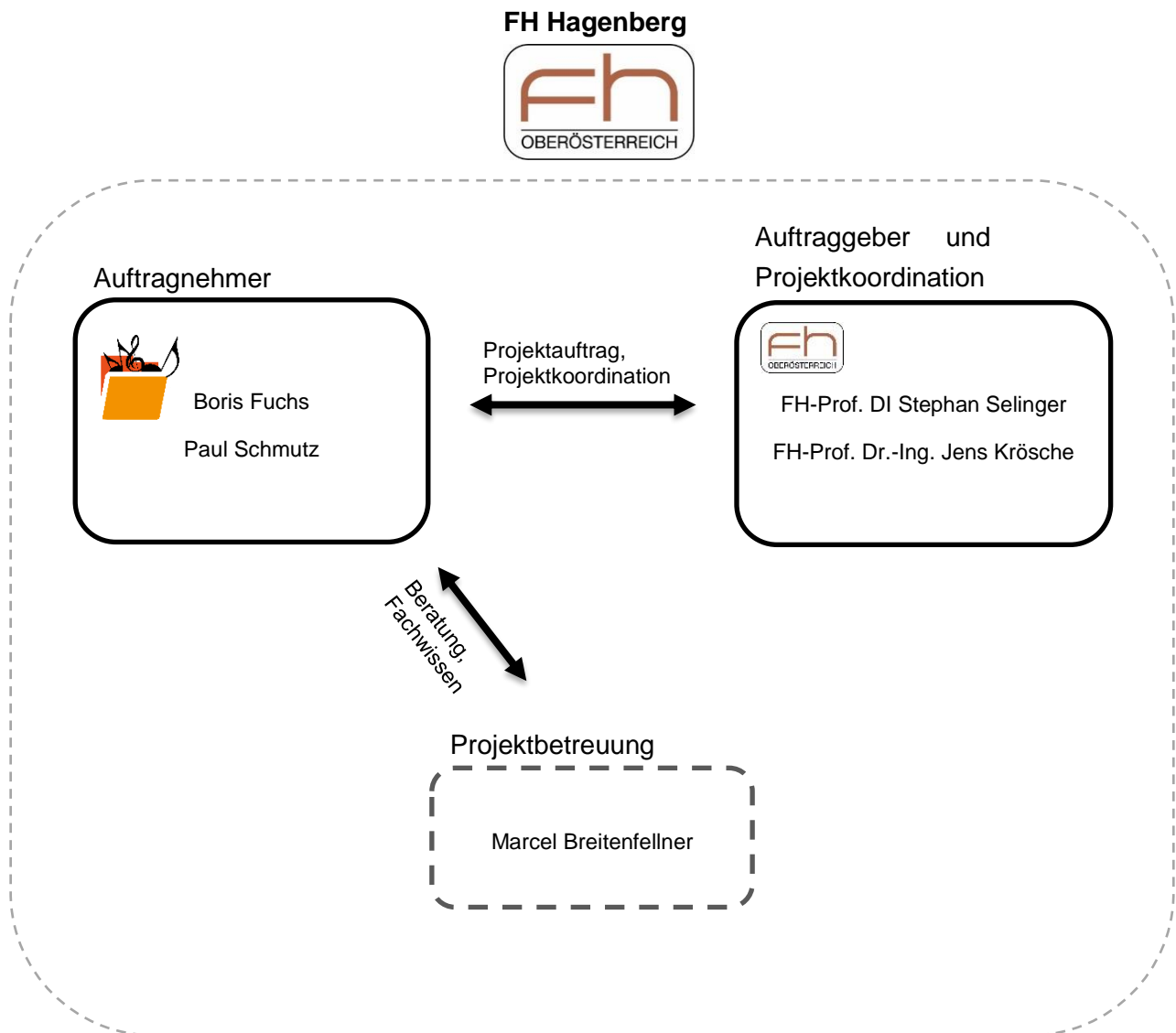
4.1.5 Überblick über das Dokument

- Allgemeine Beschreibung des Produkts
- Detaillierte Beschreibung der geforderten Produktmerkmale
- Vorgaben an die Projektabwicklung (Hardware, Software, Betriebssystem)
- Verpflichtungen des Auftraggebers

4.2 Allgemeine Beschreibung des Produkts

4.2.1 Projektumgebung

4.2.1.1 Übersicht



4.2.1.2 Kontaktdaten

FH-Prof. DI Stephan Selinger	
Telefon:	+43 5 0804 22822
E-Mail:	stephan.selinger@fh-hagenberg.at
FH-Prof. Dr.-Ing. Jens Krösche	
Telefon:	+43 5 0804 22821
E-Mail:	jens.kroesche@fh-hagenberg.at
Marcel Breitenfellner	
E-Mail:	S1610455008@students.fh-hagenberg.at
Boris Fuchs	
Telefon:	+43 676 884 003 638
E-Mail:	boris.fuchs@students.fh-hagenberg.at
Paul Schmutz	
Telefon:	+43 680 303 06 76
E-Mail:	paul.schmutz@students.fh-hagenberg.at

4.2.2 Zweck des Produkts

Zweck des Produktes/Projektes ist es ein digitales Notenarchiv für Musikvereine zur Verfügung zu stellen. Da im Moment in einem Großteil der Musikvereine das Notenarchiv nicht digital gehalten wird, soll hier unser Produkt erhebliche Vorteile bieten, wie zum Beispiel:

- Schnelleren Zugriff auf Daten (Musikstücke)
- Keine räumlichen Abhängigkeiten
- Bessere Gliederung und Strukturierung möglich
- Permanenter Datenzugriff möglich
- Schnelle Bereitstellung von ausgedrucktem Notenmaterial

4.2.3 Überblick über die geforderte Funktionalität

Archivist ist eine digitale Datenbank zur Speicherung von digitalisierten Notenblättern, sowie deren Sortierung in verschiedenste Kategorien. Dadurch kann ein Notenarchiv effizient durchsucht und Noten per PDF-Datei oder Ausdruck bereitgestellt werden.

4.2.4 Allgemeine Einschränkungen

Nachdem die Idee fürs Notenarchiv vom Projektteam selbst stammt und nicht vom Projektkoordination der FH Hagenberg vorgegeben wurde, unterliegt das Projekt keinen ausdrücklichen Einschränkungen und bietet damit Spielraum für die kreative Umsetzung vom Projektteam.

4.2.5 Vorgaben zu Hardware und Software

Grundbaustein für das System ist eine browserabhängige Anwendung, die vorrangig von einem PC/Notebook aus benutzt wird. Fokussiert wird die Nutzung über alle Betriebssysteme, wobei keine Internetanbindung erforderlich ist. Eine Erweiterung in Form einer App für Android ist angedacht, jedoch wird diese jedenfalls im jetzigen Semesterprojekt noch nicht umgesetzt. Es kommen bei der Realisierung speziell Angular JS, MySQL, usw. (siehe 4.4.1.2.2 Entwicklungsumgebung, -software und -bibliotheken) zur Anwendung.

4.2.6 Benutzer des Produkts

Das System ist für die tägliche Nutzung ausgelegt. Der Notenarchivar des Musikvereins sollte vorzugsweise der Systemadministrator sein. Vor allem aber wird das System leicht und verständlich zu bedienen sein und es jedem Nutzer ermöglichen sich in kürzester Zeit perfekt durch das User-Interface bewegen zu können.

4.3 Detaillierte Beschreibung der geforderten Produktmerkmale

Durch die eigenständige Findung der Projektidee werden die Produktmerkmale vom Projektteam selbst definiert.

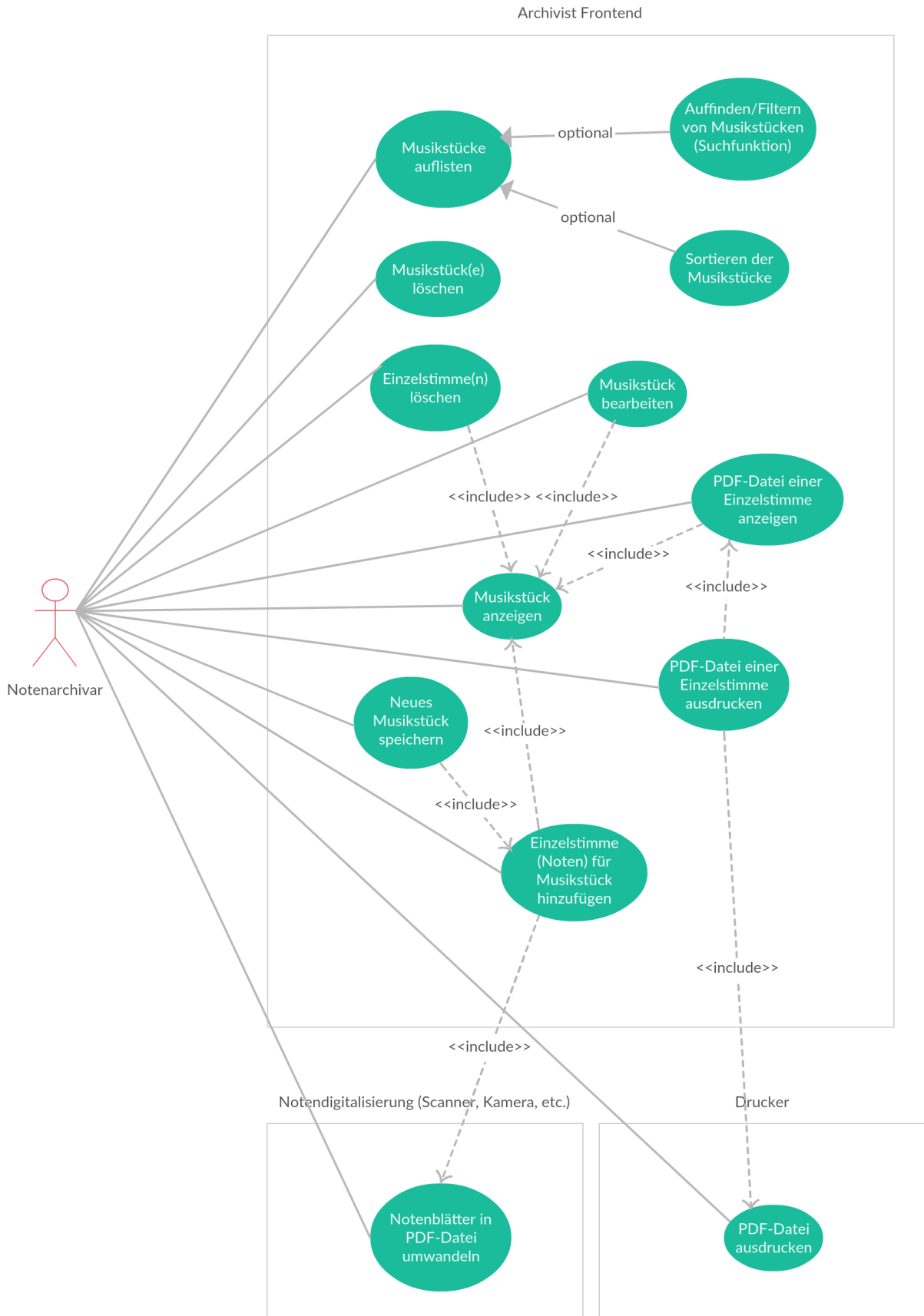
4.3.1 Lieferumfang

Das fertige auslieferbare Endprodukt enthält zwei wesentliche Bestandteile:

- Softwarepaket mit allen abhängigen Komponenten (Server, Datenbank)
- Handbuch (Installation, Bedienung)

4.3.2 Abläufe (Szenarien) von Interaktionen mit der Umgebung

Dem Benutzer des Systems stehen bestimmte Funktionalitäten zur Verfügung, welche für folgende Szenarien geeignet sind. Näheres zu „Archivist Frontend“ ist dem Punkt 4.3.3.1 Benutzerschnittstellen (User Interfaces) zu entnehmen.

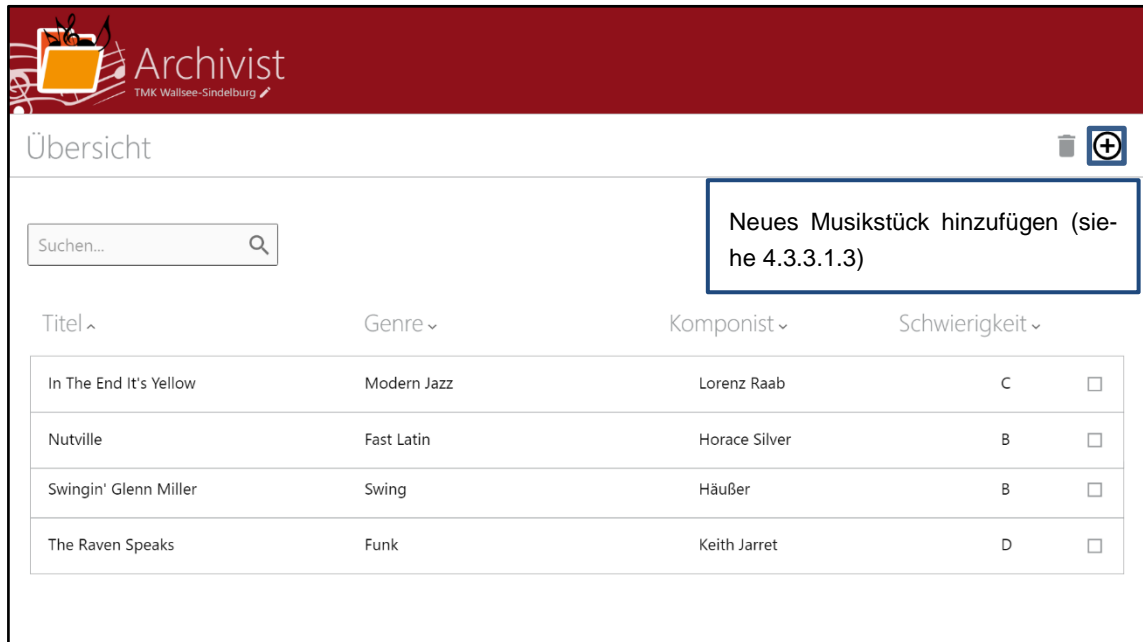


4.3.3 Externe Schnittstellen des Produkts

4.3.3.1 Benutzerschnittstellen (User Interfaces)

4.3.3.1.1 Hauptmenü

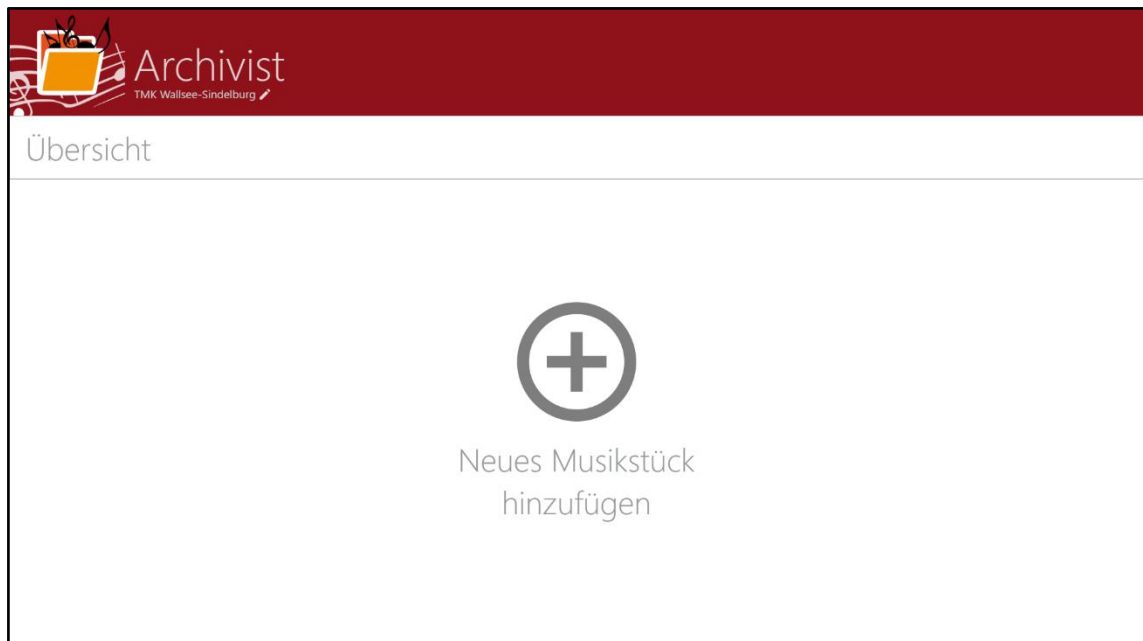
Standard-Ansicht des Hauptmenüs:



The screenshot shows the 'Archivist' application interface. At the top is a red header with the logo and 'TMK Wallsee-Sindelburg'. Below the header is a white bar with the title 'Übersicht' and a trash icon. A search bar with the placeholder 'Suchen...' and a magnifying glass icon is on the left. A blue-bordered box on the right contains the text 'Neues Musikstück hinzufügen (siehe 4.3.3.1.3)'. Below these elements is a table with columns: 'Titel', 'Genre', 'Komponist', and 'Schwierigkeit'. The table contains four rows of music pieces.

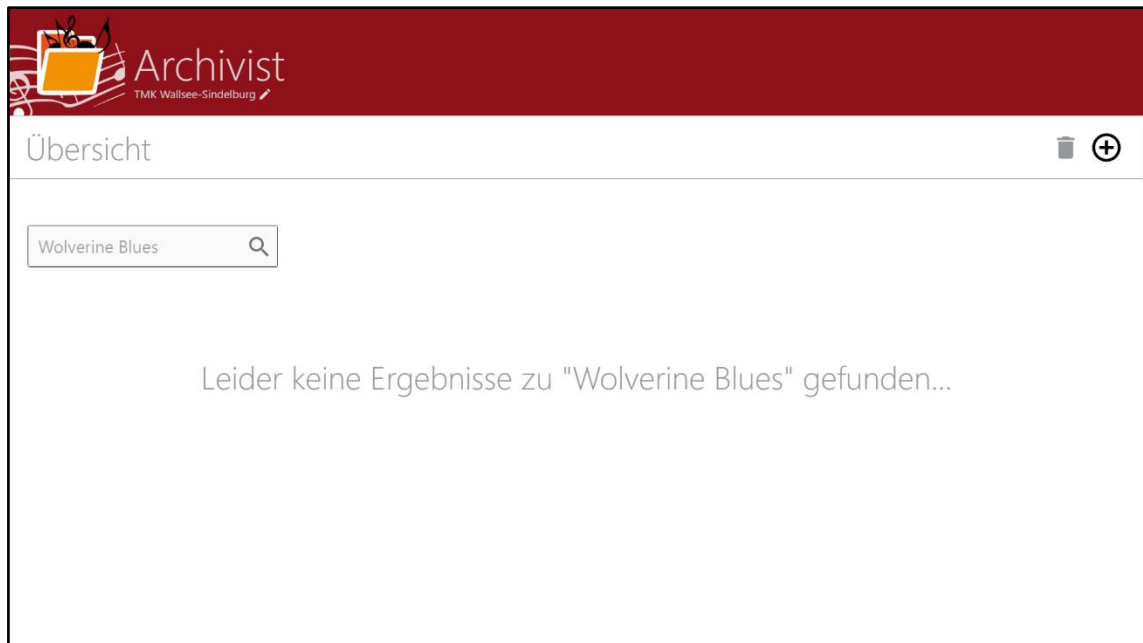
Titel	Genre	Komponist	Schwierigkeit
In The End It's Yellow	Modern Jazz	Lorenz Raab	C
Nutville	Fast Latin	Horace Silver	B
Swingin' Glenn Miller	Swing	Häußner	B
The Raven Speaks	Funk	Keith Jarrett	D

Hauptmenü, beim ersten Aufruf nach der Installation (noch keine Daten vorhanden):

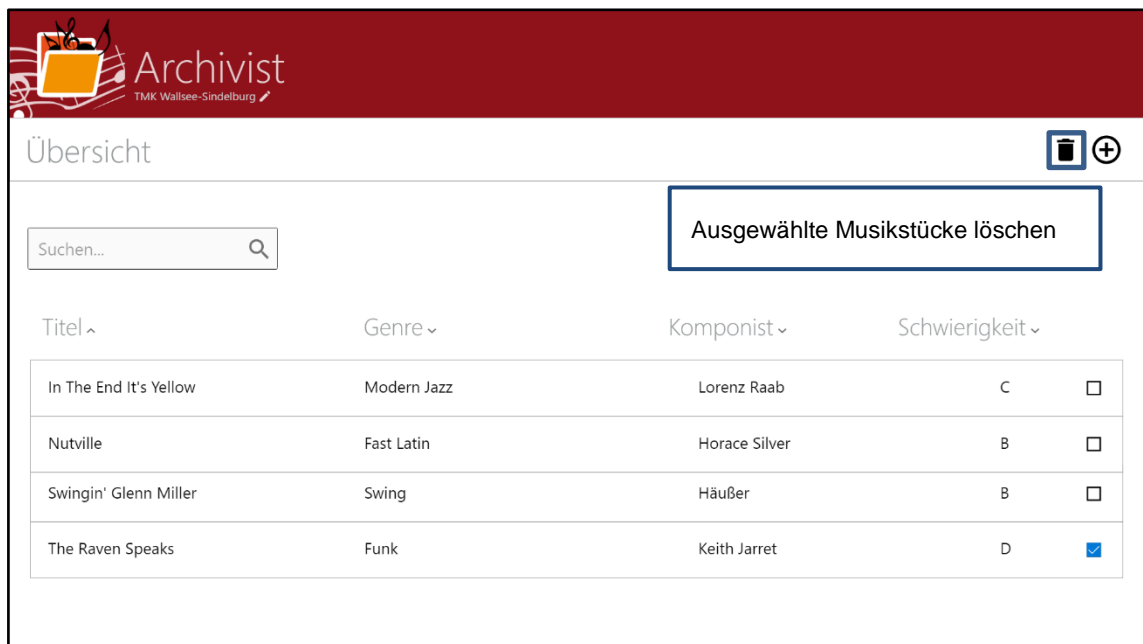


The screenshot shows the 'Archivist' application interface after installation. The header and search bar are the same as in the previous screenshot. However, the table is empty. In the center of the main content area, there is a large grey plus icon inside a circle, with the text 'Neues Musikstück hinzufügen' below it.

Suche nach Musikstücken ohne Suchergebnis:



Anhaken von mehreren Musikstücken, anschließend ist löschen möglich:



4.3.3.1.2 Musikstück Detailansicht

The screenshot shows the 'Archivist' app interface for a music piece. The header is red with the 'Archivist' logo and 'TMK Wallsee-Sindelburg'. The title 'In the End It's Yellow' is displayed. Below the title, the genre is 'Modern Jazz' and the composer is 'Lorenz Raab'. The difficulty is set to 'C'. The archive number is 'I3'. A section titled 'Noten >' shows a table of staves for different instruments: Partitur, Alt Sax, Flöte, Trompete, and Bariton Sax. A plus sign is next to the table. On the right, there are two buttons: 'Musikstück bearbeiten (siehe ...)' and 'Musikstück löschen'.

Partitur	Alt Sax	Flöte	Trompete	Bariton Sax
			1	2

4.3.3.1.3 Neues Musikstück

The screenshot shows the 'Archivist' app interface for adding a new music piece. The header is red with the 'Archivist' logo and 'TMK Wallsee-Sindelburg'. The title 'Neues Musikstück hinzufügen' is displayed. Below the title, there are input fields for 'Titel', 'Genre', and 'Komponist'. The difficulty is set to 'A'. The archive number is empty. A section titled 'Noten >' shows a plus sign button. On the right, there are two buttons: 'Einzelstimmen (siehe 4.3.3.1.5)' and 'Einzelstimmen hinzufügen (siehe 4.3.3.1.6)'. At the bottom, there are two buttons: 'Speichern' and 'Abbrechen'.

4.3.3.1.4 Musikstück bearbeiten

Archivist
TMK Wallsee-Sindelburg

← Musikstück bearbeiten

Titel
In the End It's Yellow

Genre
Modern Jazz

Komponist
Lorenz Raab

Schwierigkeit
C

Archivnummer
I3

Noten >

Partitur	Alt Sax	Flöte	Trompete	Bariton
			1	Sax

+

Einzelstimmen (siehe 4.3.3.1.5)

Einzelstimmen hinzufügen (siehe 4.3.3.1.6)

Speichern Abbrechen

4.3.3.1.5 Einzelstimmen Musikstück

Standard-Ansicht der Einzelstimmen:

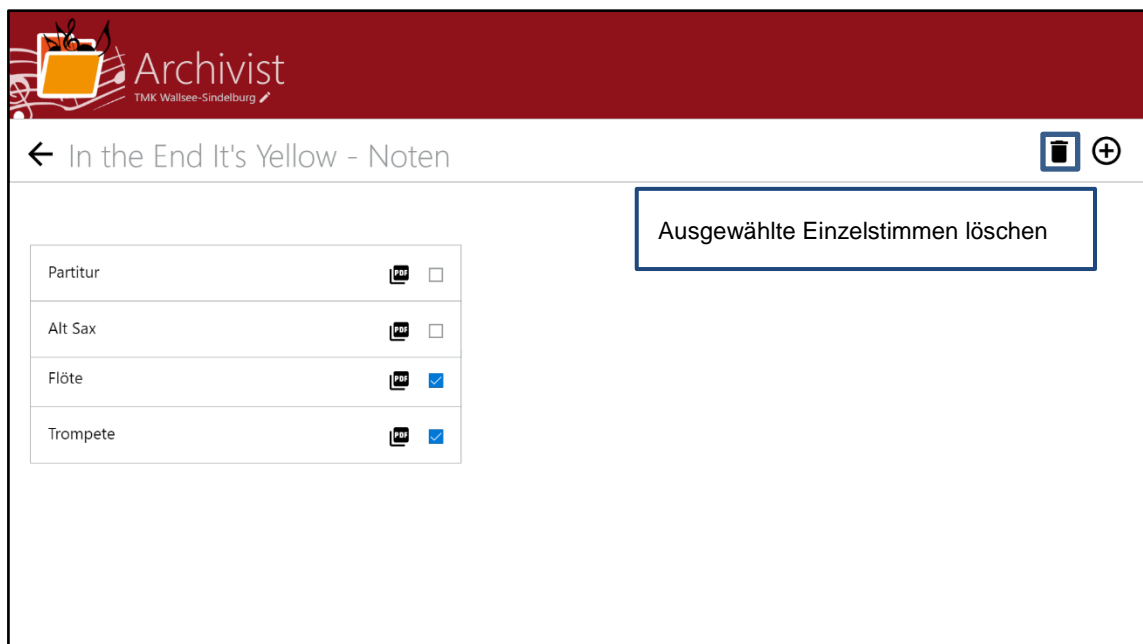
Archivist
TMK Wallsee-Sindelburg

← In the End It's Yellow - Noten

Einzelstimme hinzufügen

Partitur	PDF	<input type="checkbox"/>
Alt Sax	PDF	<input type="checkbox"/>
Flöte	PDF	<input type="checkbox"/>
Trompete	PDF	<input type="checkbox"/>

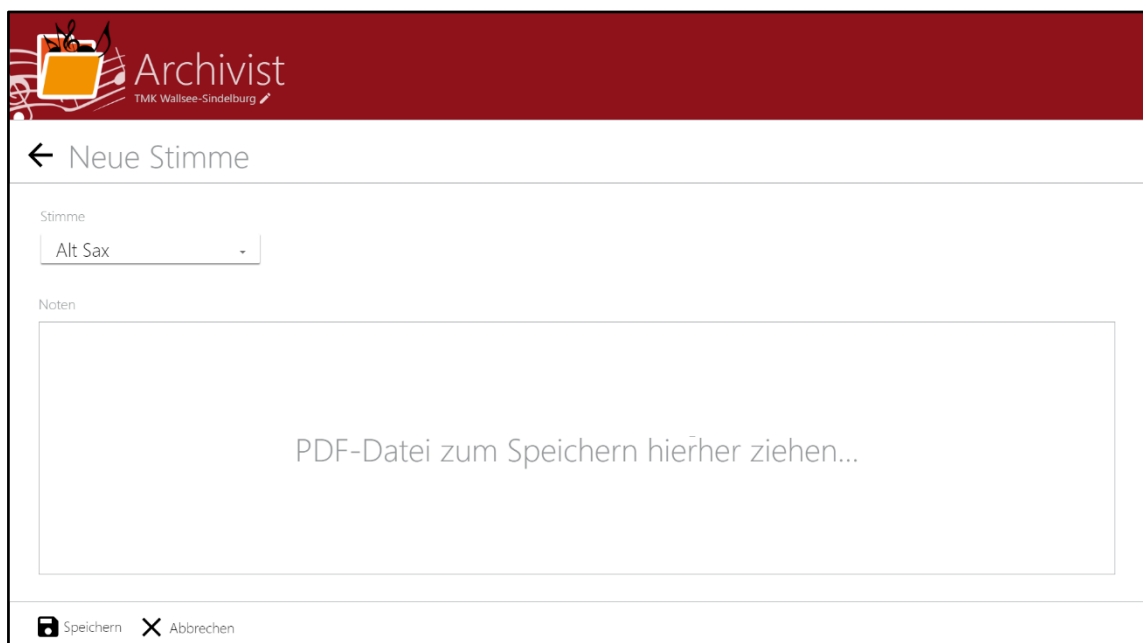
Anhaken von Einzelstimmen, anschließend löschen möglich:



The screenshot shows the Archivist web application interface. The header is dark red with the Archivist logo and 'TMK Wallsee-Sindelburg'. The main content area is white. At the top, there's a navigation bar with a back arrow and the title 'In the End It's Yellow - Noten'. To the right of the title is a trash icon and a plus sign. Below the navigation bar is a table with four rows: 'Partitur', 'Alt Sax', 'Flöte', and 'Trompete'. Each row has a PDF icon and a checkbox. The checkboxes for 'Flöte' and 'Trompete' are checked. A blue-bordered callout box with the text 'Ausgewählte Einzelstimmen löschen' points to the trash icon.

Instrument	PDF	Checkbox
Partitur	PDF	<input type="checkbox"/>
Alt Sax	PDF	<input type="checkbox"/>
Flöte	PDF	<input checked="" type="checkbox"/>
Trompete	PDF	<input checked="" type="checkbox"/>

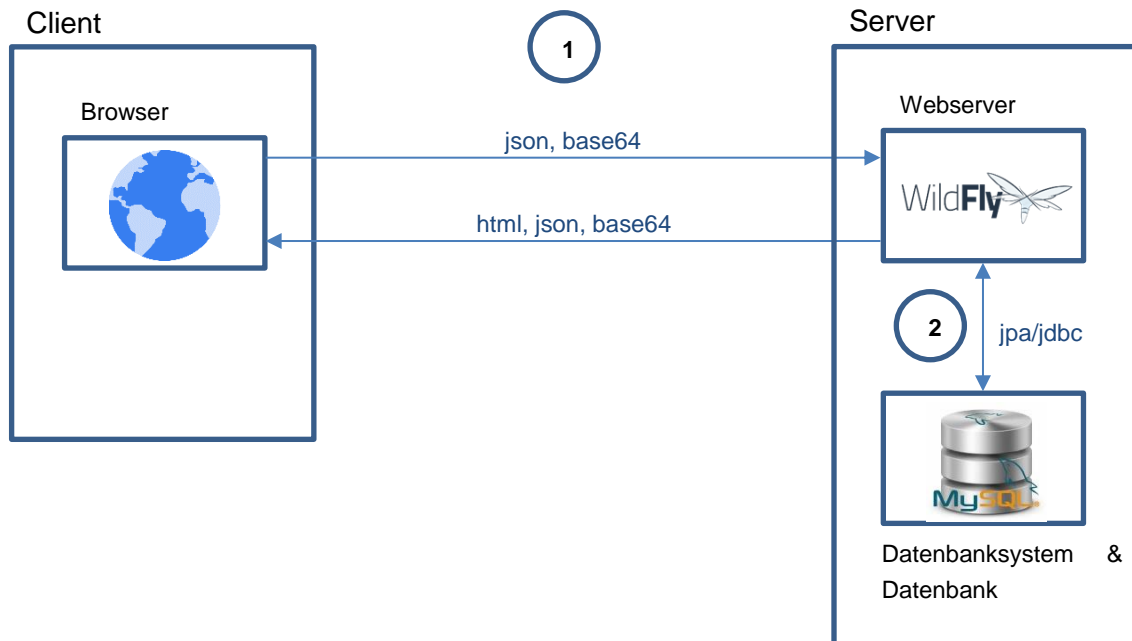
4.3.3.1.6 Neue Einzelstimme



The screenshot shows the 'Neue Stimme' page in the Archivist web application. The header is dark red with the Archivist logo and 'TMK Wallsee-Sindelburg'. The main content area is white. At the top, there's a navigation bar with a back arrow and the title 'Neue Stimme'. Below the navigation bar is a form with two sections: 'Stimme' and 'Noten'. The 'Stimme' section has a dropdown menu with 'Alt Sax' selected. The 'Noten' section is a large text area with the text 'PDF-Datei zum Speichern hierher ziehen...'. At the bottom of the page, there are two buttons: 'Speichern' (with a save icon) and 'Abbrechen' (with a close icon).

4.3.3.2 Systemschnittstellen

Die in der zu entwickelnden Anwendung enthaltenen Schnittstellen sind wie folgt strukturiert:



Anmerkung: Client und Server befinden sich auf demselben Rechner.

1 Client-Server

Protokoll: HTTP

Datenformate: HTML (für Benutzerschnittstelle für den Browser)
 JSON (für die im Browser dargestellten Anwendungsdaten)
 Base64 (für zu übertragende Dateien)

2 Webserver-Datenbank

Protokoll: JPA/JDBC

4.4 Vorgaben an die Projektabwicklung

Im Folgenden werden die Voraussetzungen zur Abwicklung des Projekts hinsichtlich technischer sowie organisatorischer Bedingungen dargestellt.

4.4.1 Anforderungen an die Realisierung

4.4.1.1 Hardware

Die benötigten Hardware-Komponenten zur Durchführung des Projekts beschränken sich auf einen standardmäßigen PC. Zum Einlesen von Notenblättern wird entweder ein Scanner oder eine Kamera (bzw. Smartphone) verwendet oder bestehendes digitales Notenmaterial verwendet.

Zusammengefasst bestehen folgende Hardware-Anforderungen:

- Standard PC
- Scanner (optional) oder
- Smartphone/Kamera (optional)

4.4.1.2 Software

4.4.1.2.1 Betriebssystem

Für das Betriebssystem des **Entwicklungsrechners** wird bevorzugt **Windows 10** verwendet. Auch andere Betriebssysteme erfüllen die Anforderungen für die Entwicklungstools, jedoch wird in diesem Dokument ausschließlich der Umgang mit Windows 10 aufgezeigt; etwaige Abweichungen für andere Systeme werden nicht behandelt.

Zusammenfassend kann gewählt werden zwischen den Betriebssystemen:

- Windows 7/10 (bevorzugt)
- Linux
- Mac

Bezüglich Betriebssystem stimmen die Anforderungen an das **Endsystem** mit jenen des Entwicklungssystems überein.


Es kann wiederum gewählt werden zwischen:

- Windows 7/10 (bevorzugt)
- Linux
- Mac

4.4.1.2.2 Entwicklungsumgebung, -software und -bibliotheken

Zur Erstellung der Software werden einige Softwaretools benötigt. Alle zur **Entwicklung** des Backends und Frontends benötigten Bestandteile sind:

<p>Spring Tool Suite (STS)</p> 	<p>STS dient hauptsächlich als Entwicklungsumgebung für das Spring Boot Backend.</p> <p>Zusätzlich kann STS auch für die Frontend-Entwicklung verwendet werden (das Frontend ist eine einfache HTML-Seite, für die jeder beliebige Text-Editor verwendet werden kann).</p> <p>Nachdem STS eine modifizierte Version von Eclipse ist, werden die Plattformen Windows, Linux und Mac unterstützt.</p>
<p>Wildfly</p> 	<p>WildFly ist ein Application Server nach dem Java-EE-Standard (Open-Source).</p> <p>Im Projekt werden das Spring Boot Backend sowie das Frontend auf einem WildFly-Server gehostet. Alternativ können auch alle Java-EE-fähigen Applikationsserver verwendet werden.</p> <p>Da WildFly mit Java implementiert wurde, kann dieser plattformunabhängig genutzt werden.</p>
<p>MySQL</p> 	<p>Die Daten des Projekts werden mithilfe des relationalen Datenbankmanagementsystems MySQL (Open-Source) verwaltet.</p>
<p>MySQL Workbench</p> 	<p>Als wichtiges Planungswerkzeug wird die MySQL Workbench verwendet. D. h. es wird damit das notwendige Datenmodell (ERD) erstellt und eine neue MySQL-Datenbank daraus erzeugt.</p>
<p>AngularJS</p> 	<p>Zur Entwicklung der Frontend-Anwendung kommt HTML + JavaScript zum Einsatz. AngularJS ist dabei ein JavaScript-Framework zur Erstellung von Single-Page-Webanwendung. Die gesamte Frontend-Programmlogik wird mithilfe AngularJS entwickelt.</p> <p>Zusätzlich zu AngularJS wird auch ein AngularJS-Tool UI-Router verwendet, das die Navigation zwischen einzelnen Seiten in der AngularJS-App erleichtert.</p>
<p>Bootstrap</p>	<p>Als CSS-Framework für die Frontend-Anwendung dient</p>

	Bootstrap. Dadurch werden einige Standard-Designs für HTML-Elemente sowie ein Grid-System zur Anordnung von Elementen (auch auf verschiedenen Displaygrößen von Endgeräten) bereitgestellt.
---	---

Die Anforderungen für **Endsysteme** bestehen aus der Installation folgender Komponenten, welche sich zum Teil mit den eben erwähnten Tools für die Entwicklung überschneiden:

- Anforderungen „lokales Backend“
 - MySQL ...zum Bereitstellen und Betreiben der Datenbank
 - WildFly ...zum Bereitstellen der http-Schnittstelle für die auf WildFly gehostete Frontend-Webapplikation
- Anforderungen Frontend
 - Browser (Unterstützung von AngularJS notwendig)
 - Beispiele:
 - Google Chrome
 - Firefox
 - Safari
 - Microsoft Edge
 - Internet Explorer 9-11

Anmerkung: Frontend und Backend werden auf ein und demselben Rechner eingerichtet!

4.4.2 Fertige und zugekaufte Komponenten

Ein Zukauf von Hardware-Komponenten ist für das Projekt nicht notwendig, da das Projektteam bereits Rechner für die Entwicklung als auch Scanner bzw. Kameras für das Einlesen von Notenblättern besitzt.

Da die benötigten Software-Komponenten kostenlos erhältlich sind, wird auch dahingehend kein Zukauf getätigt.

4.4.3 Unterauftragnehmer

Folgende Abhängigkeiten von Unterauftragnehmern bestehen:

Lieferobjekt	Unterauftragnehmer	Liefertermin	Status
Logo (in unterschiedlichen Ausführungen jeweils als Bilddatei)	Birgit Haselmayr	19.05.2017	✓

4.4.4 Lieferbedingungen

Das Projekt wird spätestens am **04.07.2017** vollständig an den Auftraggeber übergeben.

Das Projekt gilt als abgenommen, wenn sich Projektdokumentation und Source-Code zum Stichtag auf dem angegebenen SVN-Repository befinden.

SVN-Repository: <https://svn01.fh-hagenberg.at/mcm/JG16S17P01>

4.5 Verpflichtungen des Auftraggebers

Nachdem sowohl Auftragnehmer und Auftraggeber der FH Hagenberg angehören, verpflichtet sich der Auftraggeber die Auftragnehmer durch Vermittlung von Know-How und Bereitstellung von Hilfsmitteln zu unterstützen.

Für dieses Projekt besteht derzeit kein Bedarf, Hardware und Software von der FH Hagenberg zu beziehen.

Der Auftraggeber kann bei Bedarf aufgrund folgender Aspekte vom Projektteam kontaktiert werden:

- Versorgung mit spezieller Software/Hardware (zB Scanner) zur Abwicklung des Projekts
- Organisatorische (nicht-technische) Beratung zu Fragen zur Realisierung des Projekts

5 Präsentation



Programm....

...mit realem Bedarf

...mit Ausbaupotential

...mit Bezug zu Teammitglied



Idee

- Problemstellung:
 - Notenarchive meist noch in Papierformat
 - Zeitaufwendiger Zugriff
 - Zugriff kann nur vor Ort erfolgen
- Lösung:
 - Digitalisierung Notenarchive



Planung

- Kickoff-Meeting
- APP oder nicht APP?
- Entscheidung: Webanwendung



Funktion

Archivist:

Speichern...

Verwalten...

Suche...

von Noten-PDF`s




Mockups

Archivist TMK Wallsee-Sindelburg				
Übersicht 🗑️ +				
Suchen... 🔍				
Titel ^	Genre v	Komponist v	Schwierigkeit v	
In The End It's Yellow	Modern Jazz	Lorenz Raab	C	<input type="checkbox"/>
Nutville	Fast Latin	Horace Silver	B	<input type="checkbox"/>
Swingin' Glenn Miller	Swing	Häußer	B	<input type="checkbox"/>
The Raven Speaks	Funk	Keith Jarret	D	<input type="checkbox"/>



Mockups

**Archivist**
TMK Wallsee-Sindelburg

← In the End It's Yellow

Titel

In the End It's Yellow

Genre

Modern Jazz

Komponist

Lorenz Raab

Schwierigkeit

C

Archivnummer


I3

Noten >

Partitur	Alt Sax	Flöte	Trompete	Bariton	
			1	Sax	+



Mockups

**Archivist**
TMK Wallsee-Sindelburg

← Neue Stimme

Stimme

Alt Sax

Noten

PDF-Datei zum Speichern hierher ziehen...

Speichern

Abbrechen

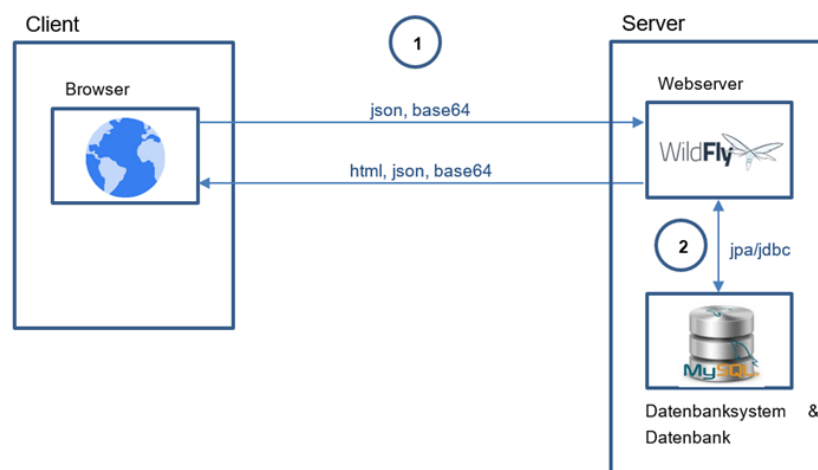


Demo

Live Test



Implementierung



Anmerkung: Client und Server befinden sich auf demselben Rechner.



Ausblick

- Switch von lokalem Server auf online Server
- Implementierung einer APP
- Vernetzung der Mitglieder
- Verwaltung von Usern
- Anlegen von „Mappen“



Danke für Ihre Aufmerksamkeit

6 Zeitaufzeichnung

Project Name		Archivist	
Current Project Week		1	
Project Type	PRO-1	3	ECTS
Project Life Span		12	Weeks
Resulting Workload		4 500	Minutes
		75	Hours
		ca. 6	Hours per week

Current Overview			
Week Nr. 1	Debit Hours (as planned)	Actual Hours	Difference
Boris Fuchs	0	4	4
Paul Schmutz	0	4	4

[illegible]

7 Meetings

7.1 1. Protokoll

Projekt Notenarchiv

Fuchs Boris, Schmutz Paul

Datum	Uhrzeit	Aufwand in Stunden	Protokollnummer
10.04.2017	8:30 – 10:30 11:00 – 12:30	3,5	1

Teilnehmer:

Fuchs Boris	Schmutz Paul

Thema

Grundlegende Strukturierung und Planung des Projekts; Brainstorming, Zieldefinition

Welche Entwicklungsumgebung? -> Java Skript für Homepage, Java Spring -> Backend

Offline + online Notenarchiv? (eventuell mit Download-Funktion)

Server? -> Simulation mit PC (zum Testen) Speicher in HDD Format verfügbar.

Qualität von (PDF-)Dokumenten -> 3 Qualitätsstufen (eventuell in späterer Ausführung des Programms)

Scannerprogramm macht das automatisch

Aktuell nur eine Qualitätsstufe

Speicherplatz am Server beschränkt? Kurze Hochrechnung des benötigten Speichers bei 1000 Musikkapellen = $1.000 \text{ Kapellen} \cdot 10.000 \text{ Seiten eines Archivs} = 10.000.000 \text{ Mb} = 10 \text{ Terabyte}$

Reduktion der Qualität des Dokuments nötig!

Eventuell Ausgabe als ZIP?

APP (im späteren Verlauf)

Verschiedene Benutzer -> Benutzerverwaltung -> verschiedene Rechte

Anwendung soll auch Offline laufen (suche nach Möglichkeiten von Java Skript auf Desktopanwendung zu kommen)

Scanner-Algorithmus -> für eingescannte Seiten (später)

ToDo`s:

Server PC-Simulation -> Java Spring

Java Skript -> Datenbank-Programm

Projektstrukturplan

Web-Service

Problemstellung

Off/Online Betrieb

Java Skript Zugriff auf internen Speicher? -> Vielleicht über Webservice?

Website als eigenständiges Programm? Für Offline Betrieb

Ionic? Phone Gap? Electron?

PDF als Einzel Bild oder als gesamt PDF ??

Ziele:

Notenarchiv über Java Skript als Webanwendung

Digitalisierung des Notenarchivs einer Kapelle/Musikverein

Auflistung des Gesamten Notenarchivs/Verwaltung-Tool

Kategorisierung nach Instrumenten

Zusammenstellen von Notenpaketen (zB Konzertprogramm, Frühschoppen-Mappe)

+Printfunktion (falls es sich im Zeitplan ausgeht) bzw. Anzeige im PDF Viewer

+Downloadfunktion (Daten sowohl auf Server als auch auf Internem Speicher

Auswahl eines Datenbanksystems (+strukturiertes Importieren der Noten)

Webservice + Backend (+Testungen)

Grafische Oberfläche

Simulation eines Servers über PC (+Testungen)

Nicht-Ziele:

Scanner-Algorithmus (zur automatischen Einsortierung von Noten) + integrierten PDF Creator (wird bei Ausgabe des Dokuments ein PDF in wahlweise 3 Qualitätsstufen erstellt)

3 Qualitätsstufen des PDF`s

Check der Technologien und deren Kompatibilität!

HTML 5 auf Desktopanwendung ummünzen

Checken der Möglichkeiten

Eventuell so wie Slack -> Electron

7.2 2. Protokoll

Projekt Notenarchiv

Fuchs Boris, Schmutz Paul

Datum	Uhrzeit	Aufwand in Stunden	Protokollnummer
11.04.2017	20:00 – 21:00	1	2

Teilnehmer:

Schmutz Paul	

Thema

Befragung Kapellmeister & Archivarin der TMK Wallsee-Sindenburg über Funktionsumfang

Gewünschter Funktionsumfang

- **Suchfunktion** anhand verschiedenen Eigenschaften der abgelegten Stücke
- Ordnen nach Kategorien & Unterkategorien (bzw. Genres)
 - Beispiel
 - Märsche
 - Konzertmärsche
 - Trauermärsche
 - Polka
 - Schneltpolka
 - Böhmisches Polka
 - Filmmusik
- Eigenschaften von Musikstücken
 - Titel
 - Kategorie/Genre/Unterkategorie
 - Komponist
 - Solostücke ja/nein + für welches Instrument
 - Schwierigkeit (A-E; A...leicht; E...schwer)
 - Hörprobe als MP3 oder Link
 - Dauer
 - usw.

Sonstiges

- Verzeichnis für Noten: www.notendatenbank.at (Zugangsdaten auf Anfrage beim Kapellmeister der TMK Wallsee-Sindelburg Josef Bierbaumer +436764205563)
 - Relevante Kriterien für das Archiv
 - Mögliche Anbindung zu diesem System in späterer Phase (Nicht-Ziel)
- Musikvereine, welche ein digitales Notenarchiv betreiben
 - Musikverein Amstetten (Kontaktperson: Sarah Holm; bereits schriftlich kontaktiert)
 - Musikverein Winklarn

7.3 3. Protokoll

Minutes of Meeting

Summary					
Meeting Topic	Erstes Meeting Projektbetreuer				
Date	02.05.17	From	10:30	To	12:00
Location	PL4.MC	Note Taker	Paul Schmutz, Fuchs Boris	Duration	1,5 hrs

Agenda	
Time	Topic
15 mins	1. Grundlegendes
75 mins	2. Planung, Mockups, Pflichtenheft

Attendee List			
#	Name	Present?	Reason of Absence
1	Boris Fuchs	Y	
2	Paul Schmutz	Y	
4	Marcel Breitenfellner	Y	

Meeting Notes	
1	Grundlegendes zur Projektstruktur
2	<p>Was müssen wir für Dokumente ausfüllen? -> alle vorhandenen Dokumente wurden durchbesprochen</p> <p>Wie sieht die allgemeine Gliederung des Projektes aus? Wann müssen wir fertig werden?</p> <p>Nächsten Termin: 23.05.17</p> <p>Fertigstellung bis zum nächsten Meeting: Mockups + Projektdokumentation + PSP</p>

--	--

Action Items			
#	Task	Person Responsible	Due Date
1	brief description	Department name OR First name Last name	Dd Mon yy
2	brief description	Department name OR First name Last name	TBD
3	brief description	Department name OR First name Last name	Dd Mon yy

7.4 4. Protokoll

Minutes of Meeting

Summary					
Meeting Topic	Zweites Meeting Projektbetreuer				
Date	23.05.17	From	11:30	To	12:30
Location	PL4.MC	Note Taker	Paul Schmutz, Boris Fuchs	Duration	1.0 hrs

Agenda	
Time	Topic
40 mins	1. Überprüfung des aktuellen Standes
20 mins	2. Implementierung

Attendee List			
#	Name	Present?	Reason of Absence
1	Boris Fuchs	Y	
2	Paul Schmutz	Y	
4	Marcel Breitenfellner	Y	

Meeting Notes	
1	Aktuelle Dokumente wurden durchbesprochen -> PSP, Mockups, Dokumentation, Time-Recording
2	Implementierung -> wie sollen wir vorgehen? -> lokaler oder webserver? -> webanwendung oder app?

Action Items

#	Task	Person Responsible	Due Date
1	brief description	Department name OR First name Last name	Dd Mon yy
2	brief description	Department name OR First name Last name	TBD
3	brief description	Department name OR First name Last name	Dd Mon yy

7.5 5. Protokoll

Projekt Notenarchiv

Fuchs Boris, Schmutz Paul

Datum	Uhrzeit	Aufwand in Stunden	Protokollnummer
27.06.2017	14:30 -15:30	1	3

Teilnehmer:

Fuchs Boris	Schmutz Paul

Thema

Projekt Weiterführung / Vorbereitung für die kommerzielle Nutzung

Erweiterung der Funktionalität:

APP -> soll für jeden Musiker einer Kappel zugänglich sein

Er kann sich alle Noten anschauen, kann aber nichts bearbeiten oder löschen

Er logt sich mit seinem Erstellen Account (erhält er von Archivar) ein, dieser beinhaltet eine Abfrage nach seinem Instrument und Stimme, damit werden ihm nur diese Noten angezeigt.

Sinn ist immer und überall auf das Archiv zugreifen zu können

Archivar

Oberfläche zum Verwalten der Berechtigungen + erstellen neuer Accounts

Archivar spielt benötigte Daten ein und lädt diese hoch

Generell

Button für das Abfragen nach bestimmten Instrumenten und Stimmen

Button für das Erstellen von Mappen (nur Archivar oder auch User?)

Button/Möglichkeit nach einzelnen Instrumenten zu filtern/abzufragen

Erstellung einer Website (erste Seite ist Info über das Programm + Login Button
-> Weiterleitung zum eigentlich Programm) (Wie werden Updates eingespielt)

Implementierung

Von Java Spring auf Windows Azure -> leichteres Servermanagement

Finanzielles

Kosten für Server? Wieviel Speicher brauchen wir?

Wieviel wird das Programm Kosten? Monatliche Abrechnung?

Registrierung nur mit Angabe von Bezahlungsmöglichkeit möglich? Oder ein freier Testmonat und dann abfrage nach Bezahlungsart?

7.6 6. Protokoll

Minutes of Meeting

Summary					
Meeting Topic	Abgabevorbereitungen				
Date	29.06.2017	From	11:30 AM	To	12:15 PM
Location	FH (PL1.MC)	Note Taker	Paul Schmutz	Duration	0.75 hrs

Agenda	
Time	Topic
15 mins	1. Präsentation- Fragen zum Ablauf und Inhalt
30 mins	2. Verbleibende Dokumentation- Fehlende Dokumente feststellen und Inhalt fixieren

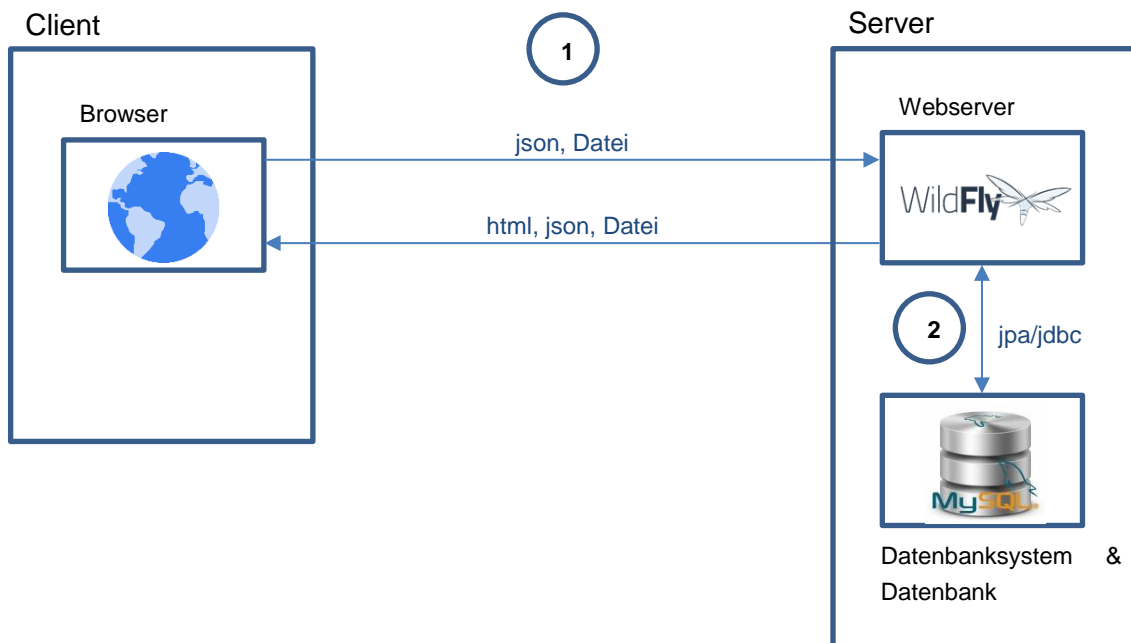
Attendee List			
#	Name	Present?	Reason of Absence
1	Marcel Breitenfellner	Y	
2	Boris Fuchs	Y	
4	Paul Schmutz	Y	

Meeting Notes	
1	<p>Präsentation:</p> <p>Inhaltsverzeichnis nicht notwendig</p> <p>Soll eine Verkaufspräsentation sein!</p> <p>Schritte von Idee über Planung zur Umsetzung</p> <p>Planung, Mockups</p> <p>Technische Tools nur kurz</p> <p>Logos von Tools auf verschiedenen Folien nebenbei einfügen als Platzfüller</p> <p>Zuerst Mockups/Planung, später erst kurz technisches (Architektur)</p>

	Beschreibung der Mockups Ausblick auch erwähnen, kommt gut an Live-Demo kommt gut an 10-12 min Präsentation (5 min, 10 min verbleibend Schilder) Anschließend Präsentation im L1/L2: Plakat in Deutsch ausdrucken!		
2.1	Website-Template Kurzer Text vom Abstract hineinkopieren ODER Text von Plakat		
2.2	System-Architektur Vom Pflichtenheft hineinkopieren +ERD erklären, Beziehungen erklären		
2.3	Demo Download von PDFs→besser in neuen Tab aufmachen		
2.4	Installations-Guide Wie wird Projekt für Entwicklung aufgesetzt (Installation.docx)		
2.5	Source-Dokumentation Kommentare anfertigen!		
Action Items			
#	Task	Person Responsible	Due Date
1	Plakat in Deutsch A3 ausdrucken	Schmutz, Fuchs	03.07.2017

8 Detaillierte Systemdokumentation

8.1 Systemschnittstellen



1 Client-Server

Protokoll: HTTP

Datenformate: HTML (für Benutzerschnittstelle für den Browser)
JSON (für die im Browser dargestellten Anwendungsdaten)
Datei (für zu übertragende PDF-Noten-Dateien)

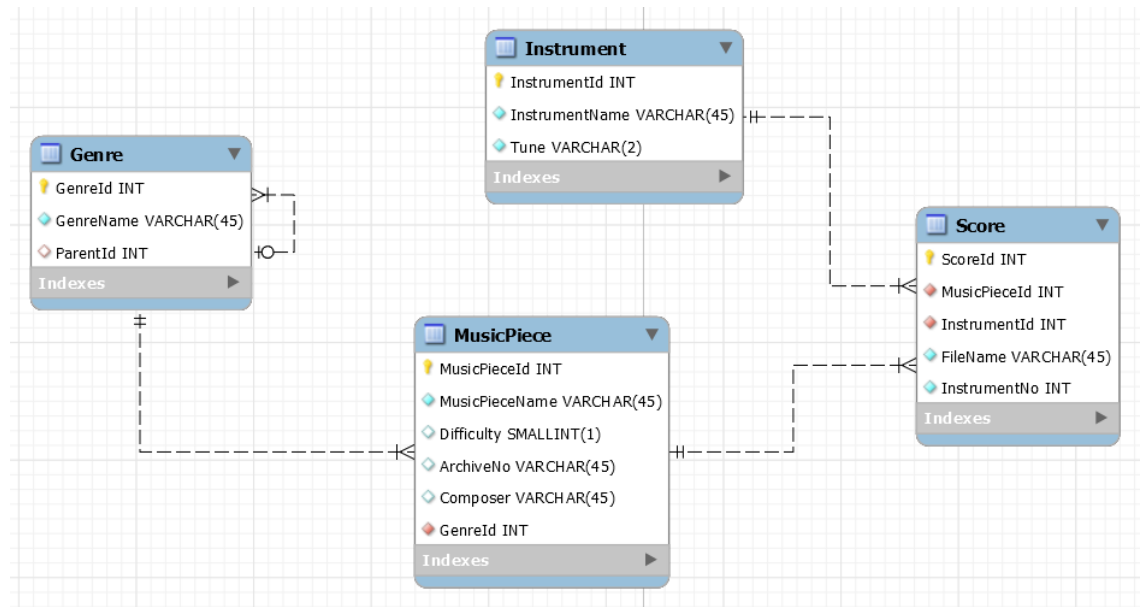
2 Webserver-Datenbank

Protokoll: JPA/JDBC

8.2 Datenbankmodell

8.2.1 ERD

Nachdem als Datenbank MySQL zum Einsatz kommt, liegt auch ein entsprechendes MySQL-Workbench-Datenbankmodell vor:



Basierend auf dem Modell kann die Datenbank mithilfe von Forward Engineering generiert werden. Weiters können in Java anschließend davon Modellklassen mittels JPA Tools abgeleitet werden.

8.2.2 Entitäten

8.2.2.1 MusicPiece

Die Entität MusicPiece stellt einen fundamentalen Baustein im Notenarchiv da und repräsentiert ein Musikstück.

Eigenschaften:

MusicPieceld	Eindeutige ID eines Musikstücks (Primärschlüssel)
MusicPieceName	Titel des Musikstücks; nicht-optional
Difficulty	Schwierigkeit des Musikstücks; 1-5 (wobei 1 einen einfachen Schwierigkeitsgrad bedeutet); optional
ArchiveNo	Archivnummer je nach Nummerierungssystem des Musikvereins; optional
Composer	Komponist des Musikstücks; optional
GenreId	Verweis auf das dazugehörige Genre des Musikstücks; nicht-optional

8.2.2.2 Genre

Die Entität Genre stellt Musikgenres für Musikstücke bereit und wird dabei hierarchisch aufgebaut. D. h. ein Genre kann mehrere Subgenres besitzen (beispielsweise könnten „Konzertmarsch“ und „Traditionsmarsch“ Subgenres von „Marsch“ sein).

Eigenschaften:

GenreId	Eindeutige ID eines Genres (Primärschlüssel)
GenreName	Bezeichnung des Genres; nicht-optional
ParentId	Optionaler Verweis auf ein übergeordnetes Genre

8.2.2.3 Instrument

Die Entität Instrument wird verwendet, um Musikinstrumente zu Einzelstimmen von Musikstücken zuzuordnen.

Eigenschaften:

InstrumentId	Eindeutige ID eines Musikinstruments (Primärschlüssel)
InstrumentName	Name des Musikinstruments; nicht-optional
Tune	Stimmung des Musikinstruments, zB Altsaxophon ist in „Eb“ gestimmt; nicht-optional

8.2.2.4 Score

Die Entität Score repräsentiert eine Einzelstimme eines Musikstücks. Einzelstimme bedeutet, dass vorliegende Noten (PDF-Datei) eines Musikstücks für genau ein Musikinstrument geeignet ist. So existieren für ein Musikstück beispielsweise Einzelstimmen für Tuba, Klarinette, Trompete usw. und auch die Partitur (Sammlung aller Stimmen für den Dirigenten; Partitur wird wie ein Instrument gehandhabt).

Eigenschaften:

ScoreId	Eindeutige ID einer Einzelstimme (Primärschlüssel)
MusicPieceId	Verweis auf das dazugehörige Musikstück; nicht-optional
InstrumentId	Verweis auf das Musikinstrument; nicht-optional
FileName	Dateiname der gespeicherten PDF-Datei für die Einzelstimme; nicht-optional
InstrumentNo	Nummer des Instruments: Fast immer liegt für Musikstücke eine Mehrfachbesetzung von Instrumenten vor. Beispielsweise gibt es für das Klarinettenregister meist je eine Einzelstimme für die 1. Klarinette, 2. Klarinette bis zur 4. Klarinette. Zur Unterscheidung von Einzelstimmen innerhalb des Instrumentenregisters dient diese Instrumentennummer.

	nicht-optional
--	----------------

8.2.3 Beziehungen

8.2.3.1 Genre – Genre

Genre: 0, 1 Genre: 1, n

Erläuterung:

Ein Genre kann mehrere Subgenres besitzen (optional). Das bedeutet gleichzeitig, dass ein Genre kein oder ein Übergenre besitzt.

8.2.3.2 MusicPiece – Genre

MusicPiece: 1, n Genre: 1

Erläuterung:

Ein Musikstück muss genau einem Genre zugeordnet werden. Genres können mehreren Musikstücken zugewiesen sein (weil zu einem Musikgenre meist mehrere Musikstücke abgelegt sind).

8.2.3.3 MusicPiece – Score

MusicPiece: 1 Score: 1, n

Erläuterung:

Ein Musikstück besteht aus mehreren Einzelstimmen (Noten) und eine Einzelstimme gehört genau zu einem Musikstück.

8.2.3.4 Instrument – Score

Instrument: 1 Score: 1, n

Erläuterung:

Eine Einzelstimme ist genau für ein Instrument spezifiziert. Ein Instrument kann natürlich für mehrere verschiedene Einzelstimmen bestimmt sein (einerseits durch mehrere Musikstücke sowie durch Mehrfachbesetzung eines Musikinstruments bei einem Musikstück).