

**Requirements Specification**

Semesterprojekt WS 2017/18

3D-Scanner

Version 1

# Inhaltsverzeichnis

Inhaltsverzeichnis 2

1 Einleitung 3

1.1 Zweck des Dokuments 3

1.2 Gültigkeit des Dokuments 3

1.3 Begriffsbestimmungen und Abkürzungen 3

1.4 Zusammenhang mit anderen Dokumenten 4

1.5 Überblick über das Dokument 4

2 Allgemeine Beschreibung des Produkts 5

2.1 Projektumgebung 5

2.1.1 Übersicht 5

2.1.2 Kontaktdaten 6

2.2 Zweck des Produkts 6

2.3 Überblick über die geforderte Funktionalität 6

2.4 Allgemeine Einschränkungen 6

2.5 Vorgaben zu Hardware und Software 7

2.6 Benutzer des Produkts 7

3 Detaillierte Beschreibung der geforderten Produktmerkmale 8

3.1 Lieferumfang 8

3.2 Abläufe (Szenarien) von Interaktionen mit der Umgebung 9

3.3 Externe Schnittstellen des Produkts 10

3.3.1 Benutzerschnittstellen (User Interfaces) 10

3.3.2 Systemschnittstellen 15

4 Vorgaben an die Projektabwicklung 16

4.1 Anforderungen an die Realisierung 16

4.1.1 Hardware 16

4.1.2 Software 16

4.2 Fertige und zugekaufte Komponenten 18

4.3 Unterauftragnehmer 18

4.4 Lieferbedingungen 19

5 Verpflichtungen des Auftraggebers 20

# Einleitung

## Zweck des Dokuments

## 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.

## 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 |

## Zusammenhang mit anderen Dokumenten

Derzeit liegt weder ein Angebot noch ein Lastenheft vor.

## Überblick über das Dokument

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

# Allgemeine Beschreibung des Produkts

## Projektumgebung

### Übersicht



**FH Hagenberg**

Auftragnehmer

Boris Fuchs

Paul Schmutz



Auftraggeber und Projektkoordination

Dipl. Ing. Dr. Erik Sonnleitner



Projektauftrag,

Projektkoordination

Beratung,

Fachwissen

Projektbetreuung

Marcel Breitenfellner

### Kontaktdaten

|  |
| --- |
| **Dipl. Ing. Dr. Erik Sonnleitner**  Telefon: +43 5 0804 22823  E-Mail: erik.sonnleitner@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 |

## 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

## 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.

## 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.

## 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 4.1.2.2 Entwicklungsumgebung, -software und -bibliotheken) zur Anwendung.

## 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.

# Detaillierte Beschreibung der geforderten Produktmerkmale

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

## Lieferumfang

Das fertige auslieferbare Endprodukt enthält zwei wesentliche Bestandteile:

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

## 3D-Skizze des Gerüsts

## Abläufe (Szenarien) von Interaktionen mit der Umgebung

Dem Benutzer des Systems stehen bestimmte Funktionalitäten zur Verfügung, welche für folgende Szenarien geeignet sind.

Use-Case-Diagram

## Externe Schnittstellen des Produkts

### Benutzerschnittstellen (User Interfaces)

### Systemschnittstellen

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

Server

Webserver



Datenbanksystem & Datenbank

jpa/jdbc

Client

Browser



json, base64

html, json, base64

**Anmerkung**: Client und Server befinden sich auf demselben Rechner.

**Client-Server**

Protokoll: HTTP

Datenformate: HTML (für Benutzerschnittstelle für den Browser)

JSON (für die im Browser dargestellten Anwendungsdaten)

Base64 (für zu übertragende Dateien)

**Webserver-Datenbank**

Protokoll: JPA/JDBC

# Vorgaben an die Projektabwicklung

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

## Anforderungen an die Realisierung

### 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)

### Software

#### 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

#### Entwicklungsumgebung, -software und -bibliotheken

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

|  |  |
| --- | --- |
| **Spring Tool Suite (STS)**  Bildergebnis für spring tool suite | STS dient hauptsächlich als Entwicklungsumgebung für das Spring Boot Backend.  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).  Nachdem STS eine modifizierte Version von Eclipse ist, werden die den Plattformen Windows, Linux und Mac unterstützt. |

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!

## 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.

## Unterauftragnehmer

Folgende Abhängigkeiten von Unterauftragnehmern bestehen:

|  |  |  |  |
| --- | --- | --- | --- |
| **Lieferobjekt** | **Unterauftragnehmer** | **Liefertermin** | **Status** |
|  |  |  |  |

## 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>

# 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