

**Installation**

Semesterprojekt SS 2017

Archivist

# Inhaltsverzeichnis

Inhaltsverzeichnis 2

1 Einleitung 3

1.1 Projektstruktur 3

1.2 Überblick der verwendeten Software 3

2 Backend 5

2.1 Datenbank-Server 5

2.1.1 A) Aufsetzen mit Command Line 5

2.1.2 B) Aufsetzen mit MySQL Workbench 5

2.2 Webserver 5

3 Frontend 6

4 Start-Programm 7

5 Debugging 8

6 Release 9

# Einleitung

## Projektstruktur

Dieses Dokument dient dazu, projektfremden Personen das Aufsetzen der Entwicklungsumgebung für das Notenarchiv-Programm Archivist zu ermöglichen.

Archivist ist in drei Softwareteilbereiche gegliedert:

* **Backend**: ein Spring Boot Webservice, der eine HTTP-Schnittstelle darstellt und Daten in einer MySQL-Datenbank verwaltet; dieser Webservice wird auf einem WildFly Application Server angeboten
* **Frontend**: eine Angular JS Web-Applikation, die das Notenarchiv grafisch aufbereitet und dargestellte Daten vom Backend bezieht; das Frontend wird ebenfalls im WildFly Application Server integriert
* **Start-Programm**: eine für den Endbenutzer vorgesehene Java-Applikation für Windows, welche bei Aufruf den WildFly Application Server startet, die Frontend-Oberfläche im Standard-Browser öffnet sowie ein Desktop-Verknüpfung für dieses Programm erstellt

Genaueres zur Systemarchitektur finden Sie im Dokument /Documentation/TechnicalDocumentation/**SystemArchitecture.docx**.

## Überblick der verwendeten Software

Zur Entwicklung der drei erwähnten Teilbereiche sind einige Software-Werkzeuge notwendig, um Source-Code zu verfassen, testen sowie eine Release-Version zu erstellen.

Systemvoraussetzung für dieses Projekt ist **Windows 10**.

Folgende Software wird zur Entwicklung von Archivist verwendet (wie und wo eine Software jeweils verwendet wird, wird in späteren Abschnitten erläutert):

|  |  |
| --- | --- |
| **Software und Version** | **Download-Link** |
| **Java JDK 1.8.0\_131 (32 bit)** | [http://www.oracle.com/technetwork/java/ javase/downloads/jdk8-downloads-2133151.html](http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html) |
| **Spring Tool Suite (STS) 3.8.4**  Bildergebnis für spring tool suite | <https://spring.io/tools/sts> |
| **Wildfly 10.1.0 Final**  Bildergebnis für wildfly | <http://wildfly.org/downloads/> |
| **MySQL Server 5.7.18 (64 bit)**  Bildergebnis für mysql | <https://dev.mysql.com/downloads/installer/> |
| **MySQL Workbench 6.3.9 CE (64 bit)**  Bildergebnis für mysql workbench logo | <https://dev.mysql.com/downloads/installer/> |
| **MiniServer MySQL 5.0.67** | [https://sourceforge.net/projects/miniserver/files/ MiniServer/MiniServer\_ %20MySQL%205.0.67%20Portable/](https://sourceforge.net/projects/miniserver/files/MiniServer/MiniServer_%20MySQL%205.0.67%20Portable/)  mini\_server\_11.zip |
| **Node.js 6.10.3 (64 bit)**  C:\Users\Paul\AppData\Local\Microsoft\Windows\INetCache\Content.Word\download.png | <https://nodejs.org/en/download/> |
| **Brackets 1.9 (32 bit)**  C:\Users\Paul\AppData\Local\Microsoft\Windows\INetCache\Content.Word\download.jpg | <http://brackets.io/> |
| **Eclipse Neon 3 (32 bit)**  Home | <http://www.eclipse.org/downloads/packages/eclipse-ide-java-developers/neon3> |

# Backend

Für das Backend müssen einerseits ein MySQL-Server und andererseits ein Spring Boot Webservice aufgesetzt werden.

## Datenbank-Server

### Installation von MySQL

Bei der Installation vom MySQL Server werden der Einfachheit halber folgende Zugangsdaten definiert:

User: **root**

Password: **admin**

### Aufsetzen der Datenbank

Für das Einrichten des Datenbank-Servers bestehen zwei Wege:

#### A) Aufsetzen mit Command Line

Voraussetzungen:

* MySQL Server
* SQL-Skripts:
  + /Implementation/SourceCode/init\_db.sql
  + /Implementation/SourceCode/TestData/archivist\_genre.sql
  + /Implementation/SourceCode/TestData/archivist\_instrument.sql
  + /Implementation/SourceCode/TestData/archivist\_musicpiece.sql
  + /Implementation/SourceCode/TestData/archivist\_score.sql

Bei dieser Variante werden die SQL-Skripts zur Erstellung der Datenbank sowie der Testdaten per Command Line eingespielt.

1. Windows Command Line Prompt im bin-Verzeichnis von MySQL öffnen (zB D:\Program Files\MySQL\MySQL Server 5.7\bin)
2. mysql --user=admin --password=admin
3. Erstellen der Datenbank und Tabellen:  
   mysql>source init\_db.sql
4. Einfügen von Testdaten und Neuerzeugung von Tabellen (Skripts können jederzeit erneut ausgeführt werden, um alle vorhandenen Testdaten zu löschen und die in den Skripts definierten Testdaten wieder in die leere Tabellen einzufügen):  
   mysql>source archivist\_genre.sql  
   mysql>source archivist\_instrument.sql  
   mysql>source archivist\_musicpiece.sql  
   mysql>source archivist\_score.sql

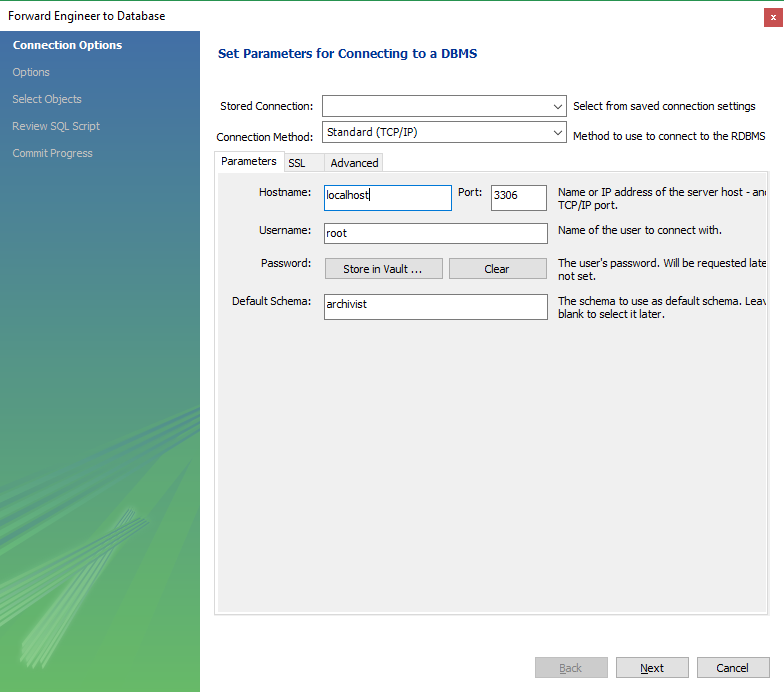
#### B) Aufsetzen mit MySQL Workbench

Voraussetzungen:

* MySQL Server
* MySQL Workbench
* /Implementation/ProjectFiles/archivist\_erd.mwb
* SQL-Skripts:
  + /Implementation/SourceCode/TestData/archivist\_genre.sql
  + /Implementation/SourceCode/TestData/archivist\_instrument.sql
  + /Implementation/SourceCode/TestData/archivist\_musicpiece.sql
  + /Implementation/SourceCode/TestData/archivist\_score.sql

Mithilfe der MySQL Workbench wird mittels Forward Engineer die Datenbank erstellt und die Testdaten mit dem Data Import Tool eingefügt.

1. Öffnen des Datenbankmodells **archivist\_erd.mwb** in der MySQL Workbench
2. Datenbank erzeugen: Database>Forward Engineer



1. Lokale Instanz öffnen und Daten importieren: Server>Data Import>Auswählen des Ordners TestData (SQL-Skripts für Testdaten)

## Webserver

Voraussetzungen:

* Installation von Java
* WildFly
* Spring Tool Suite

### Einrichten von WildFly

MySQL

Add-user

Datenquelle einrichten

### Einrichten der Spring Tool Suite

Market Place:

Eclipse Data Tools (🡪Data Tools Platform Enablement [besonders für MySQL])

JBoss Tools

Data Source Explorer

Hinzufügen von WildFly im Server Explorer

Öffnen des Projekts backend/archivist

Run on Server > WildFly

# Frontend

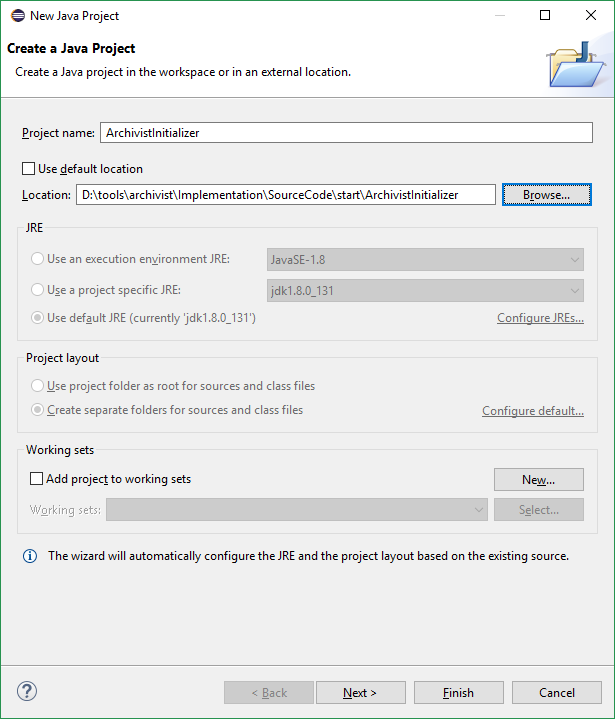
# Start-Programm

Voraussetzungen:

* Eclipse Neon
* /Implementation/SourceCode/start/ArchivistInitializer

Nach der Installation von Eclipse Neon kann das Projekt **ArchivistInitializer** geöffnet werden.

File>New>Java Project…



**Anmerkung**: Sollten während der Entwicklung Assets (zB Bild-Dateien oder Batch-Skripts) im Source-Folder aktualisiert bzw. neue hinzugefügt werden, ist das Ausführen von Project>Clean… für dieses Projekt hilfreich, damit beim Debuggen keine Assets von vorigen Builds verwendet werden.

# Debugging

# Release