

# SimTV MAM

Ihr Metadaten - Verwalter



**Semesterarbeit im Modul PA\_5**

**von**

**Simon Wittker**

**Gruppe\_ZH\_7; HS 21/22**

**Turbenthal 23.12.2021**

## Dokumentenversionierung

Version	Datum	Änderungen	Autoren
0.1	11.09.	Initiale Erstellung	Simon W.
0.2	15.10	Inhalt nach Statusmeeting anpassen	Billy U.
0.3	08.11.	Ganze Doku nach personellen Änderungen anpassen	Simon W.
0.4	15.11.	Komplettierung nach Abschluss Phase 3	Simon W.
0.5	21.12.	Komplettierung nach Abschluss Phase 4	Simon W.
1.0	23.12.	Finale Fassung	Simon W.

## Inhaltsverzeichnis

<b>1 Projektmanagement.....</b>	<b>1</b>
1.1 Projektauftrag .....	1
1.1.1 Ausgangslage.....	1
1.1.2 Projektziele und Inhalte .....	2
1.1.3 Rahmenbedingungen .....	3
1.1.4 Randbedingungen und Eckwerte .....	4
1.1.5 Grundlagen.....	5
1.1.6 Begründeter Lösungsansatz .....	5
1.1.7 Nutzwertanalyse.....	6
1.1.8 Risikoeinschätzung .....	8
1.1.9 Probleme .....	10
1.1.10 Messbare Lieferobjekte .....	10
1.2 Projektplanung .....	12
1.2.1 Projektstrukturplan .....	12
1.2.2 Meilensteine .....	12
1.2.3 Netzplan mit Meilensteinen .....	13
1.2.4 Terminplanung .....	14
1.2.5 Arbeitspakete .....	21
<b>2 Software-Engineering.....</b>	<b>33</b>
2.1 Analysedokumente.....	33
2.1.1 Stakeholder .....	33
2.1.2 Systemidee.....	33
2.1.3 Produktanforderungen.....	34
2.1.4 Qualitätszieleinschätzung.....	35
2.1.5 Domänenmodell .....	35
2.1.6 UseCase.....	36
2.1.7 Schnittstellenbeschreibung .....	41
2.2 Softwareentwurfsdokumente.....	43
2.2.1 Architekturpattern .....	43
2.2.2 Sichten der Softwarearchitektur .....	45
2.2.3 Klassendiagramm.....	47
2.2.4 Dynamische Modelle .....	58
2.2.5 Spezifikation der Bedienoberflächen.....	62

2.2.6	Testkonzept, Tests und Qualitätssicherung .....	73
2.3	Inbetriebnahme .....	94
2.3.1	Lieferobjekt.....	94
2.3.2	Installationsanleitung .....	94
2.3.3	Benutzerhandbuch .....	99
<b>3</b>	<b>Code .....</b>	<b>106</b>
3.1	Code-Richtlinien.....	106
3.1.1	Richtlinien für die Namensgebung: .....	106
3.1.2	Richtlinien für die Programmieren .....	106
3.1.3	Richtlinien für die Code-Dokumentation.....	106
3.1.4	Begründung .....	107
<b>4</b>	<b>Abschlussbericht .....</b>	<b>108</b>
4.1	Evaluationsbericht.....	108
4.1.1	Übersicht über den Projektverlauf .....	108
4.1.2	Ziele erreicht?.....	108
4.1.3	Schwierigkeiten? → Ja .....	108
4.1.4	Sonnen- und Schattenseiten .....	109
4.1.5	Mein Fazit .....	109
<b>5</b>	<b>Literaturverzeichnis .....</b>	<b>110</b>
<b>6</b>	<b>Abbildungsverzeichnis .....</b>	<b>111</b>
<b>7</b>	<b>Tabellenverzeichnis .....</b>	<b>113</b>
<b>8</b>	<b>Anhang .....</b>	<b>116</b>
8.1	Akzeptanzerklärung .....	116
8.2	Entscheidungs-Log .....	117
8.2.1	Übersicht .....	117
8.2.2	Die einzelnen Entscheidungen.....	118
8.3	Protokolle .....	128
8.3.1	Protokoll vom Telefonmeeting vom 13.09.21 .....	128
8.3.2	Protokoll vom Teamsmeeting vom 23.09.21 .....	129
8.3.3	Protokoll vom Teams-Meeting vom 13.10.21 .....	130
8.3.4	Protokoll vom Telefonmeeting vom 28.10.2021 .....	132
8.3.5	Statusberichte und Protokolle der Statusmeetings .....	133
8.3.6	Sonstiges.....	146

## Glossar und Abkürzungen

Nr.	Begriff	Bedeutung
	Asset	Der Begriff kommt aus der Finanzwirtschaft und bedeutet Vermögen oder Vermögenswert.
	Media Asset	Informationsobjekte wie z.B. Bilder, Text oder Video zusammen mit den Metainformationen z.B. wie Titel, Ersteller oder Erstellungsdatum. Ein verwertbares Medienobjekt.
	MAM	Kurz für Media Asset Management.
	MVP	Minimum Viable Product
	RE	Requirements Engineer
	DB	Datenbank
	CRUD	Basisoperationen eines Programms, steht für create, read, update und delete.
	GUI	Graphic Unit Interface, engl. für Benutzeroberfläche
	SQL	Structured Query Language, engl. für Strukturierte Abfrage-Sprache
	NoSQL	No Structured Query Language, also keine SQL
	PM	Projektmanagement
	SWE	Software-Engineering
	OOP	Objekt-Orientiertes Programmieren
	Obj.	Kurz für Objekt.
	prog.	Kurz für programmieren
	jdk	Kurz für Java SE Development Kit

Tabelle 1: Glossar und Abkürzungen

# 1 Projektmanagement

## 1.1 Projektauftrag

### 1.1.1 Ausgangslage

#### 1.1.1.1 Ist-Situation

*SimTV* ist ein regionaler Fernsehsender in der Ostschweiz. In einem Fernsehunternehmen müssen viele technische Systeme zusammenarbeiten und die Prozesse sind komplex. Beispiele davon sind Programmplanungs-, Workflowautomatisierung- und Reportingssysteme. Alle basieren aber auf einer zentralen Datenbank, die alle zur Verfügung stehende Medienobjekte und deren Metadaten beinhaltet.

Anstatt eine existierende Software zu kaufen, hat *SimTV* vor ungefähr 20 Jahren ein eigenes System mit der Software «FileMaker» entwickelt. Mit dem Aufkommen vom Internet und den vielen verschiedenen neuen Medienformate (Podcast, Social Media, Website, usw.) kommt nun dieses System an seine Grenzen, weshalb *SimTV* den Auftrag erteilt hat, das System neuzudenken und vom Grund neu zu konzipieren. Der erste Schritt soll ein erweiterbares System sein, um alle Medienobjekte samt Metadaten zu verwalten. Ein so genanntes Media Assets Management System.

#### 1.1.1.2 Definition Media Assets Management System

*“Das Media Asset Management sorgt für die Aufnahme, Speicherung, Kategorisierung, Indizierung und Bereitstellung von Media Assets mit der Prämisse, eine hohe Bearbeitungsgeschwindigkeit und eine optimale Wiederverwendung zu ermöglichen. Ein Media Asset Management System verwaltet beliebige Inhalte über Metadaten und Datenbanken.“* [1]

#### 1.1.1.3 Soll-Konzept

Das System soll, um Platz und Ressourcen zu sparen, die Metadaten der Media Assets verwalten statt des gesamten Media Asset. Ferner sollen neue Metadaten über eine Maske eingegeben und in einer DB gespeichert werden. Diese Eingabemaske sollte erweiterbar sein. Es soll auch möglich sein, die Metadaten in einen Ordner auf dem PC zu speichern und von dort wieder in die Applikation zu laden. Das wichtigste ist aber eine Suchfunktion, die die DB (nicht das Ordner) durchsuchen soll. Diese soll mithilfe von Suchbegriffen bestimmte Metadaten finden und anzeigen können. Die Zugriffszeit soll unter 3 Sekunden liegen. Wenn keine

Metadaten mit den Suchbegriffen vorhanden sind, so soll das auch nach spätestens 5 Sekunden angezeigt werden. Die Suchresultate sollen bearbeitbar sein. Die Suchergebnisse sollten mittels Auswahl «neuste Ergebnisse zuerst» oder «älteste Ergebnisse zuerst» angezeigt werden. Entscheidend soll dabei das Erstellungsdatum sein. Die Applikation sollte über eine User- und deren Rechte-Verwaltung verfügen. Die Applikation sollte mit einem Passwort gesichert sein.

## 1.1.2 Projektziele und Inhalte

### 1.1.2.1 Projektziele

Das Ziel in diesem Projekt ist ein lauffähiges Media Asset Management-System. Dabei sollen folgende Ziele erreicht werden:

ID	Ziel	Muss/Kann
P01	Metadaten erstellen, einsehen, bearbeiten und löschen (CRUD)	M
P02	Funktionierendes GUI vorhanden	M
P03	Metadaten in einer DB speichern und aus der DB laden	M
P04	Metadaten in ein Ordner speichern und aus einem Ordner laden	K
P05	Funktionierende Suchfunktion vorhanden	M
P06	Zugriffszeit der Suchfunktion < 3 Sekunden	M
P07	Suchalgorithmus < 5 Sekunden terminieren	M
P08	Erweiterbare Eingabemaske vorhanden	K
P09	Suchresultate nach Erstellungsdatum sortiert angezeigt	K
P10	User- mit Rechteverwaltung vorhanden	K
P11	Passwortgeschützt	K
P12	Potenzielle Auftraggeber können das Projekts gebrauchen und darauf aufbauen.	K

Tabelle 2: Projektziele

### 1.1.2.2 Minimum Viable Product (MVP)

Das MVP hat folgende Eigenschaften:

- Metadaten erstellen, einsehen, bearbeiten und löschen (CRUD)
- Metadaten in einer DB speichern
- Die DB mit Suchbegriffen nach Metadaten durchsuchen
- Ein funktionierendes GUI

### 1.1.2.3 Grobanforderungen

- Die Software verfügt über ein GUI.
- Die Software kann mithilfe einer Eingabemaske neue Metadaten der DB oder einem Ordner hinzufügen.
- Die Software kann von der DB und von einem Ordner Metadaten abrufen.
- Die Software verfügt über eine Suchfunktion, die mithilfe eines Suchalgorithmus die DB nach den Suchbegriffen durchsucht.
- Die Suche soll innerhalb von 3 Sekunden die Ergebnisse anzeigen, oder innerhalb von 5 Sekunden anzeigen, dass keine Resultate zu finden sind und terminieren.
- Die Resultate der Suchfunktion werden in dem GUI angezeigt.
- Die einzelnen Suchresultate werden mithilfe eines Mausklicks detailliert angezeigt.
- Die Detailansicht der Suchresultate enthält die Option, diese in einem Editor zu Bearbeiten und wieder abzuspeichern oder zu löschen.

### 1.1.2.4 Abgrenzung

- Die User- und Rechteverwaltung wird in diesem Projekt nicht realisiert.
- Die Verwaltung sowie die Bearbeitung der Mediafiles ist nicht Teil des Projekts
- Die Lizenzverwaltung, die zu einem MAM gehört, wird nicht realisiert.

## 1.1.3 Rahmenbedingungen

### 1.1.3.1 Technische Vorgaben (FFHS)

Als Programmiersprache ist hauptsächlich, d.h. zu min 51%, Java zu verwenden. Die Regeln und Sprachmittel der objektorientierten Programmierung sollen beachtet werden.

Die Nutzung von weiteren Sprachen ist möglich.

### 1.1.3.2 Technische Ressourcen

Die Applikation wurde mit Java SE Development Kit 17.0.1, kurz jdk17, programmiert und soll mit einer aktuellen Version des Java Runtime Environment ausführbar sein. Für frühere jdk-Versionen wird keine Garantie gegeben, dass die Applikation fehlerfrei ausführbar ist.

## 1.1.4 Randbedingungen und Eckwerte

### 1.1.4.1 Projektdauer

Das Projekt startet offiziell am 28.08.2021 mit einem Kick-Off-Event in der Gleisarena in Zürich und endet mit der finalen Abgabe am 23.12.2021 und der Schlusspräsentation am 14.01.2022.

### 1.1.4.2 Aufwand

Die geplante Bearbeitungszeit dieses Projektes beträgt ca. 150 Stunden.

### 1.1.4.3 Phasentermine

Dieses Projekt verwendet das Phasenkonzept vom Lehrmittel «Handbuch Projektmanagement» [2]. Wir passen die Phasen sinnvoll auf das Projekt an.



Abbildung 1: Phasenmodell nach "Handbuch Projektmanagement"

Für unseres Projekt haben wir uns für das Wasserfallmodell entschieden, ein lineares (nicht iteratives) Vorgehensmodell, dass sich für dieses Projekt mit vorgegebenen Anforderungen und Leistungen präzise beschreiben lässt.

Nr.	Phase	Beschreibung
1	Beauftragung	<i>Auftrag bekommen, Planung der nächsten Phase, grobe Schätzungen, Projektantrag freigeben.</i>
2	Initialisierung	<i>"Im Rahmen der Initialisierungsphase müssen verbindliche Aussagen zu Machbarkeit, Risiken und Stakeholdern erarbeitet werden. Wesentliche Grundlagen dazu sind die Analyse der aktuellen Situation sowie klar vereinbarte Ziele und die Formulierung der Anforderungen an das Resultat des Projekts, z. B. an das zu entwickelnde Produkt.."</i>
3	Konzeption	<i>"In der Konzeptphase werden zu den Anforderungen Lösungsvarianten ausgearbeitet. Für die ausgewählte Lösungsvariante erfolgt eine detaillierte Ausarbeitung des Lösungskonzeptes. Basierend auf dem Lösungskonzept erfolgt die Detailplanung des Projektes."</i>

<b>4</b>	Realisierung	<i>“In der Realisierungsphase werden die Pläne aus der Konzeptphase verwirklicht.”</i>
<b>5</b>	Einführung/Ab-schluss	Übergabe organisieren, Abschlussbericht und Schlussrechnung erstellen. Projektdokumentation ergänzen, Projektbeurteilung und «Lessons Learned» verarbeiten. Evaluierungsbericht erstellen.

Tabelle 3: Projektphasen

## 1.1.5 Grundlagen

Mir ist keine bereits geleistete Arbeit bekannt, auf der ich dieses MAM aufbauen könnte. Daher wurde die Applikation von Grund auf eigenständig erstellt.

## 1.1.6 Begründeter Lösungsansatz

Bevor ich die verschiedenen Varianten, die sich als Lösungsansatz eignen, vorstelle, definiere ich die Muss- und Wunschkriterien.

### 1.1.6.1 Musskriterien

Gemäss meinen Grobanforderungen habe ich die nachfolgenden Musskriterien definiert:

Ziel	Beschreibung
<b>MZ01</b>	Das System muss eine Benutzeroberfläche zur Verfügung stellen.
<b>MZ02</b>	Das System muss neue Metadaten über eine Eingabemaske hinzufügen können.
<b>MZ03</b>	Das System muss Metadaten bearbeiten und löschen können.
<b>MZ04</b>	Das System muss die Metadaten in eine DB speichern können.
<b>MZ05</b>	Das System muss Daten aus der DB laden können.
<b>MZ06</b>	Das System muss die Metadaten in einen lokalen Ordner speichern können
<b>MZ07</b>	Das System muss Daten aus einem lokalen Ordner laden können.
<b>MZ08</b>	Das System muss über eine Suchfunktion, die die DB nach Suchbegriffen durchsucht, verfügen.
<b>MZ09</b>	Die Suchfunktion muss innerhalb von 3 Sekunden Ergebnisse anzeigen oder innerhalb von 5 Sekunden terminieren.
<b>MZ10</b>	Das System muss die Suchresultate auf dem GUI anzeigen können.
<b>MZ11</b>	Die Suchresultate müssen in der Detailansicht angezeigt, bearbeitet und wieder gespeichert werden können.
<b>MZ12</b>	Die Suchresultate müssen in der Detailansicht aus der DB gelöscht werden können

Tabelle 4: Musskriterien

Für die Realisierung des MAM-Systems bieten sich mehrere Varianten an:

Als «Backend» kommen die Programmiersprachen Java und Python in Frage.

Als GUI kommt eine Webanwendung mit HTML5, CSS und JS, eine GUI mit JavaFX oder eine GUI mit JavaFXML programmiert in Frage.

Als DB kommen eine Oracle-DB mit SQL, eine MongoDB mit NoSQL in Frage, oder selbst eine DB programmieren.

Für die Wahl des richtigen Lösungsansatzes stelle ich die einzelnen Lösungsaspekte in einem morphologischen Kasten dar:

Lösungsausprägungen			
Backend	Java	Python	
GUI	Webanwendung	JavaFX	JavaFXML
DB	Oracle DB	MongoDB	Selbst prog.

Tabelle 5: Lösungsausprägungen

Für die richtige Auswahl werde ich die Lösungsausprägungen anhand der Wunschkriterien in einer Nutzwertanalyse bewerten.

### 1.1.6.2 Wunschkriterien

Ziel	Beschreibung
<b>WZ01</b>	Das System ist mit meinen Kenntnissen programmierbar.
<b>WZ02</b>	Das System sollte innerhalb der vorgegebenen Zeit realisierbar sein.
<b>WZ03</b>	Das System sollte nicht unnötig komplizierter sein als geplant.

Tabelle 6: Wunschkriterien

### 1.1.7 Nutzwertanalyse

#### 1.1.7.1 Präferenzmatrix

Punktevergabe	
<b>0</b>	weniger wichtig
<b>1</b>	gleich wichtig
<b>2</b>	wichtiger

Tabelle 7: Punktevergabe

Präferenzmatrix					
Kriterium	WZ01	WZ02	WZ03	Summe	Faktor %
WZ01		0	1	1	17
WZ02	2		1	3	50
WZ03	1	1		2	33
<b>Summe</b>				<b>6</b>	<b>100</b>

Tabelle 8: Präferenzmatrix

Damit habe ich die Gewichtung der Nutzwertanalyse.

Bewertungsbeschreibung	
<b>1</b>	Erfüllt das Kriterium nicht
<b>2</b>	Erfüllt das Kriterium nur teilweise.
<b>3</b>	Erfüllt das Kriterium vollständig.
<b>4</b>	Übertrifft das Kriterium

Tabelle 9: Bewertungsbeschreibung

### 1.1.7.2 Nutzwertanalyse «Backend»

Kriterium	Gewich-	Java		Python	
		tung	Bewertung	Punkte	Bewertung
WZ01	17	3	51	3	51
WZ02	50	3	150	2	100
WZ03	33	2	66	2	66
<b>Nutzwert</b>	<b>100</b>		<b>267</b>		<b>217</b>

Tabelle 10: Nutzwertanalyse Backend

### 1.1.7.3 Nutzwertanalyse GUI

Kriterium	Gewich-	Webanwendung		JavaFX		JavaFXML	
		tung	Bewertung	Punkte	Bewertung	Punkte	Bewertung
WZ01	17	2	34	1	17	2	34
WZ02	50	1	50	2	100	3	150
WZ03	33	1	33	2	66	2	66
<b>Nutzwert</b>	<b>100</b>		<b>117</b>		<b>183</b>		<b>250</b>

Tabelle 11: Nutzwertanalyse GUI

### 1.1.7.4 Nutzwertanalyse DB

Kriterium	Gewichtung	Selbst eine DB. programmieren		Oracle-DB SQL		MongoDB NoSQL	
		Bewertung	Punkte	Bewertung	Punkte	Bewertung	Punkte
WZ01	17	1	17	2	34	2	34
WZ02	50	1	50	2	100	3	150
WZ03	33	1	33	2	66	3	99
Nutzwert	100		100		200		283

Tabelle 12: Nutzwertanalyse DB

### 1.1.7.5 Lösungsentscheid

Als Lösungsausprägungen wurden

- für das «Backend» die Programmiersprache Java,
- für das GUI JavaFXML,
- für die DB die MongoDB mit NoSQL und

gewählt.

Damit wird auch die Vorgabe der FFHS, Minimum 51% Java-Codeanteil erfüllt.

### 1.1.8 Risikoeinschätzung

Ich habe folgende Risiken identifiziert:

- Nur schulische Erfahrungen in der Durchführung von Projektmanagement (kurz: fehlende Erfahrung PM).
- Nur schulische Erfahrungen in der Programmierung mit Java und JavaFX. (kurz: fehlende Erfahrung in OOP).
- Nur schulische Erfahrungen mit dem Entwickeln von Software. (kurz: fehlende Erfahrung SWE).
- Erfahrungsgemäss schlechtes Zeitmanagement
- Wenig Erfahrung mit DB-Technologien.

ID	Risiko	EW 1-5	TW 1-5	RW 1-25	Auswirkung	
R1	Fehlende Erfahrung PM	4	5	20	Schlechte Planung, Zeitverlust	
R2	Fehlende Erfahrung SWE	5	4	20	Zeitverzögerung, Unsicherheiten	

<b>R3</b>	Fehlende Erfahrung OOP	4	3	12	Zeitverlust durch Schreiben und wieder Löschen von Klassen und Funktionen, Zeitverlust durch Debugging und Fehlersuche,
<b>R4</b>	Wenig Erfahrung mit DB	3	3	9	Unsicherheit im Implementieren und dadurch höherer Zeitaufwand.
<b>R5</b>	Generell schlechtes Zeitmanagement	5	3	15	Unnötiger Zeitverlust.

Tabelle 13: Risikoeinschätzung

EW = Eintrittswahrscheinlichkeit (1 = tiefe EW; 5 = hohe EW)

TW = Tragweite (1 = kleine TW; 5 = grosse TW)

RW = Risikowert

Daraus ergibt sich folgende Risikomatrix:

EW 5			R5	R2	
EW 4				R1	
EW 3			R4		R3
EW 2					
EW 1					
	TW 1	TW 2	TW 3	TW 4	TW 5

Tabelle 14: Risikomatrix

Daher wurden folgende Massnahmen getroffen:

ID	Risiko	Massnahmen
<b>R1</b>	Fehlende Erfahrung PM	Methodisches Vorgehen, in abgeschlossenen Zeitintervallen ungestört arbeiten, sauber Dokumentieren.
<b>R2</b>	Fehlende Erfahrung SWE	Methodisch Vorgehen nach Lehrbuch, gute Planung von PM, KISS-Prinzip
<b>R3</b>	Fehlende Erfahrung OOP	Bekannte Methoden anwenden, Übungen gebrauchen, bei Fragen -> in der Doku nachschlagen, gute Vorarbeit von SWE.
<b>R5</b>	Wenig Erfahrung mit DB	Einfache, bekannte und gut dokumentierte DB benutzen, Lehrvideos gebrauchen, Open Source statt selbst implementieren.
<b>R4</b>	Generell schlechtes Zeitmanagement	Zeitintervalle vorbereiten und planen, keine Ablenkungen (Handy auf Flugmodus), fixe Pausen und Erholungen einplanen.

Tabelle 15: Massnahmentabelle

## 1.1.9 Probleme

Falls während der Projektarbeit Probleme auftauchen, dann werden diese in der nachfolgenden Tabelle festgehalten. Dabei wird das Problem stichwortartig umschrieben, es wird festgehalten wer das Problem betrifft, was für Massnamen getroffen werden und welche Nachkontrolle nötig sein wird.

Problem	Name	Massnahme	Datum	Status
Mehr als eine Woche Terminverzug	Alle	Alle Wunschkriterien werden weggelassen	14.10	erfüllt
Billy tritt aus dem Projekt aus	Simon Wittker und Sabrina Küenzi	Die Stunden werden um (150h – geleistete Stunden von Billy) 131 h gekürzt.	26.10.	erfüllt
Sabrina tritt aus dem Projekt aus	Simon Wittker	Die Stunden werden um (150h – geleistete Stunden von Sabrina) 148 h gekürzt.	04.11.	erfüllt
Viele Lücken im Projekt und Simon arbeitet ab jetzt allein an dem Projekt.	Simon Wittker	Das Projekt wird überarbeitet und das PM erneut durchgeführt.	06.11.	erfüllt

Tabelle 16: Problemtabelle

Ab dem Zeitpunkt, an dem ich allein an diesem Projekt arbeite, führe ich ein Entscheidungs-Log. Dort halte ich die auftretenden Probleme fest die einen Lösungsentscheide brauchen und beschreibe die konkrete Umsetzung. Deshalb wird diese Tabelle dann nicht vorgeführt.

## 1.1.10 Messbare Lieferobjekte

Hier eine Auflistung der wichtigsten Lieferartefakte, die in dieser Arbeit erarbeitet werden:

- Projektauftrag
- Projektstrukturplan
- Netzplan
- Ressourcenplanung
- Statusberichte
- Produktanforderungen (funktionale und nicht-funktionale Anforderungen)
- Technisches Konzept
- Testkonzept
- Quellcode

- Unit-Test
- Testprotokolle
- Java Dokumentation
- Benutzerhandbuch
- Projektdokumentation

## 1.2 Projektplanung

### 1.2.1 Projektstrukturplan

Der nachfolgende Projektstrukturplan zeigt die grobe Struktur und Arbeitspakete unseres Projektes. Wir haben die detaillierte Ansicht der Arbeitspakete aus Gründen der Lesbarkeit und aus Platzgründen weggelassen.

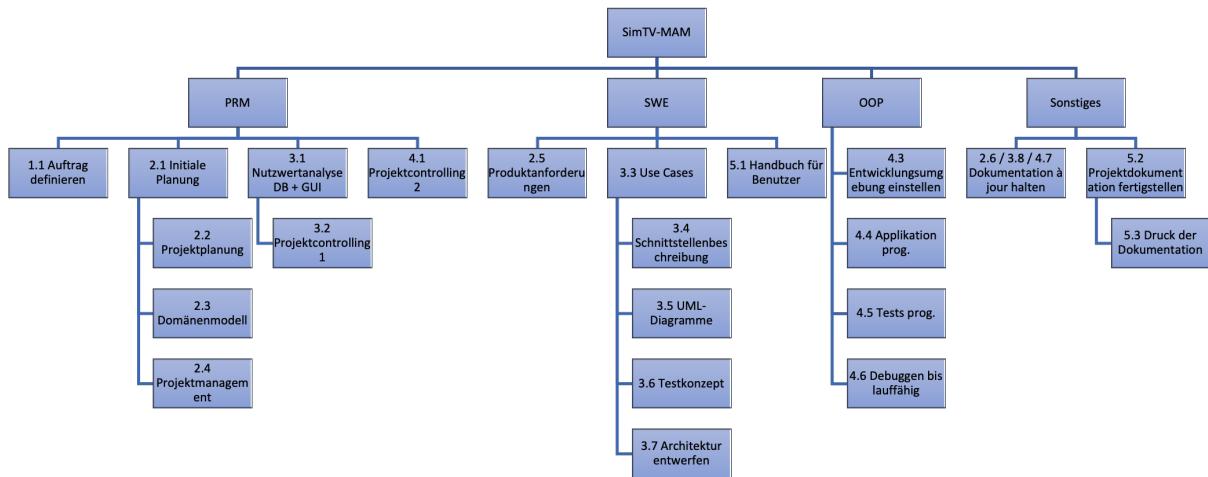


Abbildung 2: Projektstrukturplan

### 1.2.2 Meilensteine

Die nachfolgende Tabelle enthält die Meilensteine. Dabei gibt es keinen Meilenstein am Ende der Beauftragungsphase, weil diese mit dem Schulschluss des 28.08.2021 erfolgte.

ID	Meilensteinbeschreibung	Ende der Phase
M1	Die Projektplanung ist abgeschlossen und die Produktanforderungen fertig formuliert.	Initialisierung
M2	Alle Diagramme und das Testkonzept sind fertig erstellt.	Konzeption
M3	Die programmierte Applikation ist lauffähig.	Realisierung
M4	Der Versand wurde erfolgreich erledigt.	Einführung

Tabelle 17: Meilensteine

### 1.2.3 Netzplan mit Meilensteinen

Tabellenlegende:

AP# = ArbeitspaketNummer  
D = Dauer  
AP = ArbeitspaketName

AP#	D
AP	

Meilenstein:

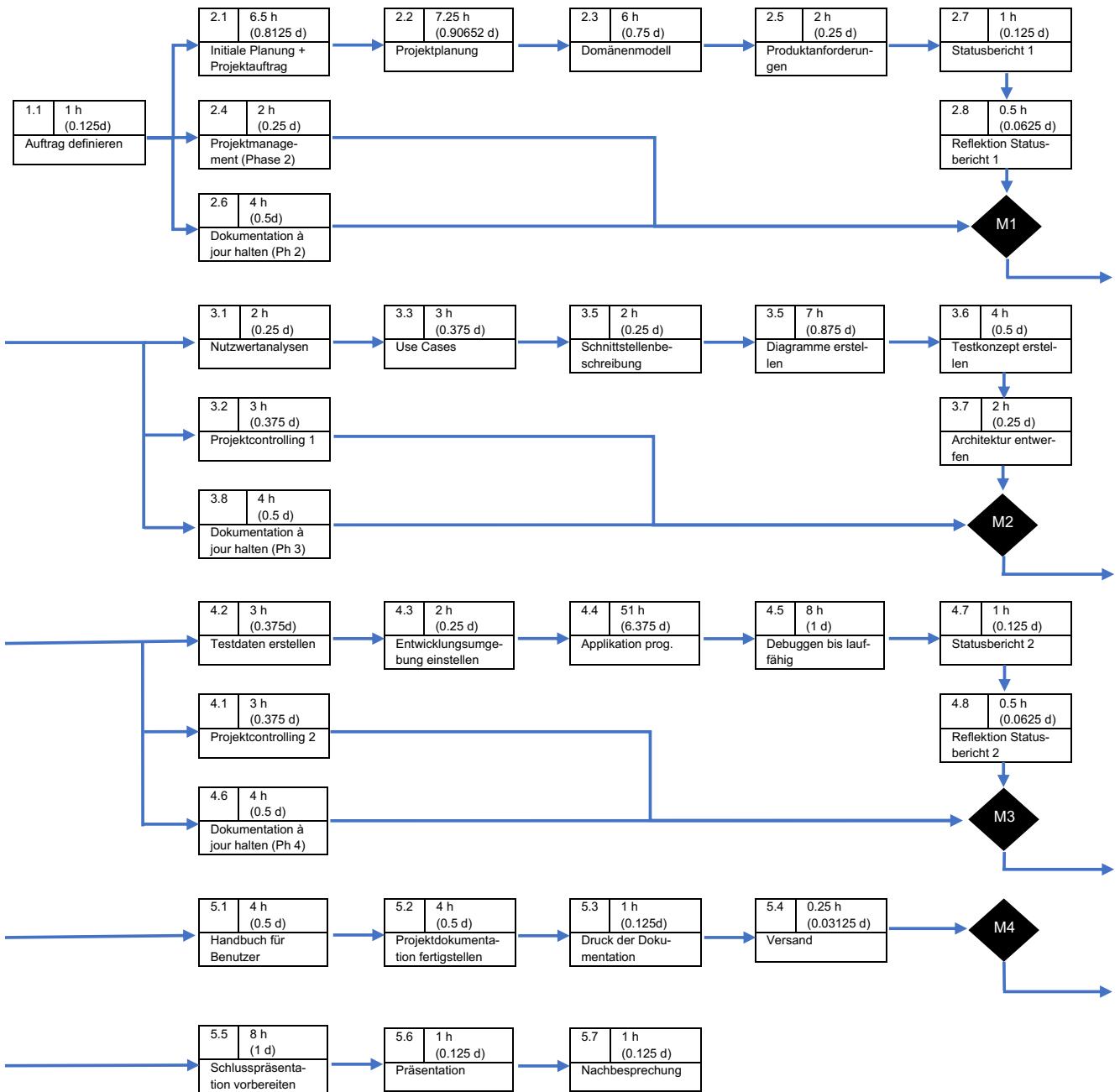


Abbildung 3: Netzplan

### 1.2.4 Terminplanung

In der Tabelle ist der Ressourcenplan mit der Stundenabrechnung und co. zu sehen

	SimTV MAM - Projekt							
Nr.	Arbeitspaket	Fertig	SOLL-h	IST-h	Abweichung	Bemerkungen	Vorbedingung	
<b>1</b>	<b>Beauftragung</b>		3.00	3.00	0.00			
<b>1.1</b>	Auftrag definieren	x	1.00	1.00	0.00	PVA	keine	
<b>1.2</b>	Team kennenlernen	x	1.00	1.00	0.00	1h für SBS-TV / 0h für SimTV	1.1	
<b>1.3</b>	Organisation / Kommunikation	x	1.00	1.00	0.00	1h für SBS-TV / 0h für SimTV	1.2	
<b>2</b>	<b>Initialisierung</b>		29.25	45.25	-16.00			
<b>2.1</b>	<i>Initiale Planung + Projektauftrag</i>		6.50	7.00				
<b>2.1.1</b>	Ist-Analyse + Soll-Konzept	x	1.50	0.50	1.00		1.1	
<b>2.1.2</b>	Projektziele definieren	x	1.50	4.50	-3.00	4h für SBS-TV / 0.5h für SimTV	2.1.1	
<b>2.1.3</b>	Grobanforderungen und Rahmendbedingungen	x	2.50	1.50	1.00	3h für SBS-TV / 0.5h für SimTV	2.1.2	
<b>2.1.4</b>	Risikoeinschätzung	x	1.00	0.50	0.50		2.1.3	
<b>2.2</b>	<i>Projektplanung</i>		7.25	13.75				
<b>2.2.1</b>	Projektstrukturplan	x	2.00	4.25	-2.25	3h für SBS-TV / 1.5h für SimTV	2.1.4	
<b>2.2.2</b>	Arbeitspakete definieren	x	2.00	2.00	0.00		2.2.1	
<b>2.2.3</b>	Netzplan	x	2.00	6.25	-4.25	4h für SBS-TV / 2.25h für SimTV	2.2.2	
<b>2.2.4</b>	Meilensteine planen / definieren	x	0.25	0.25	0.00		2.2.3	
<b>2.2.5</b>	Terminplanung	x	1.00	1.00	0.00		2.2.4	
<b>2.3</b>	Domänenmodell	x	6.00	6.50	-0.50	6h für SBS-TV / 0.5h für SimTV	2.2.5	
<b>2.4</b>	Projektmanagement (Phase 2)	x	2.00	2.50	-0.50		1.1	
<b>2.5</b>	Produktanforderungen	x	2.00	1.50	0.50	1h für SBS-TV / 0.5h für SimTV	2.3	
<b>2.6</b>	Dokumentation à jour halten (Ph 2)	x	4.00	10.50	-6.50	8h für SBS-TV / 2.5h für SimTV	1.1	
<b>2.7</b>	Statusbericht 1	x	1.00	1.00	0.00	1h für SBS-TV / 0h für SimTV	2.5	
<b>2.8</b>	Reflektion Statusbericht 1	x	0.50	0.50	0.00		2.7	
<b>TS</b>	(Teamsitzungen als SBS-TV)		0.00	2.00	-2.00			
<b>M1</b>	<b>Projektplanung abgeschlossen und Produktanforderungen formuliert</b>							

<b>3</b>	<b>Konzeption</b>		27.00	43.00	-16.00		
<b>3.1</b>	Nutzwertanalyse	x	2.00	1.00	1.00		2.8
<b>3.2</b>	Projektcontrolling 1	x	3.00	6.00	-3.00		2.8
<b>3.3</b>	Use Cases	x	3.00	11.75	-8.75	4h + 6h für SBS-TV / 1.5h für SimTV	3.1
<b>3.4</b>	Schnittstellenbeschreibung	x	2.00	1.50	0.50		3.3
<b>3.5</b>	<i>Diagramme erstellen</i>		7.00	9.75			
<b>3.5.1</b>	Kontextabgrenzung	x	1.00	1.00	0.00		3.4
<b>3.5.2</b>	Bausteinsicht + Verteilungssicht	x	2.00	2.00	0.00		3.5.1
<b>3.5.3</b>	Klassendiagramm	x	2.00	3.00	-1.00	1h für SBS-TV / 2h für SimTV	3.5.2
<b>3.5.4</b>	Sequenzdiagramm	x	2.00	3.75	-1.75		3.5.3
<b>3.6</b>	Testkonzept erstellen	x	4.00	2.00	2.00		3.5.4
<b>3.7</b>	Architektur entwerfen	x	2.00	1.25	0.75		3.6
<b>3.8</b>	Dokumentation à jour halten (Ph 3)	x	4.00	3.25	0.75	2h für SBS-TV / 0h für SimTV	2.8
<b>MO</b>	(MockUp's als SBS-TV)		0.00	6.50	-6.50	6h für SBS-TV / 0.5h für SimTV	3.7
<b>M2</b>	<b>Alle Diagramme und das Testkonzept sind fertig erstellt</b>						
<b>4</b>	<b>Realisierung</b>		76.00	96.50	-20.50		
<b>4.1</b>	Projektcontrolling 2	x	3.00	6.50	-3.50		3.7
<b>4.2</b>	Testdaten erstellen	x	3.00	3.00	0.00		3.7
<b>4.3</b>	<i>Entwicklungsumgebung einstellen</i>		2.00	2.00			
<b>4.3.1</b>	Git Repo und Berechtigung	x	0.50	1.00	-0.50		4.2
<b>4.3.2</b>	Code Richtlinien definieren	x	1.50	1.00	0.50		4.3.1
<b>4.4</b>	<i>Applikation prog.</i>		54.50	63.50			
<b>4.4.1</b>	Start- und Metadaten-Klasse prog.	x	5.00	3.50	1.50		4.3.2
<b>4.4.2</b>	GUI prog.	x	9.00	10.00	-1.00		4.4.1
<b>4.4.3</b>	Daten-Eingabefunktion prog.	x	6.00	7.50	-1.50		4.4.2
<b>4.4.4</b>	DB einbinden	x	5.00	8.00	-3.00		4.4.3
<b>4.4.5</b>	Bearbeitungs- und Suchfunktion prog.	x	6.00	8.00	-2.00		4.4.4
<b>4.4.6</b>	Ordner speichern/laden	x	6.00	7.00	-1.00		4.4.5

<b>4.4.7</b>	Code mit Javadoc kommentieren	x	5.50	7.50	-2.00		4.4.6
<b>4.4.8</b>	Tests prog.	x	12.00	12.00	0.00		4.4.7
<b>4.5</b>	Debuggen bis lauffähig	x	8.00	8.00	0.00		4.4.8
<b>4.6</b>	Dokumentation à jour halten (Ph 4)	x	4.00	12.00	-8.00		3.7
<b>4.7</b>	Statusbericht 2	x	1.00	1.00	0.00		4.5
<b>4.8</b>	Reflektion Statusbericht 2	x	0.50	0.50	0.00		4.7
<b>M3</b>	<b>Die programmierte Applikation ist lauffähig</b>						
<b>5</b>	<b>Einführung / Abschluss</b>		19.25	14.00	5.25		
<b>5.1</b>	Handbuch für Benutzer	x	4.00	3.00	1.00		4.8
<b>5.2</b>	<i>Projektdokumentation fertigstellen</i>		4.00	10.00			
<b>5.2.1</b>	Dokumentation vervollständigen	x	2.00	8.00	-6.00		5.1
<b>5.2.2</b>	Evaluierungsbericht verfassen	x	2.00	2.00	0.00		5.2.1
<b>5.3</b>	Druck der Dokumentation	x	1.00	1.00	0.00		5.2.2
<b>5.4</b>	Versandt		0.25		0.25		5.3
<b>5.5</b>	Schlusspräsentation vorbereiten		8.00		8.00		5.4
<b>5.6</b>	Präsentation		1.00		1.00		5.5
<b>5.7</b>	Nachbesprechung		1.00		1.00		5.6
<b>M4</b>	<b>Der Versand wurde erfolgreich erledigt</b>						
	Gesamte Zeit		154.50				
	Verbrauchte Zeit			201.75			
	Verbleibende Zeit				-47.25		

Tabelle 18: Terminplanung mit Reihenfolge der Arbeitspakete

In den zwei Tabellen ist der Terminplan zu sehen. Die Phasen, in denen, in dem Ausschnitt, keine Einträge waren, wurden weggelassen.

SimTV MAM – Projekt										
---------------------	--	--	--	--	--	--	--	--	--	--

Nr.	Wochen- tag:	Sa	Mo	Mo	Mo	Mi	Mo	Mo	Mo	Mo	Mo	Mo	Mo	Sa	Mo	Di	Mi	Do
	Datum:	28.08	30.08	06.09	13.09	15.09	20.09	27.09	04.10	11.10	18.10	25.10	01.11	06.11	08.11	09.11	10.11	11.11
<b>1</b>	Phase 1																	
<b>1.1</b>	Geplante	1   1																
<b>1.2</b>	Effektive	1   1																
<b>1.3</b>		1   1																
<b>2</b>	Phase 2																	
<b>2.1</b>																		
<b>2.1.1</b>		1.5												0.5				
<b>2.1.2</b>		1.5   3												0.5				
<b>2.1.3</b>		2.5   2											2	0.5				
<b>2.1.4</b>		1												0.5				
<b>2.2</b>																		
<b>2.2.1</b>		1	1   1										2	1.5				
<b>2.2.2</b>		2												2				
<b>2.2.3</b>		2   4												2.25				
<b>2.2.4</b>		0.25												0.25				
<b>2.2.5</b>		1												1				
<b>2.3</b>		1   1	5   5											0.5				
<b>2.4</b>		0.5					0.5							0.5	2			
<b>2.5</b>		2		1										0.5	0.5			
<b>2.6</b>		0.5   2	0.75   1	1   3			1.25   2							2	0.5			
<b>2.7</b>				1   1														
<b>2.8</b>				0.5   0.5														
<b>M1</b>					M1									M1				
<b>3</b>	Phase 3																	

3.1							2						1			
3.2							1		1	1			2	2		
3.3							3   4	4				2		1	0.5	
3.4								2					1.5			
3.5													1			
3.5.1								1					1			
3.5.2								2					2			
3.5.3								2	1				2			
3.5.4									2				1			
3.6									4							
3.7										3			1			
3.8							1   1		1   1	1		0.5				
<b>M2</b>							3				<b>M2</b>	0.5				
<b>4</b>	Phase 4															
4.1													1			
4.2									3							
4.3																
4.3.1										0.5		0.5				
4.3.2										1.5						
4.4																
4.4.1										5						
4.4.2											7	2				
4.4.3												3				
4.4.4																
4.4.5																
4.4.6																
4.4.7										0.5		1				
4.4.8																
4.5																

<b>4.6</b>													<b>1</b>	<b>0.5</b>		<b>1</b>			
<b>4.7</b>																<b>1   1</b>			
<b>4.8</b>																<b>0.5</b>			
<b>M3</b>																<b>  0.5</b>			

Da in der Phase 5 in dem Zeitraum keine Einträge sind, lasse ich diesen hier weg.

Tabelle 19: Tabelle 1 des Terminplanes vom 28.08 bis 11.11

SimTV MAM – Projekt																				
Nr.	Wochen-	Mo	Di	Mo	Di	Mo	Di	Mi	Mo	Di	Mo	Di	Mo	Di	Mo	Di	Mi	Do	Mo	Fr
	tag:	Datum:	15.11	16.11	22.11	23.11	29.11	30.11	01.12	06.12	07.12	13.12	14.12	20.12	21.12	22.12	23.12	27.12	14.01	
Da in der Phase 1 und 2 in dem Zeitraum keine Einträge sind, lasse ich diesen hier weg.																				
<b>3</b>	Phase 3																			
<b>3.1</b>	Geplante																			
<b>3.2</b>	Effektive	2																		
<b>3.3</b>		0.25																		
<b>3.4</b>																				
<b>3.5</b>																				
<b>3.5.1</b>																				
<b>3.5.2</b>																				
<b>3.5.3</b>																				
<b>3.5.4</b>		2.75																		
<b>3.6</b>		2																		
<b>3.7</b>		0.25																		
<b>3.8</b>		0.75																		
<b>M2</b>	<b>M2</b>																			
<b>4</b>	Phase 4																			
<b>4.1</b>			0.5	1	0.5	0.5			0.5			1	2	2						
<b>4.2</b>			3																	

Tabelle 20: Tabelle 2 des Terminplanes vom 15.11. bis zum 14.01.

## 1.2.5 Arbeitspakete

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	1.1
<b>Arbeitspaket Name:</b>	Auftrag definieren
<b>Phase:</b>	Beauftragung
<b>Vorbedingung:</b>	keine
<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Einzelheiten über die Aufgabenstellung der Projektarbeit kennenzulernen</li> <li>- Thema festlegen</li> </ul>
<b>Ergebnisse:</b>	Ein Projektthema; Wissen, was zu tun ist.
<b>Arbeitsaufwand:</b>	1 Stunde
<b>Planmässiger Abschluss:</b>	28.08.21
<b>Effektiver Abschluss:</b>	28.08.21

Tabelle 21: Arbeitspaket 1.1

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	2.1
<b>Arbeitspaket Name:</b>	Initiale Planung + Projektauftrag
<b>Phase:</b>	Initialisierung
<b>Vorbedingung:</b>	Arbeitspaket (kurz AP) 1.1
<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Ist-Analyse durchführen + Soll-Konzept erstellen</li> <li>- Projektziele definieren</li> <li>- Grobanforderungen und Rahmenbedingungen festlegen</li> <li>- Risikoeinschätzung durchführen und Massnamen definieren</li> </ul>
<b>Ergebnisse:</b>	Projektauftrag
<b>Arbeitsaufwand:</b>	6.5 Stunden
<b>Planmässiger Abschluss:</b>	30.08.21
<b>Effektiver Abschluss:</b>	06.11.21

Tabelle 22: Arbeitspaket 2.1

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	2.2
<b>Arbeitspaket Name:</b>	Projektplanung
<b>Phase:</b>	Initialisierung
<b>Vorbedingung:</b>	AP 2.1

<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- PSP erstellen</li> <li>- Arbeitspakete definieren</li> <li>- Netzplan erstellen</li> <li>- Meilensteine definieren und planen</li> <li>- Terminplanung durchführen</li> </ul>
<b>Ergebnisse:</b>	Lieferobjekte wie PSP, Netzplan, Terminplanung erstellt.
<b>Arbeitsaufwand:</b>	7.25 Stunden
<b>Planmässiger Abschluss:</b>	06.09.21
<b>Effektiver Abschluss:</b>	08.11.21

Tabelle 23: Arbeitspaket 2.2

<b>Projektname:</b>	<b>SimTV MAM</b>
<b>Arbeitspaket Nummer:</b>	2.3
<b>Arbeitspaket Name:</b>	Domänenmodell
<b>Phase:</b>	Initialisierung
<b>Vorbedingung:</b>	AP 2.2
<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Domänenmodell erstellen</li> </ul>
<b>Ergebnisse:</b>	Domänenmodell
<b>Arbeitsaufwand:</b>	6 Stunden
<b>Planmässiger Abschluss:</b>	13.09.21
<b>Effektiver Abschluss:</b>	08.11.21

Tabelle 24: Arbeitspaket 2.3

<b>Projektname:</b>	<b>SimTV MAM</b>
<b>Arbeitspaket Nummer:</b>	2.4
<b>Arbeitspaket Name:</b>	Projektmanagement
<b>Phase:</b>	Initialisierung
<b>Vorbedingung:</b>	AP 1.1
<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Die effektiv verwendeten Zeiten werden überwacht und Planungen dementsprechend angepasst.</li> <li>- Jede Arbeitseinheit wird vorbereitet und die Arbeiten geplant.</li> </ul>
<b>Ergebnisse:</b>	Weniger Zeitverlust durch effizientere Ausnutzung der Projektzeit in der Phase «Initialisierung»
<b>Arbeitsaufwand:</b>	2 Stunde
<b>Planmässiger Abschluss:</b>	20.09.21
<b>Effektiver Abschluss:</b>	8.11.21

Tabelle 25: Arbeitspaket 2.4

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	2.5
<b>Arbeitspaket Name:</b>	Produktanforderungen
<b>Phase:</b>	Initialisierung
<b>Vorbedingung:</b>	AP 2.3
<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Die funktionalen und nicht-funktionalen Anforderungen werden definiert</li> </ul>
<b>Ergebnisse:</b>	Funktionale und nicht-funktionale Anforderungen
<b>Arbeitsaufwand:</b>	2 Stunden
<b>Planmässiger Abschluss:</b>	13.09.21
<b>Effektiver Abschluss:</b>	08.11.21

Tabelle 26: Arbeitspaket 2.5

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	2.6
<b>Arbeitspaket Name:</b>	Dokumentation à jour halten (Ph 2)
<b>Phase:</b>	Initialisierung
<b>Vorbedingung:</b>	AP 1.1
<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Die Dokumentation wird laufend aktualisiert, in dem die Lieferobjekte hinzugefügt und beschrieben werden.</li> </ul>
<b>Ergebnisse:</b>	Am Ende der Phase «Initialisierung» ist die Dokumentation aktuell.
<b>Arbeitsaufwand:</b>	4 Stunden
<b>Planmässiger Abschluss:</b>	20.09.21
<b>Effektiver Abschluss:</b>	08.11.21

Tabelle 27: Arbeitspaket 2.6

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	2.7
<b>Arbeitspaket Name:</b>	Statusbericht 1
<b>Phase:</b>	Initialisierung
<b>Vorbedingung:</b>	AP 2.5
<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Ein Statusbericht für die 1. Statussitzung wird verfasst</li> </ul>
<b>Ergebnisse:</b>	Der Statusbericht 1
<b>Arbeitsaufwand:</b>	1 Stunde
<b>Planmässiger Abschluss:</b>	15.09.21
<b>Effektiver Abschluss:</b>	15.09.21

Tabelle 28: Arbeitspaket 2.7

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	2.8
<b>Arbeitspaket Name:</b>	Reflektion Statusbericht 1
<b>Phase:</b>	Initialisierung
<b>Vorbedingung:</b>	AP 2.7
<b>Leistungsumfang:</b>	- Protokoll des ersten Statusmeeting wird verfasst.
<b>Ergebnisse:</b>	Protokoll des ersten Statusmeeting
<b>Arbeitsaufwand:</b>	0.5 Stunden
<b>Planmässiger Abschluss:</b>	15.09.21
<b>Effektiver Abschluss:</b>	15.09.21

Tabelle 29: Arbeitspaket 2.8

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	3.1
<b>Arbeitspaket Name:</b>	Nutzwertanalysen
<b>Phase:</b>	Konzeption
<b>Vorbedingung:</b>	AP 2.8
<b>Leistungsumfang:</b>	- Vier Nutzwertanalysen werden durchgeführt, je eine für das «Backend», für das GUI, für die DB und für den Suchalgorithmus.
<b>Ergebnisse:</b>	Lösungskonzept
<b>Arbeitsaufwand:</b>	2 Stunden
<b>Planmässiger Abschluss:</b>	27.09.21
<b>Effektiver Abschluss:</b>	08.11.21

Tabelle 30: Arbeitspaket 3.1

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	3.2
<b>Arbeitspaket Name:</b>	Projektcontrolling 1
<b>Phase:</b>	Konzeption
<b>Vorbedingung:</b>	AP 2.8
<b>Leistungsumfang:</b>	- Die effektiv verwendeten Zeiten werden überwacht und Planungen dementsprechend angepasst. - Jede Arbeitseinheit wird vorbereitet und die Arbeiten geplant.
<b>Ergebnisse:</b>	Weniger Zeitverlust durch effizientere Ausnutzung der Projektzeit in der Phase «Konzeption».

<b>Arbeitsaufwand:</b>	3 Stunden
<b>Planmässiger Abschluss:</b>	18.10.21
<b>Effektiver Abschluss:</b>	15.11.21

Tabelle 31: Arbeitspaket 3.2

<b>Projektname:</b>	<b>SimTV MAM</b>
<b>Arbeitspaket Nummer:</b>	3.3
<b>Arbeitspaket Name:</b>	UseCases
<b>Phase:</b>	Konzeption
<b>Vorbedingung:</b>	AP 3.1
<b>Leistungsumfang:</b>	- Für die Applikation werden die UseCase definiert und mittels eines UseCase-Diagramms dargestellt.
<b>Ergebnisse:</b>	UseCase-Diagramm mit Beschreibung
<b>Arbeitsaufwand:</b>	3 Stunden
<b>Planmässiger Abschluss:</b>	27.10.21
<b>Effektiver Abschluss:</b>	15.11.21

Tabelle 32: Arbeitspaket 3.3

<b>Projektname:</b>	<b>SimTV MAM</b>
<b>Arbeitspaket Nummer:</b>	3.4
<b>Arbeitspaket Name:</b>	Schnittstellenbeschreibung
<b>Phase:</b>	Konzeption
<b>Vorbedingung:</b>	AP 3.3
<b>Leistungsumfang:</b>	- Die Schnittstellen werden definiert und in einem Schema dargestellt und beschrieben
<b>Ergebnisse:</b>	Schnittstellenschema mit Beschreibung
<b>Arbeitsaufwand:</b>	2 Stunden
<b>Planmässiger Abschluss:</b>	04.10.21
<b>Effektiver Abschluss:</b>	09.11.21

Tabelle 33: Arbeitspaket 3.4

<b>Projektname:</b>	<b>SimTV MAM</b>
<b>Arbeitspaket Nummer:</b>	3.5
<b>Arbeitspaket Name:</b>	Diagramme erstellen
<b>Phase:</b>	Konzeption
<b>Vorbedingung:</b>	AP 3.4

<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Eine Kontextabgrenzung wird erstellt.</li> <li>- Eine Bausteine- und eine Verteilungssicht wird erstellt.</li> <li>- Ein Klassendiagramm wird erstellt.</li> <li>- Ein Aktivitätsdiagramm wird erstellt.</li> </ul>
<b>Ergebnisse:</b>	Statische und dynamische Diagramme
<b>Arbeitsaufwand:</b>	7 Stunden
<b>Planmässiger Abschluss:</b>	11.10.21
<b>Effektiver Abschluss:</b>	15.11.21

Tabelle 34: Arbeitspaket 3.5

<b>Projektname:</b>	<b>SimTV MAM</b>
<b>Arbeitspaket Nummer:</b>	3.6
<b>Arbeitspaket Name:</b>	Testkonzept erstellen
<b>Phase:</b>	Konzeption
<b>Vorbedingung:</b>	AP 3.5
<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Ein Testkonzept für die Applikation erstellen</li> </ul>
<b>Ergebnisse:</b>	Testkonzept
<b>Arbeitsaufwand:</b>	4 Stunden
<b>Planmässiger Abschluss:</b>	11.10.21
<b>Effektiver Abschluss:</b>	15.11.21

Tabelle 35: Arbeitspaket 3.6

<b>Projektname:</b>	<b>SimTV MAM</b>
<b>Arbeitspaket Nummer:</b>	3.7
<b>Arbeitspaket Name:</b>	Architektur entwerfen
<b>Phase:</b>	Konzeption
<b>Vorbedingung:</b>	AP 3.6
<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Die Architektur der Applikation entwerfen</li> </ul>
<b>Ergebnisse:</b>	Architektur-Entwurf
<b>Arbeitsaufwand:</b>	2 Stunden
<b>Planmässiger Abschluss:</b>	18.10.21
<b>Effektiver Abschluss:</b>	15.11.21

Tabelle 36: Arbeitspaket 3.7

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	3.8
<b>Arbeitspaket Name:</b>	Dokumentation à jour halten (Ph 3)
<b>Phase:</b>	Konzeption
<b>Vorbedingung:</b>	AP 2.8
<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Die Dokumentation wird laufend aktualisiert, in dem die Lieferobjekte hinzugefügt und beschrieben werden.</li> </ul>
<b>Ergebnisse:</b>	Am Ende der Phase «Konzeption» ist die Dokumentation aktuell.
<b>Arbeitsaufwand:</b>	4 Stunden
<b>Planmässiger Abschluss:</b>	18.10.21
<b>Effektiver Abschluss:</b>	15.11.21

Tabelle 37: Arbeitspaket 3.8

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	4.1
<b>Arbeitspaket Name:</b>	Projektcontrolling 2
<b>Phase:</b>	Realisierung
<b>Vorbedingung:</b>	AP 3.7
<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Die effektiv verwendeten Zeiten werden überwacht und Planungen dementsprechend angepasst.</li> <li>- Jede Arbeitseinheit wird vorbereitet und die Arbeiten geplant.</li> <li>- Überprüfen der Doku mit der Arbeitsmappe (Was fehlt noch?).</li> </ul>
<b>Ergebnisse:</b>	Weniger Zeitverlust durch effizientere Ausnutzung der Projektzeit in der Phase «Realisierung».
<b>Arbeitsaufwand:</b>	3 Stunden
<b>Planmässiger Abschluss:</b>	30.11.21
<b>Effektiver Abschluss:</b>	20.12.21

Tabelle 38: Arbeitspaket 4.1

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	4.2
<b>Arbeitspaket Name:</b>	Testdaten erstellen
<b>Phase:</b>	Realisierung
<b>Vorbedingung:</b>	AP 3.7
<b>Leistungsumfang:</b>	Die Test-Metadaten werden erstellt
<b>Ergebnisse:</b>	Testdaten

<b>Arbeitsaufwand:</b>	2 Stunden
<b>Planmässiger Abschluss:</b>	18.10.21
<b>Effektiver Abschluss:</b>	16.11.21

Tabelle 39: Arbeitspaket 4.2

<b>Projektname:</b>	<b>SimTV MAM</b>
<b>Arbeitspaket Nummer:</b>	4.3
<b>Arbeitspaket Name:</b>	Entwicklungsumgebung einstellen
<b>Phase:</b>	Realisierung
<b>Vorbedingung:</b>	AP 4.2
<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Git Repository erstellen und mit der IDE verbinden</li> <li>- Code-Richtlinien definieren</li> </ul>
<b>Ergebnisse:</b>	Git Repository; Code-Richtlinien
<b>Arbeitsaufwand:</b>	2 Stunden
<b>Planmässiger Abschluss:</b>	25.10.21
<b>Effektiver Abschluss:</b>	16.11.21

Tabelle 40: Arbeitspaket 4.3

<b>Projektname:</b>	<b>SimTV MAM</b>
<b>Arbeitspaket Nummer:</b>	4.4
<b>Arbeitspaket Name:</b>	Applikation prog.
<b>Phase:</b>	Realisierung
<b>Vorbedingung:</b>	AP 4.3
<b>Leistungsumfang:</b>	<ul style="list-style-type: none"> <li>- Die Start- und Metadaten-Klasse programmieren</li> <li>- Das GUI programmieren</li> <li>- Die Daten-Eingabefunktion programmieren</li> <li>- Die DB einbinden</li> <li>- Die Bearbeitungs- und Suchfunktion programmieren</li> <li>- Ordner speichern und laden</li> <li>- Die Tests programmieren</li> </ul>
<b>Ergebnisse:</b>	Eine programmierte Applikation
<b>Arbeitsaufwand:</b>	51 Stunden
<b>Planmässiger Abschluss:</b>	29.11.21
<b>Effektiver Abschluss:</b>	14.12.21

Tabelle 41: Arbeitspaket 4.4

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	4.5
<b>Arbeitspaket Name:</b>	Debuggen bis lauffähig
<b>Phase:</b>	Realisierung
<b>Vorbedingung:</b>	AP 4.4
<b>Leistungsumfang:</b>	- Den programmierten Code debuggen, damit er lauffähig wird.
<b>Ergebnisse:</b>	Lauffähige Applikation
<b>Arbeitsaufwand:</b>	8 Stunden
<b>Planmässiger Abschluss:</b>	30.11.21
<b>Effektiver Abschluss:</b>	14.12.21

Tabelle 42: Arbeitspaket 4.5

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	4.6
<b>Arbeitspaket Name:</b>	Dokumentation à jour halten (Ph 4)
<b>Phase:</b>	Realisierung
<b>Vorbedingung:</b>	AP 3.7
<b>Leistungsumfang:</b>	- Die Dokumentation wird laufend aktualisiert, in dem die Lieferobjekte hinzugefügt und beschrieben werden.
<b>Ergebnisse:</b>	Am Ende der Phase «Realisierung» ist die Dokumentation aktuell.
<b>Arbeitsaufwand:</b>	4 Stunden
<b>Planmässiger Abschluss:</b>	06.12.21
<b>Effektiver Abschluss:</b>	21.12.21

Tabelle 43: Arbeitspaket 4.6

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	4.7
<b>Arbeitspaket Name:</b>	Statusbericht 2
<b>Phase:</b>	Realisierung
<b>Vorbedingung:</b>	AP 4.5
<b>Leistungsumfang:</b>	Ein Statusbericht für die 2. Statussitzung wird verfasst
<b>Ergebnisse:</b>	Statusbericht 2
<b>Arbeitsaufwand:</b>	1 Stunde
<b>Planmässiger Abschluss:</b>	10.11.21
<b>Effektiver Abschluss:</b>	10.11.21

Tabelle 44: Arbeitspaket 4.7

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	4.8
<b>Arbeitspaket Name:</b>	Reflektion Statusbericht 2
<b>Phase:</b>	Realisierung
<b>Vorbedingung:</b>	AP 4.7
<b>Leistungsumfang:</b>	Ein Reflektionsprotokoll des 2. Statusmeetings verfassen
<b>Ergebnisse:</b>	Reflektionsprotokoll Statusmeeting 2
<b>Arbeitsaufwand:</b>	0.5 Stunden
<b>Planmässiger Abschluss:</b>	10.11.21
<b>Effektiver Abschluss:</b>	10.11.21

Tabelle 45: Arbeitspaket 4.8

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	5.1
<b>Arbeitspaket Name:</b>	Handbuch für Benutzer
<b>Phase:</b>	Einführung / Abschluss
<b>Vorbedingung:</b>	AP 4.8
<b>Leistungsumfang:</b>	Das Handbuch für die Benutzung der Applikation verfassen.
<b>Ergebnisse:</b>	Benutzerhandbuch
<b>Arbeitsaufwand:</b>	4 Stunden
<b>Planmässiger Abschluss:</b>	06.12.21
<b>Effektiver Abschluss:</b>	21.12.21

Tabelle 46: Arbeitspaket 5.1

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	5.2
<b>Arbeitspaket Name:</b>	Projektdokumentation fertigstellen
<b>Phase:</b>	Einführung / Abschluss
<b>Vorbedingung:</b>	AP 5.1
<b>Leistungsumfang:</b>	Die Projektdokumentation wird fertiggestellt, dass sie zum Druck kann. Dazu gehört auch der Evaluationsbericht anfertigen und einfügen.
<b>Ergebnisse:</b>	Fertige Projektdokumentation
<b>Arbeitsaufwand:</b>	4 Stunden
<b>Planmässiger Abschluss:</b>	07.12.21
<b>Effektiver Abschluss:</b>	23.12.21

Tabelle 47: Arbeitspaket 5.2

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	5.3
<b>Arbeitspaket Name:</b>	Druck der Dokumentation
<b>Phase:</b>	Einführung / Abschluss
<b>Vorbedingung:</b>	AP 5.2
<b>Leistungsumfang:</b>	Die Projektdokumentation wird gedruckt, gebunden und versandfertig gemacht.
<b>Ergebnisse:</b>	Versandfertige gebundene Projektdokumentation
<b>Arbeitsaufwand:</b>	1 Stunde
<b>Planmässiger Abschluss:</b>	13.12.21
<b>Effektiver Abschluss:</b>	23.12.21

Tabelle 48: Arbeitspaket 5.3

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	5.4
<b>Arbeitspaket Name:</b>	Versandt
<b>Phase:</b>	Einführung / Abschluss
<b>Vorbedingung:</b>	AP 5.3
<b>Leistungsumfang:</b>	Die Projektdokumentation und eventuell ein Datenträger wird per Post zu den Dozenten versendet.
<b>Ergebnisse:</b>	Projekt versandt
<b>Arbeitsaufwand:</b>	0.25
<b>Planmässiger Abschluss:</b>	Spätestens 23.12.21
<b>Effektiver Abschluss:</b>	

Tabelle 49: Arbeitspaket 5.4

<b>Projektname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	5.5
<b>Arbeitspaket Name:</b>	Schlusspräsentation vorbereiten
<b>Phase:</b>	Einführung / Abschluss
<b>Vorbedingung:</b>	AP 5.4
<b>Leistungsumfang:</b>	Die Präsentation, die am Schluss der Projektarbeit stattfindet, wird vorbereitet.
<b>Ergebnisse:</b>	Gut vorberichtet für die Präsentation

<b>Arbeitsaufwand:</b>	8 Stunden
<b>Planmässiger Abschluss:</b>	27.12.21
<b>Effektiver Abschluss:</b>	

Tabelle 50: Arbeitspaket 5.5

<b>Projektnname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	5.6
<b>Arbeitspaket Name:</b>	Präsentation
<b>Phase:</b>	Einführung / Abschluss
<b>Vorbedingung:</b>	AP 5.5
<b>Leistungsumfang:</b>	Die Präsentation wird gehalten und Expertenfragen beantwortet.
<b>Ergebnisse:</b>	Präsentation gehalten
<b>Arbeitsaufwand:</b>	1 Stunde
<b>Planmässiger Abschluss:</b>	14.01.22
<b>Effektiver Abschluss:</b>	

Tabelle 51: Arbeitspaket 5.6

<b>Projektnname:</b>	SimTV MAM
<b>Arbeitspaket Nummer:</b>	5.7
<b>Arbeitspaket Name:</b>	Nachbesprechung
<b>Phase:</b>	Einführung / Abschluss
<b>Vorbedingung:</b>	AP 5.6
<b>Leistungsumfang:</b>	Die gesamte Projektarbeit mit der Projektdokumentation, der Applikation und der Präsentation wird besprochen, bewertet und eventuell benotet.
<b>Ergebnisse:</b>	Abschluss der Projektarbeit (und hoffentlich eine gute Note:)
<b>Arbeitsaufwand:</b>	1 Stunde
<b>Planmässiger Abschluss:</b>	14.01.22
<b>Effektiver Abschluss:</b>	

Tabelle 52: Arbeitspaket 5.7

## 2 Software-Engineering

Dieses Kapitel enthält den Teil der Softwareentwicklung für diese Projektarbeit. Grob aufgeteilt in die Analysedokumente für die Vorbereitung als Artefakt des Requirement Engineering, die Softwareentwurfsdokumente für den Hauptteil und die Inbetriebnahme für den Abschluss.

### 2.1 Analysedokumente

#### 2.1.1 Stakeholder

Die massgebenden Stakeholder in der nachfolgenden Liste:

ID	Name	Rolle im Studium	Rolle im Projekt
1	Urs-Martin Künzi Roman Fischer Alexander Winiger	Dozierende, Ausschuss	<ul style="list-style-type: none"> <li>- Kontrollboard</li> <li>- Verabschiedung der Meilensteine und Projektzielen</li> <li>- Abnahme des Endprodukts</li> </ul>
2	Simon Wittker	Student	<ul style="list-style-type: none"> <li>- Requirement Engineering</li> <li>- Software-Entwickler</li> <li>- Programmierer</li> <li>- Projektleiter</li> <li>- Tester</li> </ul>
3	SimTV	keine	Kundenfirma

Tabelle 53: Stakeholder-Tabelle

#### 2.1.2 Systemidee

Das System soll als Media Asset Management-System realisiert werden. Der Begriff «Media Asset» ist aber nicht ganz passen, da mit dem System die Metadaten verwaltet werden und nicht die ganzen Media Assets. Der User verwendet dieses Programm, um die Metadaten seiner Media Assets zu verwalten. Dabei kann der User neue Metadaten hinzufügen und bereits gespeicherte über die Suchfunktion suchen. Die Suchfunktion liefert spätestens nach 3 Sekunden ein Ergebnis, oder terminiert nach spätestens 5 Sekunden. Die Suchergebnisse können in einer Detailansicht eingesehen, verändert, wieder gespeichert und gelöscht werden. Das System soll so gebaut werden, dass es zu einem späteren Zeitpunkt erweitert werden kann.

Das System verfügt über keine Verbindung mit dem Internet, ausser dem Zugriff auf die DB. Somit ist der Datenaustausch mit anderen Geräten zurzeit nicht möglich.

## 2.1.3 Produktanforderungen

### 2.1.3.1 Funktionale Anforderungen

ID	Beschreibung	Muss/Kann
FA01	Das System muss eine Benutzeroberfläche (GUI) bereitstellen.	M
FA02	Das GUI kann (nur) mit der Maus bedienbar sein.	K
FA03	Das System muss neue Metadaten mithilfe einer Eingabemaske hinzufügen können.	M
FA04	Das System muss die Metadaten bearbeiten und löschen können.	M
FA05	Das System muss die Metadaten in die DB speichern können.	M
FA06	Das System muss über eine Suchfunktion verfügen, die die DB durchsuchen kann.	M
FA07	Das System muss die Suchresultate auf der GUI anzeigen können.	M
FA08	Das System muss die Suchresultate in einer Detailansicht anzeigen können.	M
FA09	Von der Detailansicht sollen sich die Daten in einem Editor öffnen lassen können.	M
FA10	In dem Editor müssen sich das abgespeicherte Metadatenobjekt bearbeiten lassen können.	M
FA11	Im Editor müssen sich die Metadaten abspeichern lassen können.	M
FA12	In der Detailansicht oder im Editor muss das Metadatenobjekt aus der DB gelöscht werden können.	M
FA13	Das System muss erweiterbar sein.	K
FA14	Das System kann eine erweiterte Suche durchführen. Damit soll gezielt nach einer oder mehreren Eigenschaften der Metadaten gesucht werden können	K
FA15	Das System kann über eine beliebig erweiterbare Eingabemaske verfügen.	K
FA16	Das System kann mit einem Passwort gesichert werden.	K

Tabelle 54: Funktionale Anforderungen

### 2.1.3.2 Nicht-Funktionale Anforderungen

ID	Beschreibung	Muss/Kann
NFA01	Das GUI muss leicht und intuitiv bedienbar sein.	M
NFA02	Die Suchfunktion muss ein Suchergebnis in unter 3 Sekunden anzeigen.	M
NFA03	Das System soll eine informative Fehlermeldung bei nicht-korrektem Speichern anzeigen.	M
NFA04	Die Suchfunktion muss innerhalb von 5 Sekunden terminieren.	M

NFA05	Die Suchresultate sollen übersichtlich dargestellt werden	K
NFA02	Die Suchresultate sollen nach auswählbaren Kriterien sortiert werden können.	K

Tabelle 55: Nicht-Funktionale Anforderungen

## 2.1.4 Qualitätszieleinschätzung

	Sehr wichtig	wichtig	weniger wichtig	unwichtig
Funktionalität		x		
Zuverlässigkeit		x		
Benutzbarkeit		x		
Effizienz			x	
Änderbarkeit	x			
Übertragbarkeit		x		

Tabelle 56: Qualitätsanforderungen

## 2.1.5 Domänenmodell

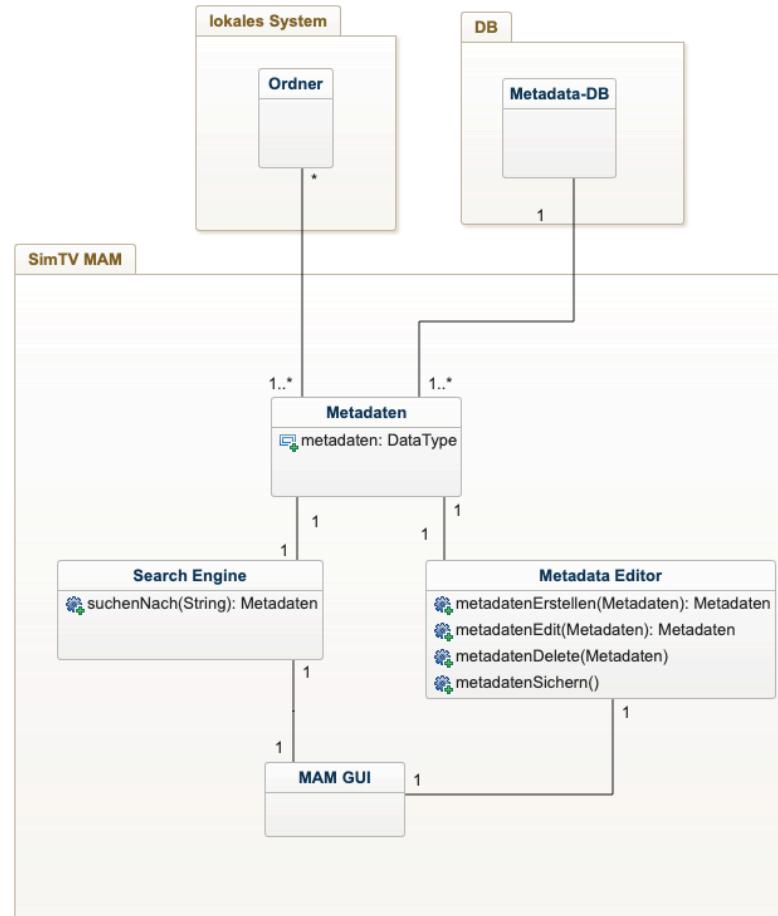


Abbildung 4: Domänenmodell

## 2.1.6 UseCase

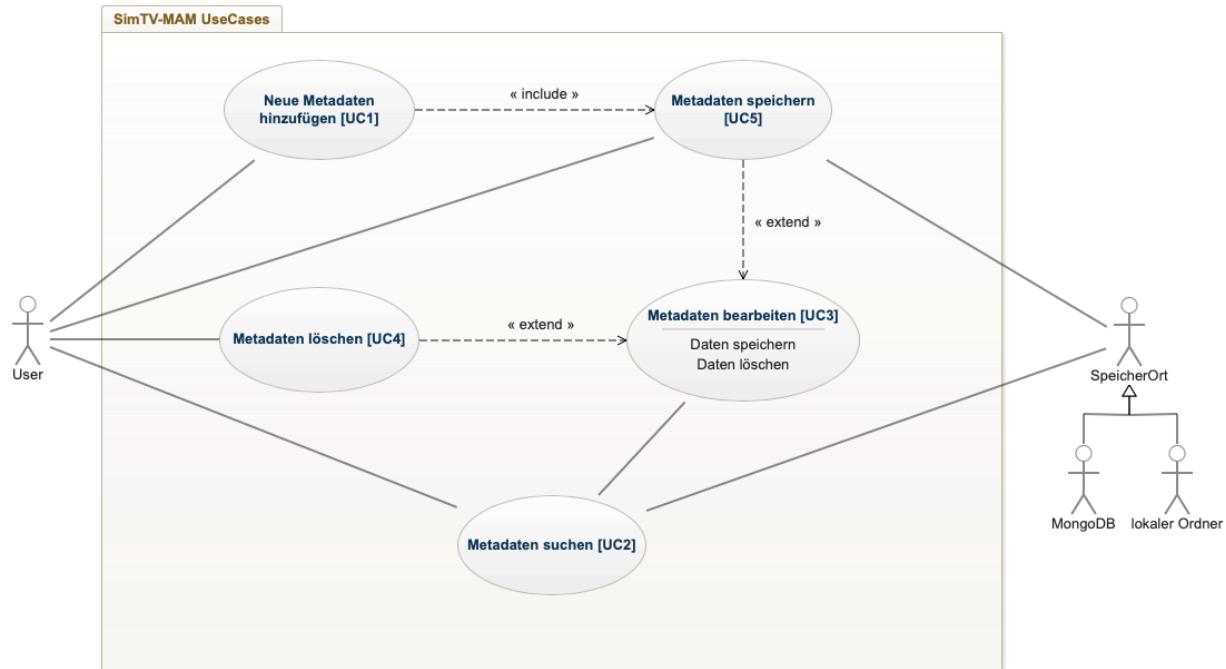


Abbildung 5: UseCase-Diagramm

### 2.1.6.1 Neue Metadaten hinzufügen

Abschnitt	Inhalt
ID:	UC1
Name:	Neue Metadaten hinzufügen
Version:	1.4
Datum:	15.11.2021
Kurzbeschreibung:	Der User fügt neue Metadaten hinzu, indem er die vorgefertigte Maske ausfüllt und dann auf «Daten speichern» klickt.
Vorbedingung:	Es wurde «Neue Metadaten hinzufügen» gedrückt.
Nachbedingung:	Eine Eingabemaske erscheint, in die die Daten der Metadaten eingegeben werden.
Fehlersituation:	Die Seite mit der Eingabemaske kann nicht geladen werden.
Systemzustand im Fehlerfall:	Durch eine Fehlermeldung wird der User über den Fehler beim Laden informiert.
Akteure:	User
Trigger:	Der User will neue Metadaten hinzufügen.
Standardablauf:	<ol style="list-style-type: none"> <li>Der User klickt auf «Neue Metadaten hinzufügen».</li> <li>Ein neues Fenster geht auf mit einer Eingabemaske.</li> </ol>

	<p>3. Der User schreibt die Metadaten in die Eingabemaske ein.</p> <p>4. Der User klickt auf «Daten speichern» -&gt; UC5.</p>
<b>Alternativabläufe:</b>	<p>(1-3) wie oben.</p> <p>4. Der User klickt auf «zurück» oder «Abbrechen».</p> <p>5. Ein Mitteilungsfenster erscheint und informiert den User, dass bei einem vorzeitigen Verlassen die Daten nicht gespeichert werden.</p> <p>6. Bei «OK» wird die Seite verlassen und der Home-Screen wird angezeigt.</p> <p>7. Bei «Abbrechen» bleibt der User auf der Seite, mit den eingegebenen Daten.</p>

Tabelle 57: UseCase 1

### 2.1.6.2 Metadaten suchen

Abschnitt	Inhalt
<b>ID:</b>	UC2
<b>Name:</b>	Metadaten suchen
<b>Version:</b>	1.4
<b>Datum:</b>	15.11.2021
<b>Kurzbeschreibung:</b>	Der User sucht mithilfe von eines Suchbegriffes nach bestehenden Metadaten.
<b>Vorbedingung:</b>	Es wurde ein Suchbegriff eingegeben und auf «Suche starten» geklickt.
<b>Nachbedingung:</b>	Die Suchresultate erscheinen in einem neuen Fenster.
<b>Fehlersituation:</b>	Es gab einen Fehler beim Suchvorgang.
<b>Systemzustand im Fehlerfall:</b>	Eine Meldung informiert den User über den Fehler.
<b>Akteure:</b>	User
<b>Trigger:</b>	Der User will nach bestimmten Metadaten suchen.
<b>Standardablauf:</b>	<ol style="list-style-type: none"> <li>Der User gibt einen Suchbegriff ein und klickt auf «Suche starten».</li> <li>Der User startet die Suche mit einem Klick auf «Suche starten».</li> <li>Das Fenster wird gewechselt.</li> <li>Die Suchresultate erscheinen.</li> </ol>

	<p>5. Der User kann ausgewählte Suchresultate bearbeiten, indem er sie anklickt und auf «Metadaten bearbeiten» klickt (siehe UC3).</p> <p>6. Der User kann durch Betätigen von «Zurück» die Suchfunktion verlassen.</p>
<b>Alternativabläufe:</b>	<p>(1-3) wie oben.</p> <p>4. Es gibt keine Suchresultate.</p> <p>5. Es wird angezeigt, dass keine Suchtreffer gefunden wurden.</p> <p>6. Der User kann eine neue Suche starten (UC2) oder durch Betätigen von «zurück» das Suchresultatfenster verlassen.</p>

Tabelle 58: UseCase 2

### 2.1.6.3 Metadaten bearbeiten

Abschnitt	Inhalt
<b>ID:</b>	UC3
<b>Name:</b>	Metadaten bearbeiten
<b>Version:</b>	1.4
<b>Datum:</b>	15.11.2021
<b>Kurzbeschreibung:</b>	Der User bearbeitet bestehende Metadaten, indem er ein Suchresultat angewählt hat, auf «ausgewählte Metadaten bearbeiten» geklickt hat. Die Daten, die in die Datenmaske geladen wurden, bearbeitet hat und dann auf «Daten speichern» klickt.
<b>Vorbedingung:</b>	Eine Suche wurde getätig (UC2), ein Suchergebnis wurde angewählt und auf «ausgewählte Metadaten bearbeiten» geklickt.
<b>Nachbedingung:</b>	Die Daten in der Maske werden gespeichert -> siehe UC5
<b>Fehlersituation:</b>	Es wurde auf «Daten speichern» geklickt und entweder ein ungültiges Zeichen steht in einem Feld oder/und ein zwingend auszufüllendes Feld ist.
<b>Systemzustand im Fehlerfall:</b>	Durch eine Fehlermeldung wird der User über die fehlerhafte Eingabe oder über das leere Feld informiert.
<b>Akteure:</b>	User
<b>Trigger:</b>	Der User will bestehende Metadaten bearbeiten.
<b>Standardablauf:</b>	<ol style="list-style-type: none"> <li>1. Die Seite mit der Datenmaske erscheint mit den Daten des zu bearbeitenden Metadaten-Satzes.</li> <li>2. Der User bearbeitet die Daten in die Datenmaske.</li> <li>3. Der User klickt auf «Metadaten speichern».</li> </ol>

<b>Alternativabläufe:</b>	<p>(1-2) wie oben.</p> <ol style="list-style-type: none"> <li>3. Der User klickt auf «zurück» oder «Abbrechen».</li> <li>4. Ein Mitteilungsfenster erscheint und informiert den User, dass bei einem vorzeitigen Verlassen die Daten nicht gespeichert werden.</li> <li>5. Bei «OK» wird die Seite verlassen und die Seite mit den Suchresultaten wird angezeigt.</li> </ol>
<b>Alternativabläufe:</b>	<p>(1) wie oben.</p> <ol style="list-style-type: none"> <li>2. Der User klickt auf «Daten löschen» (UC4).</li> </ol>

Tabelle 59: UseCase3

#### 2.1.6.4 Daten löschen

Abschnitt	Inhalt
<b>ID:</b>	UC4
<b>Name:</b>	Metadaten löschen
<b>Version:</b>	1.4
<b>Datum:</b>	15.11.2021
<b>Kurzbeschreibung:</b>	Der User löscht bestehender Metadaten-Satz.
<b>Vorbedingung:</b>	Eine Suche wurde getätigt (UC2), ein Suchergebnis wurde angewählt und auf «ausgewählte Metadaten bearbeiten» geklickt.
<b>Nachbedingung:</b>	Die Metadaten sind gelöscht.
<b>Fehlersituation:</b>	Beim Löschen ist ein Fehler aufgetreten und die Metadaten konnte nicht gelöscht werden.
<b>Systemzustand im Fehlerfall:</b>	Eine Meldung informiert den User darüber, dass die Metadaten nicht gelöscht werden konnten.
<b>Akteure:</b>	User
<b>Trigger:</b>	Der User will die Metadaten unwiderruflich löschen.
<b>Standardablauf:</b>	<ol style="list-style-type: none"> <li>1. Der User klickt auf «Daten löschen».</li> <li>2. Eine Abfrage erscheint, ob er die Metadaten wirklich löschen will oder nicht.</li> <li>3. Der User bestätigt das Löschen.</li> <li>4. Die Daten werden gelöscht.</li> <li>5. Eine Meldung erscheint, dass die Daten gelöscht werden könnten.</li> </ol>
<b>Alternativabläufe:</b>	(1-2) wie oben.

- |  |  |
|--|--|
|  | <ul style="list-style-type: none"> <li>3. Der User verneint das Löschen.</li> <li>4. Der User bleibt auf der Datenmasken-Seite.</li> </ul> |
|--|--|

Tabelle 60: UseCase 4

### 2.1.6.5 Metadaten speichern

Abschnitt	Inhalt
<b>ID:</b>	UC5
<b>Name:</b>	Metadaten speichern
<b>Version:</b>	1.4
<b>Datum:</b>	15.11.2021
<b>Kurzbeschreibung:</b>	Die Daten in der Eingabemaske werden gespeichert.
<b>Vorbedingung:</b>	Die Eingabemaske bzw. Datenmaske ist ausgefüllt und es wurde aus «Daten speichern» geklickt.
<b>Nachbedingung:</b>	Die Metadaten werden gespeichert und eine Meldung erscheint, dass das Speichern erfolgreich war.
<b>Fehlersituation:</b>	Es wurde auf «Daten speichern» geklickt und entweder ein ungültiges Zeichen steht in einem Feld, ein zwingend auszufüllendes Feld ist leer.
<b>Systemzustand im Fehlerfall:</b>	Eine Fehlermeldung erscheint und die Eingabemaske, mit den Daten bleibt auf dem Schirm.
<b>Akteure:</b>	User
<b>Trigger:</b>	Der User will die eingegebenen Daten abspeichern.
<b>Standardablauf:</b>	<ol style="list-style-type: none"> <li>1. Der User klickt auf «Daten speichern».</li> <li>2. Die Metadaten werden gespeichert.</li> <li>3. Eine Meldung erscheint, dass die Metadaten erfolgreich gespeichert wurden.</li> </ol>
<b>Alternativabläufe:</b>	<ol style="list-style-type: none"> <li>(1) wie oben.</li> <li>2. Ein zwingend auszufüllendes Feld ist leer oder ein falsches Zeichen wurde in einem Feld gefunden.</li> <li>3. Der User wird über den Fehler informiert.</li> <li>4. Hat der User das Pflichtfeld ausgefüllt oder die Eingabe korrigiert, kann er noch einmal «Daten speichern» klicken -&gt; UC5.</li> </ol> <ol style="list-style-type: none"> <li>1. Der User klickt auf «zurück» oder «Abbrechen».</li> </ol>

	<ol style="list-style-type: none"> <li>2. Ein Mitteilungsfenster erscheint und informiert den User, dass bei einem vorzeitigen Verlassen die Daten nicht gespeichert werden.</li> <li>5. Bei «OK» wird die Seite verlassen und der Home-Screen wird angezeigt.</li> <li>6. Bei «Abbrechen» bleibt der User auf der Seite, mit den eingegebenen Daten.</li> </ol>
	<ol style="list-style-type: none"> <li>(1) wie oben.</li> <li>1. Die Metadaten konnten nicht gespeichert werden.</li> <li>2. Der User wird mit einer Fehlermeldung informiert.</li> </ol>

Tabelle 61: UseCase 5

## 2.1.7 Schnittstellenbeschreibung

Die wichtigsten Schnittstellen sind die zur DB und zum lokalen Ordner. Über diese Schnittstellen werden die Metadaten gespeichert, gesucht, geladen, upgedatet und gelöscht.

Die Form der Einträge ist immer gleich. Für jedes Attribute des Metadaten-Objekts soll ein Wert stehen, notfalls ein Default-Wert. Das gilt für die Einträge der Datenbank aber auch für die Files auf dem Lokalen Ordner.

Damit ein Metadaten-File erstellt werden kann, braucht es zwei Werte, die der User auf jeden Fall eingeben muss: Die File-ID und der File-Titel. Für die Identifikation ist die MetadatenID und der Speicherort massgebend, aber diese werden vom System erstellt.

Die anderen Einträge sind fakultativ. Im nachfolgenden sind alle Attribute, die der User mitgeben muss bzw. kann, aufgelistet:

- Metadaten-ID (*metadataID*): Identifiziert die Metadaten eindeutig und besteht aus einer Kombination aus dem File-ID, dem File-Title und einer Zufallszahl mit 6 Stellen. Sie wird automatisch beim Eingeben der Metadaten generiert.
- File-ID (*fileID*): Dabei handelt es sich um die ID des Media Asset File, dessen Metadaten hier gespeichert werden sollen. Der Eintrag kann aus Buchstaben und/oder Zahlen bestehen. Wird vom User eingegeben und darf nicht leer gelassen werden.
- File-Titel (*fileTitle*): Dabei handelt es sich um den Titel des Media Asset File, dessen Metadaten hier gespeichert werden sollen. Der Eintrag kann aus Buchstaben und/oder Zahlen bestehen. Wird vom User eingegeben und darf nicht leer gelassen werden.

- File-Typ (*fileType*): Damit ist der Media-typ des Media Asset gemeint. Dabei kann der User zwischen den verschiedenen Media-Typen («Unbekannt», «Text», «Text mit Bild», «Bild», «Audio (Podcast)», «Audio (Musik)», «Video (Serie)», «Video (Film)» oder «Video (Sendung)») auswählen. Dieser Eintrag ist fakultativ. Der Default-Wert ist «Unbekannt».
- File-Grösse (*fileSize*): Hier geht es um die Grösse des Media Asset, dessen Metadaten gespeichert werden sollen. Die Eingabe soll eine Zahl mit einem Punkt als Trennzeichen sein. Dieser Eintrag ist fakultativ. Der Default-Wert ist «0.1».
- Grösseneinheit (*fileSizeUnit*): Die Grösseneinheit der File-Grösse wird wahlweise in «B» (Byte), «KB» (Kilobyte), «MB» (Megabyte) oder «GB» (Gigabyte) angegeben, die der User auswählen kann. Dieser Eintrag ist fakultativ. Der Default-Wert ist «B».
- Erstellungsdatum (*fileDate*): Dieser Eintrag steht für das Erstellungsdatum des Media Asset Files, dessen Metadaten gespeichert werden. Dieser Eintrag ist fakultativ. Das Default-Datum ist der 1.1.1970, aber natürlich geht auch ein älteres Datum.
- Autor (*fileAuthor*): Damit wird hinterlegt, wer der Ersteller des Media Asset Files war. Dieser Eintrag ist fakultativ. Der Default-Wert ist «».
- Besetzung (*fileCast*): Dabei kann der User den Cast, also die Schauspieler bei einem Video oder die Sprecher eins Audios, eintragen und abspeichern. Dieser Eintrag ist fakultativ. Der Default-Wert ist «».
- Beschreibung (*description*): Hier hat der User die Gelegenheit das Media Asset File näher zu beschreiben. Dieser Eintrag ist fakultativ. Der Default-Wert ist «keine Beschreibung vorhanden.».
- Speicherort (*saveLocation*): Dieser Eintrag beschreibt der Speicherort des Metadaten-Files. Die Wahl trifft der User, die Applikation schreibt dann den Eintrag.

## 2.2 Softwareentwurfsdokumente

### 2.2.1 Architekturpattern

#### 2.2.1.1 GUI

Für das GUI verwende ich Java FXML und damit das Model-View-Controller-Pattern.

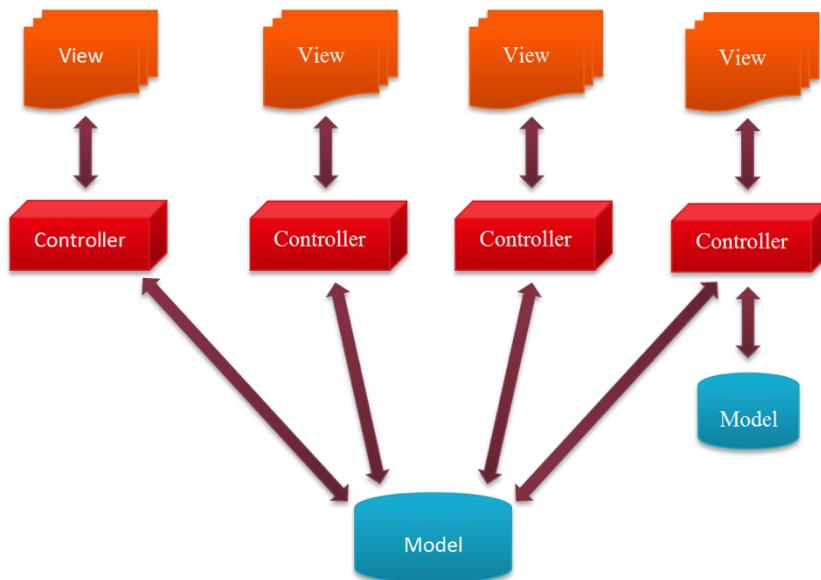


Abbildung 6: Model-View-Controller-Pattern (Quelle [3])

Das **Model**, auf deutsch das Datenmodell, ist zuständig für die Speicherung und Bereitstellen der Daten zuständig die die Controller von der View haben oder auf ihr brauchen. Dazu stellt das Model den Controllern Hilfsfunktionen zur Verfügung, um Codeduplikation zu verhindern. Die **View**, auf deutsch die Präsentation, ist für den Style und das Layout der Grafikoberfläche zuständig.

Der **Controller**, auf deutsch Programmsteuerung, ist das Bindeglied zwischen Model und View. Er regelt die Logik der View, steuert die Elemente der View an und stellt ihr Methoden zur Verfügung, wie z.B. die «ButtonHandler». [3]

#### 2.2.1.2 Daten speichern und laden

Damit ich die Daten aus der DB und aus einem lokalen Ordner aufbereiten kann, sei es zum Speichern oder zum Abrufen, benutze ich das Strategy-Pattern.

*Das Verhalten (Funktionalität, Algorithmus) eines Objekts (dem Context) in eine eigene Strategiekasse (Strategie = gekapselter Algorithmus) ausgelagert. Der Context hält eine Referenz auf sein Strategieobjekt und wenn er das ausgelagerte Verhalten ausführen soll, so delegiert er den Aufruf an sein referenziertes Strategieobjekt. Der Context arbeitet dabei nicht mit einer konkreten Implementierung, sondern mit einer Schnittstelle. Er ist damit implementierungsunabhängig und kann somit mit neuen Verhalten ausgestattet werden, ohne dass sein Code dafür geändert werden muss. Einzige Bedingung ist, dass die neue Strategie das Strategy-Interface korrekt implementiert. [4]*

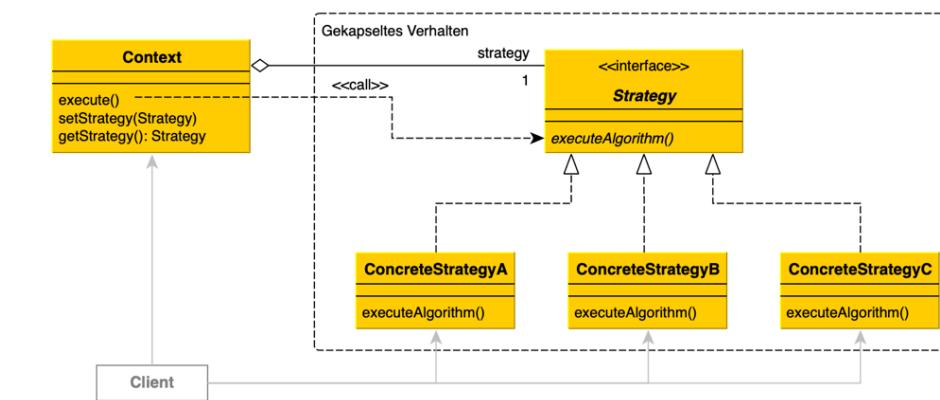


Abbildung 7: Strategy-Pattern (Quelle [4])

### 2.2.1.3 Singleton-Pattern

Im Model speichere ich die Daten, die in die View eingegeben und vom Controller ausgelesen werden, und leite sie weiter. Die gespeicherten Daten werden aber auch in anderen Views gebraucht. Damit ich das gewährleisten kann, muss ich sicherstellen, dass nur ein Model-Objekt in der Applikation erstellt und darauf zugegriffen wird. Dies erreiche ich mit einem Singleton-Pattern. Dieses Pattern hat zum Ziel, dass eben nur ein Objekt einer Klasse gemacht wird und auch immer auf dieses zugegriffen wird. Dies wird mit einem `private static final`-Objekt erreicht, dass einmal erstellt wird und allen anderen Klassen zur Verfügung steht.

## 2.2.2 Sichten der Softwarearchitektur

### 2.2.2.1 Verteilungssicht

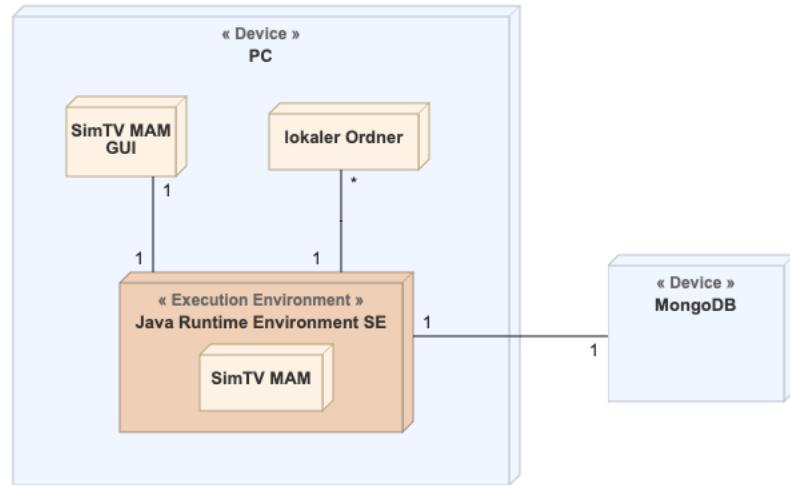


Abbildung 8: Verteilungsschicht

### 2.2.2.2 Kontextabgrenzung

Der User bedient über das GUI die Applikation. Die Applikation dagegen kommuniziert ausserhalb des Systems mit einem lokalen Ordner mit der MongoDB.

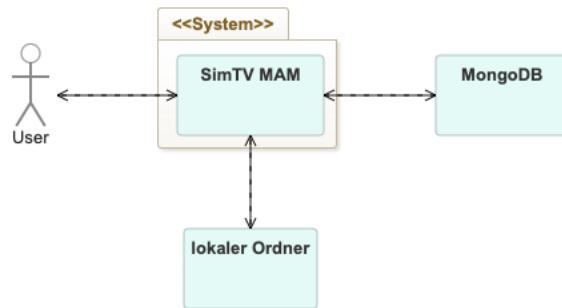


Abbildung 9: Kontextabgrenzung

### 2.2.2.3 Bausteinsicht Gesamtsystem

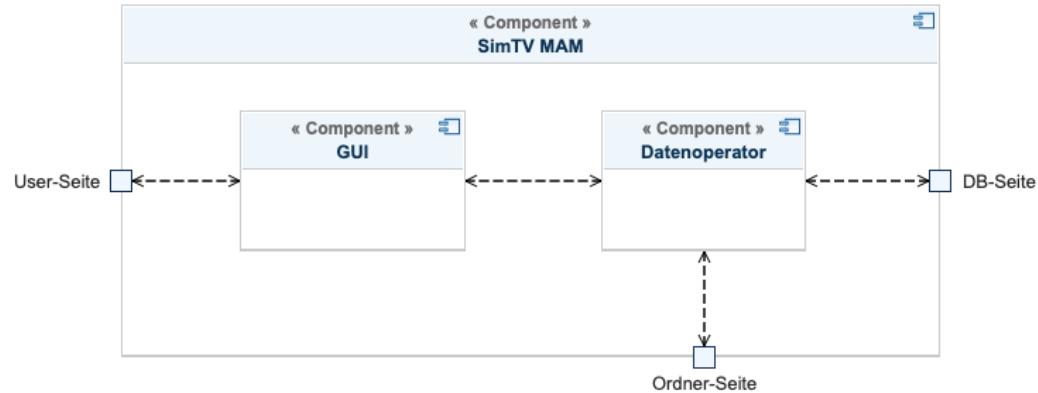


Abbildung 10: Bausteinsicht Gesamtsystem

### 2.2.2.4 Bausteinsicht GUI

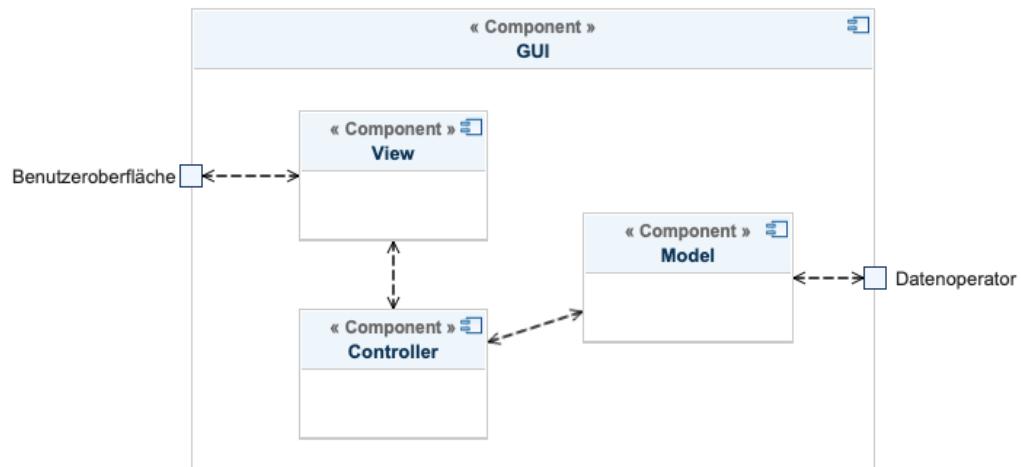


Abbildung 11: Bausteinsicht GUI

### 2.2.2.5 Bausteinsicht Datenoperator

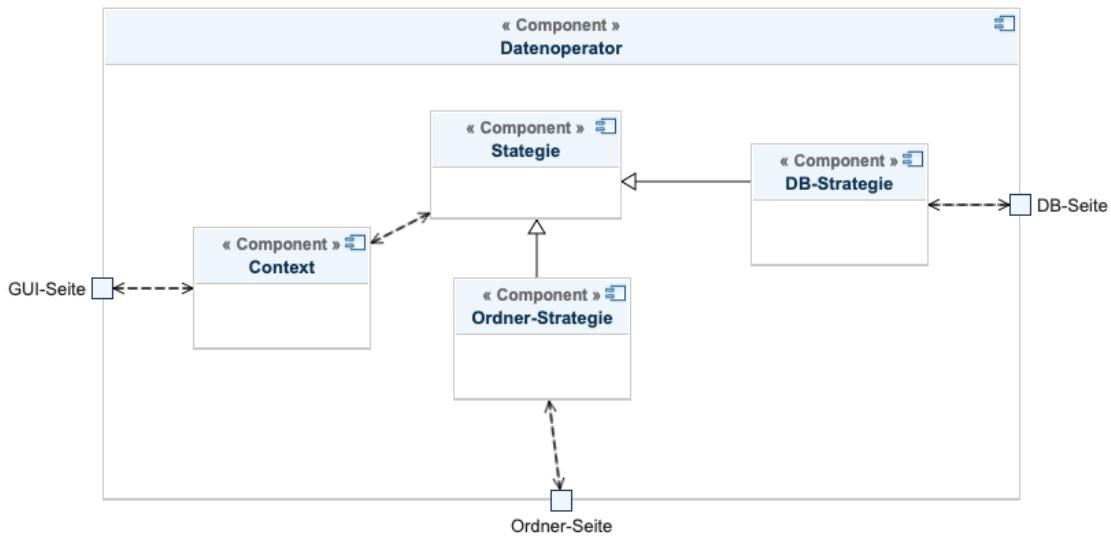


Abbildung 12: Bausteinsicht Datenoperator

### 2.2.3 Klassendiagramm

#### 2.2.3.1 Das geplante Klassendiagramm

Nach einem Überblick über das geplante Klassendiagramm, ohne Methoden oder Attributen, folgen die Detailansichten des Klassendiagramms auf den folgenden Seiten.

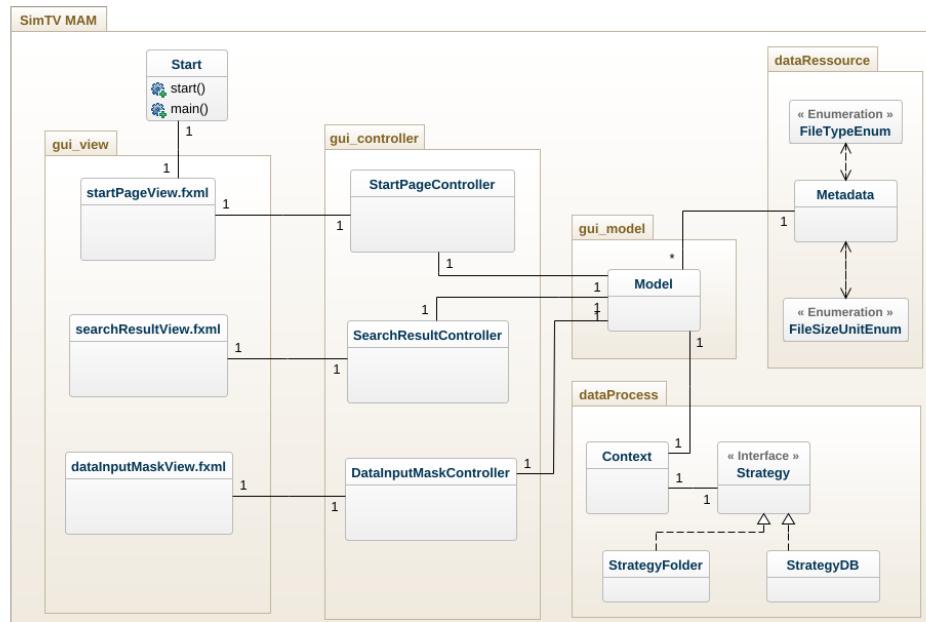


Abbildung 13: Geplantes Klassendiagramm Übersicht

Die Detailansichten zeigen die Ausschnitte, die von den Rechtecken, auf der nächsten Abbildung, umfasst werden.

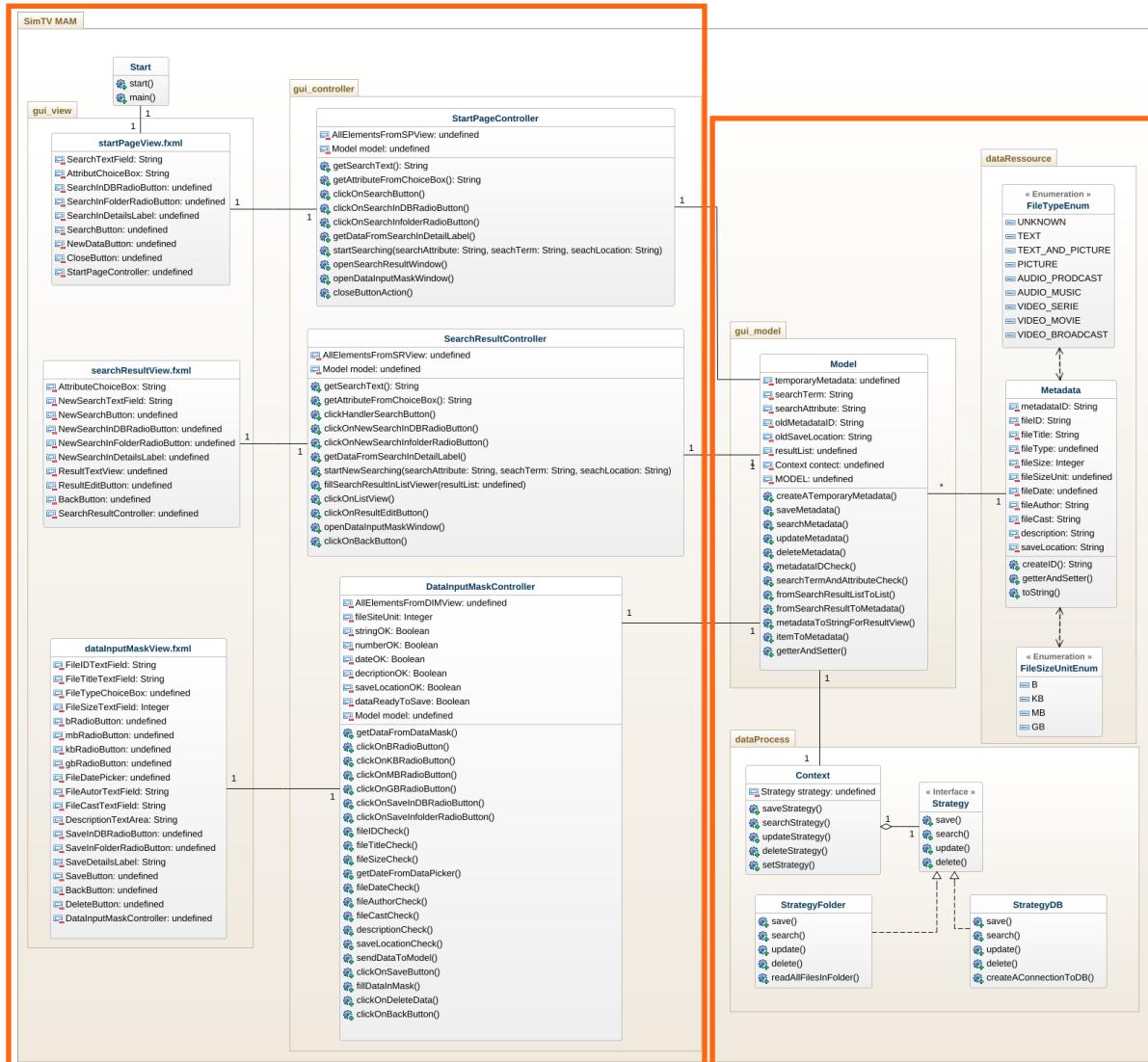


Abbildung 14: Einteilung für die Detailansichten

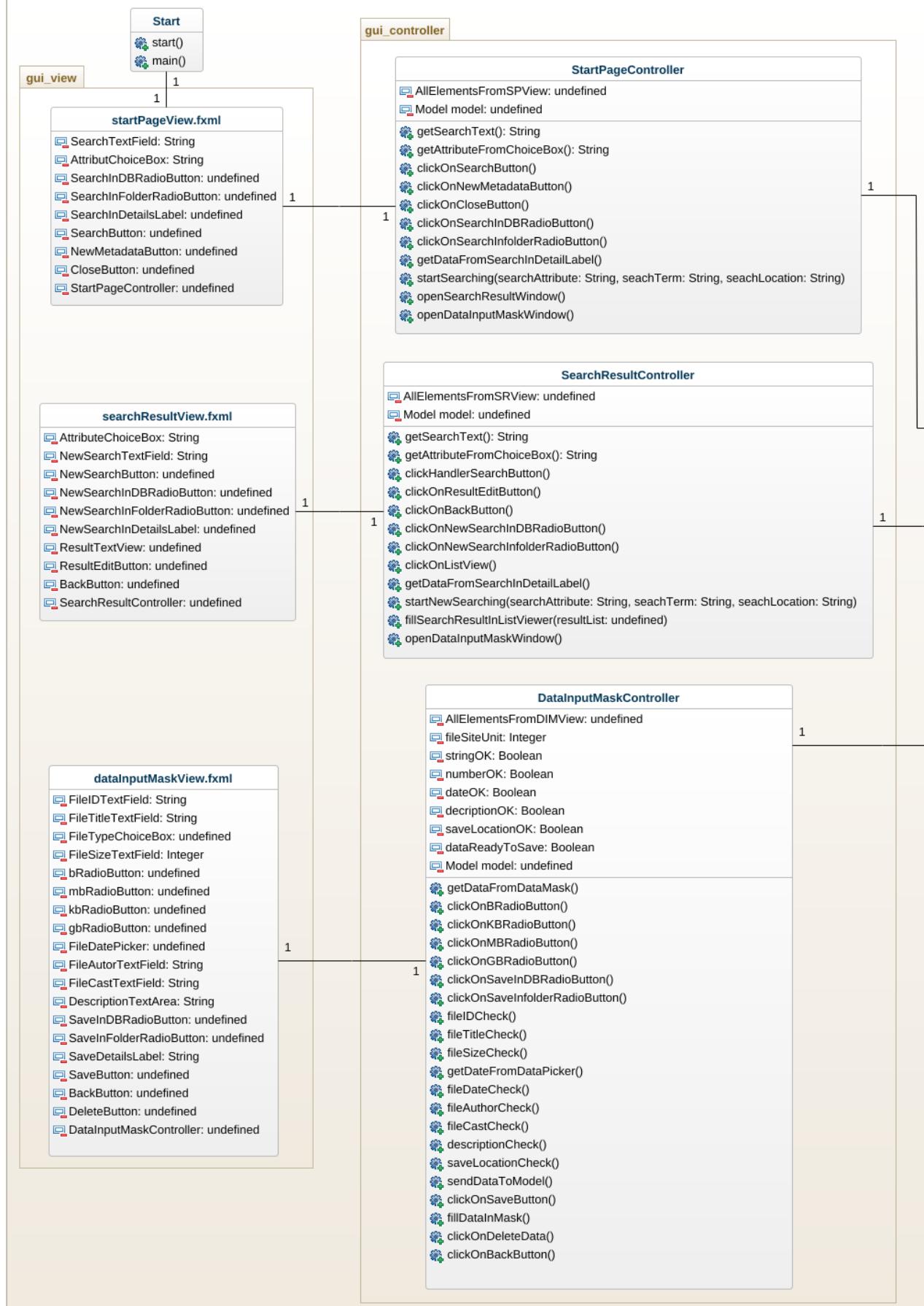


Abbildung 15: Klassendiagramm Detailansicht links

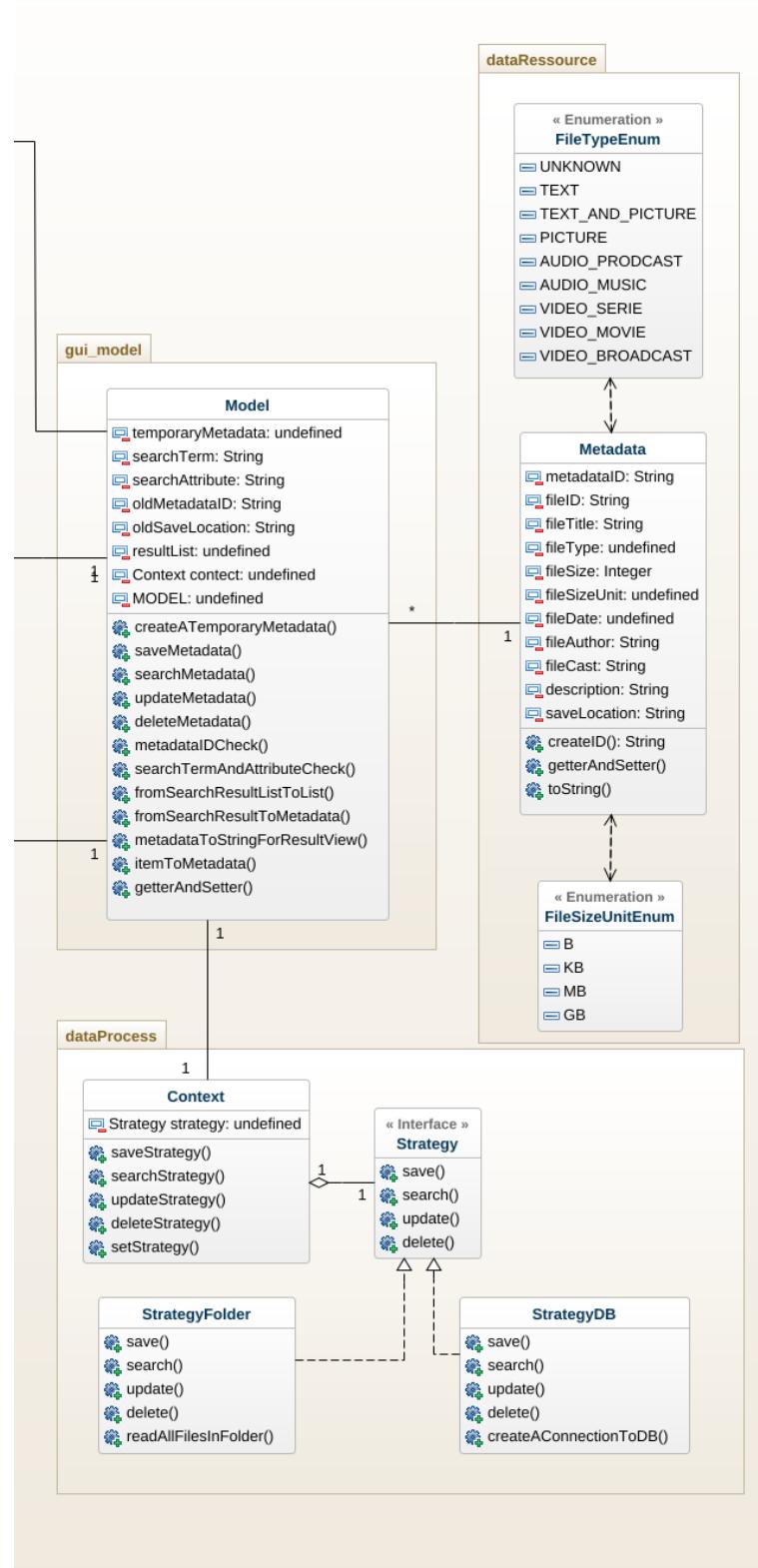


Abbildung 16: Klassendiagramm Detailansicht rechts

### 2.2.3.2 Das implementierte Klassendiagramm

Wie im Entscheidungs-Log EL-Nr. 19 beschrieben, füge ich nachfolgend ein weiteres Klassenmodell ein. Dieses habe ich mit IntelliJ erstellt und es beinhaltet die Klassen, die ich schlussendlich im abgegebenen Projekt habe. Die Unterschiede zum geplanten Klassendiagramm hat verschiedene Gründe: Vielerorts musste ich grosse Methoden auf mehrere Kleine aufteilen damit die Anforderung «Eine Aufgabe Pro Methode», soweit es ging, gewährleistet blieb. Auch musste ich einige Methoden ausgelagert, um Code-Duplikationen zu verhindern. Dazu kommen einige Namensänderungen, die ich während des Implementierungsphase getätigt habe. Das implementierte Klassendiagramm ist aber leider zu gross, als dass es auf eine Seite passen würde. Darum habe ich es in fünf Teile aufgeteilt. In der nachfolgenden Abbildung ist eine Übersicht mit violetten Rechtecken als Klassen zu sehen. Darin zeigen die Rechtecke die jeweiligen detaillierten Ausschnitte an. Am Schluss noch eine weniger detaillierte Ansicht mit den Verbindungen und Beziehungen zwischen den Klassen.

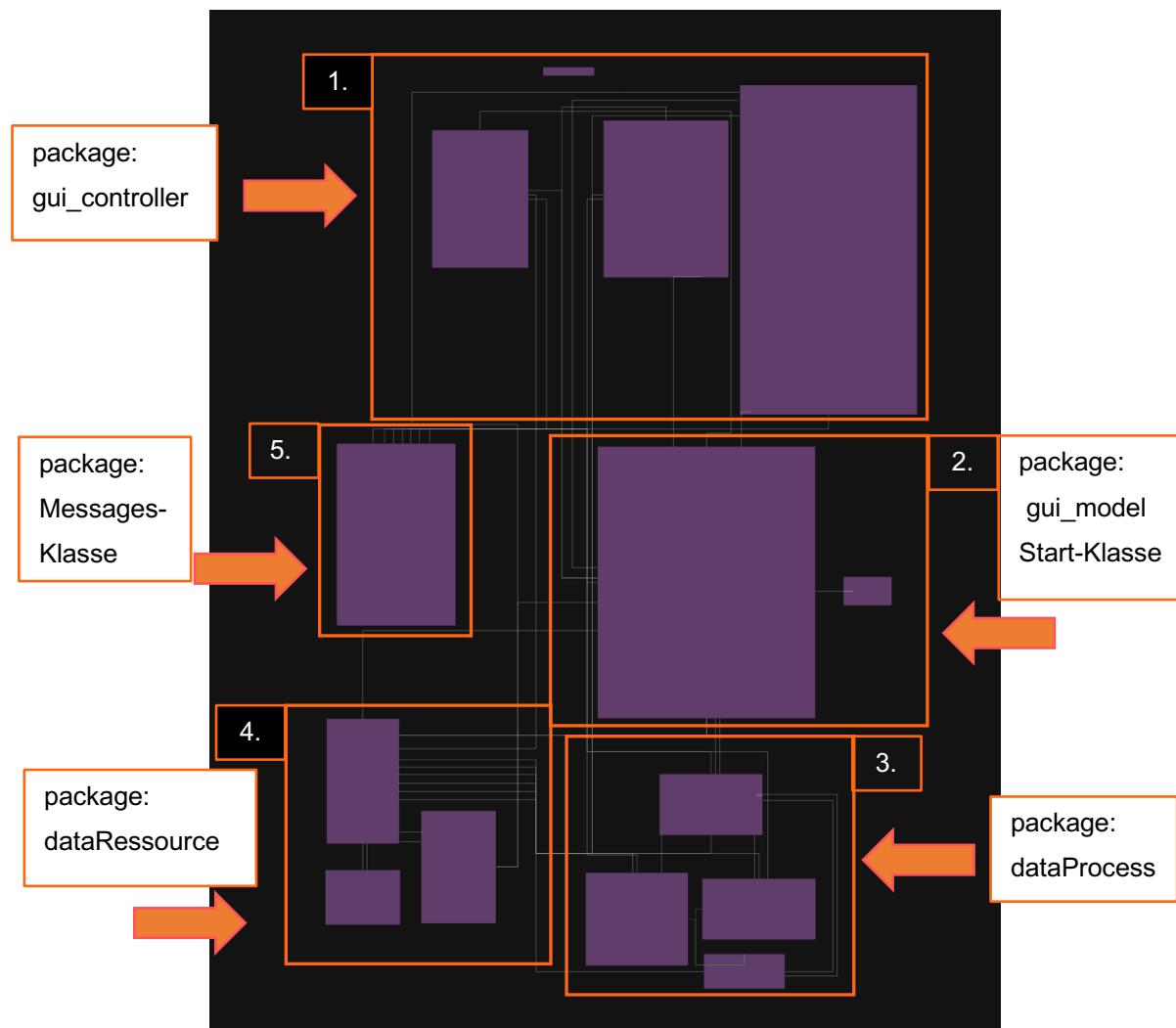


Abbildung 17: Implementiertes Klassendiagramm Übersicht

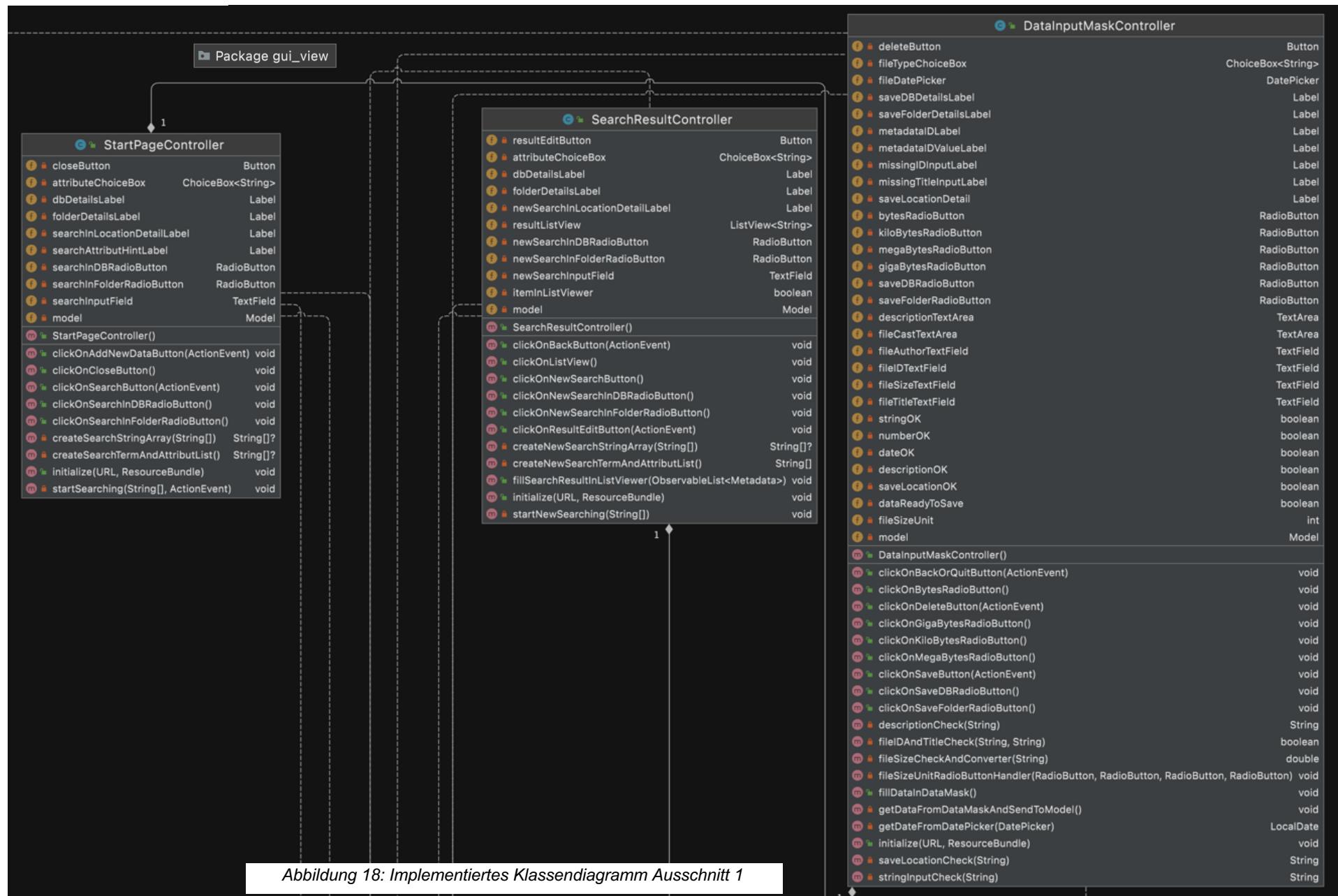


Abbildung 18: Implementiertes Klassendiagramm Ausschnitt 1

### 2.2.3.2.1 Implementiertes Klassendiagramm, Ausschnitt 1

In dem ersten Ausschnitt, in der Abbildung vorher, ist das Paket «gui\_controller» zu sehen. Das beinhaltet die drei Kontroller für die drei Views. Da diese drei Views des Paketes «gui\_view» keine Java-Klassen, sondern FXML-Files sind, sind sie auf dem Abschnitt nur als Paketname zu sehen. Jedoch bedient jeder Controller eine View und stellt die Methoden dafür zur Verfügung.

### 2.2.3.2.2 Implementiertes Klassendiagramm, Ausschnitt 2

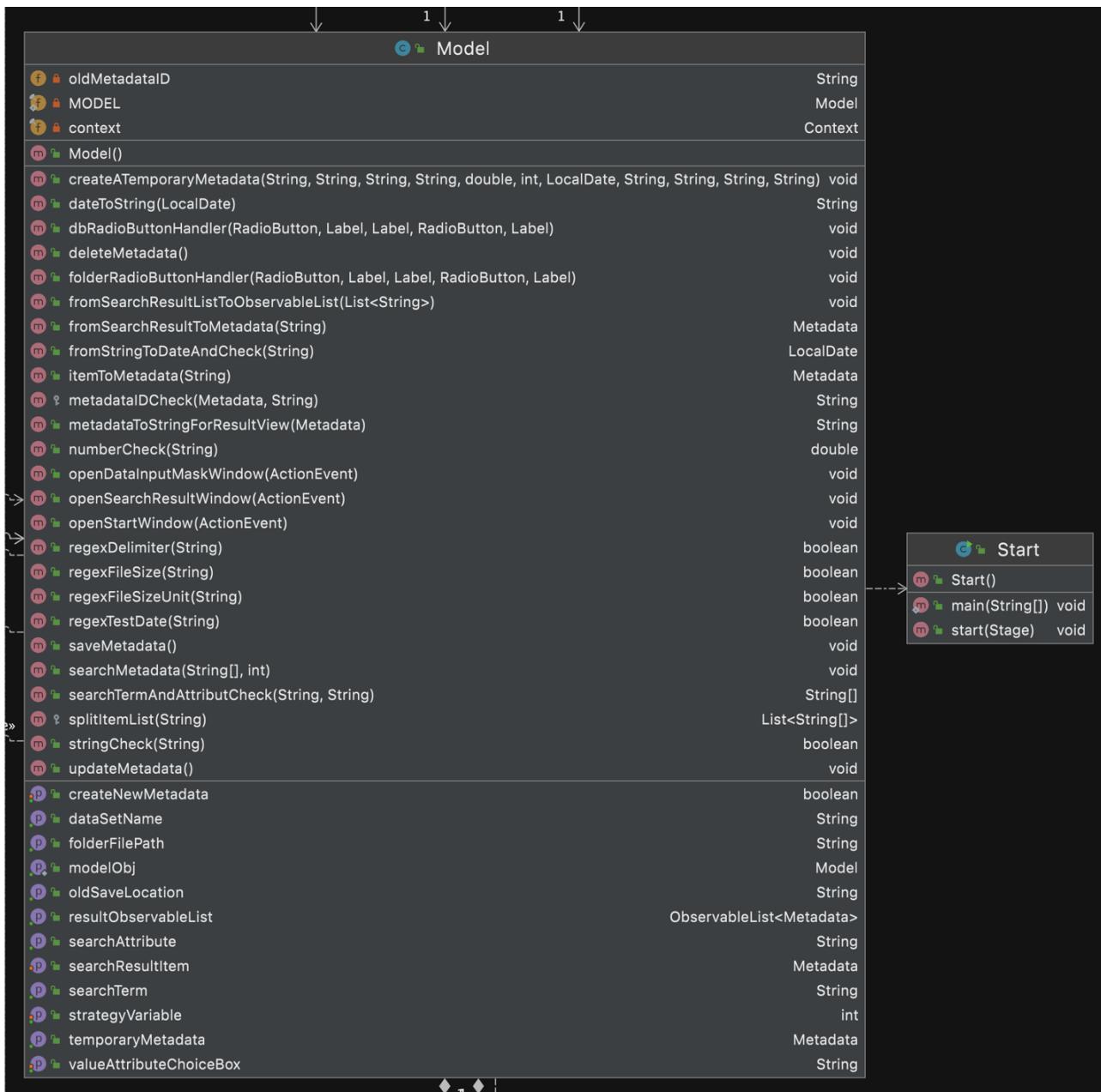


Abbildung 19: Implementiertes Klassendiagramm Ausschnitt 2

Der zweite Ausschnitt zeigt die Model-Klasse. Diese ist die einzige Klasse im Paket «gui\_model». Sie verbinden die drei Controller miteinander, sichert ihre Daten und stellt ihnen auch wieder Daten zur Verfügung. Gleichzeitig verbindet das Model das GUI mit dem Context und sorgt für den Datenaustausch. Die Start-Klasse wurde von IntelliJ automatisch mit dem Model verknüpft. Die Start-Klasse hat nur eine Aufgabe, nämlich das Starten der Anwendung.

### 2.2.3.2.3 Implementiertes Klassendiagramm, Ausschnitt 3

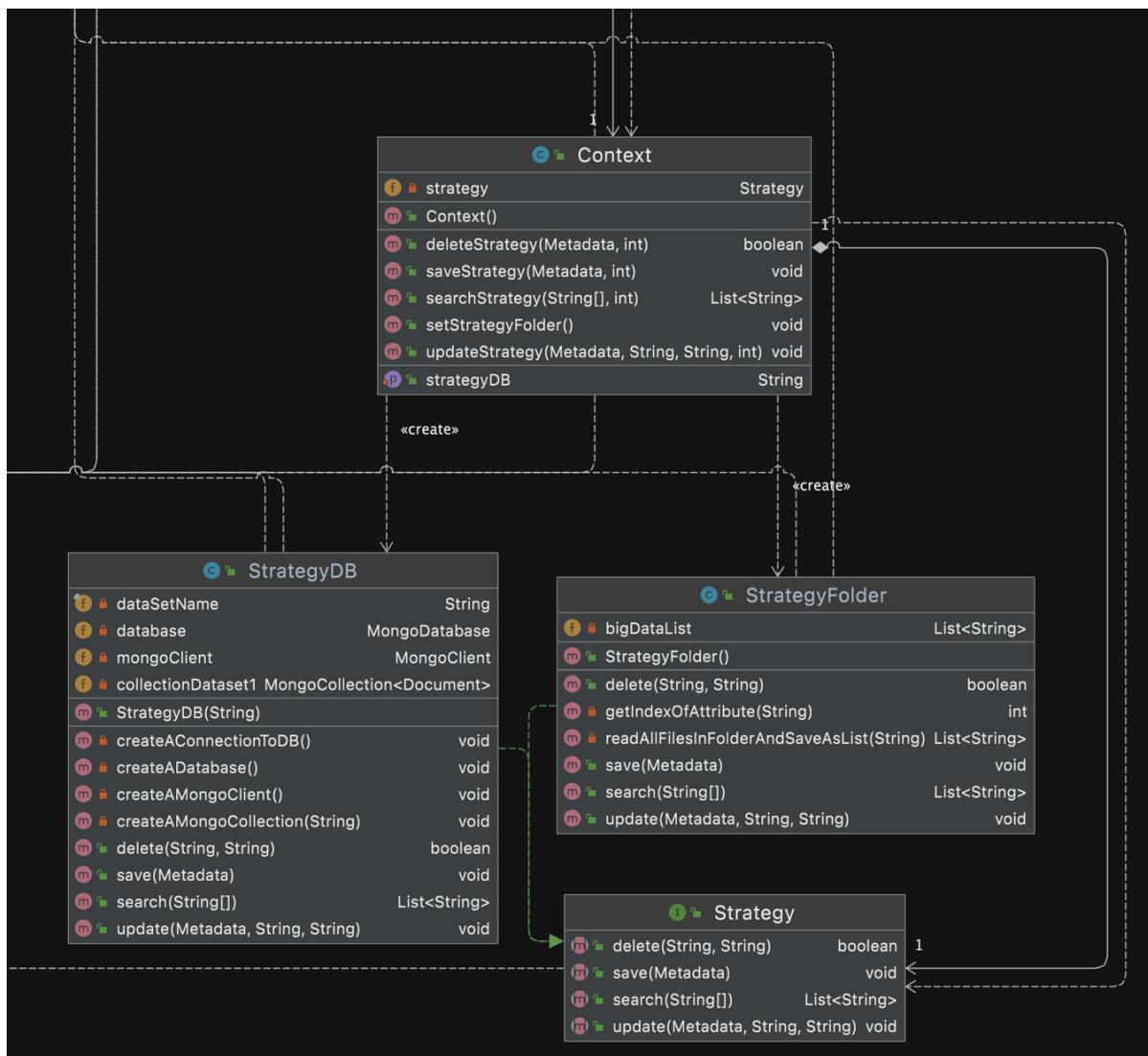


Abbildung 20: Implementiertes Klassendiagramm Ausschnitt 3

Der dritte Ausschnitt zeigt das «dataProcess»-Paket, das das Strategy-Pattern umsetzt, dass im Abschnitt 2.2.1.2 beschreiben ist. Ich bin der Namensgebung, so weit wie möglich gefolgt.

#### 2.2.3.2.4 Implementiertes Klassendiagramm, Ausschnitt 4

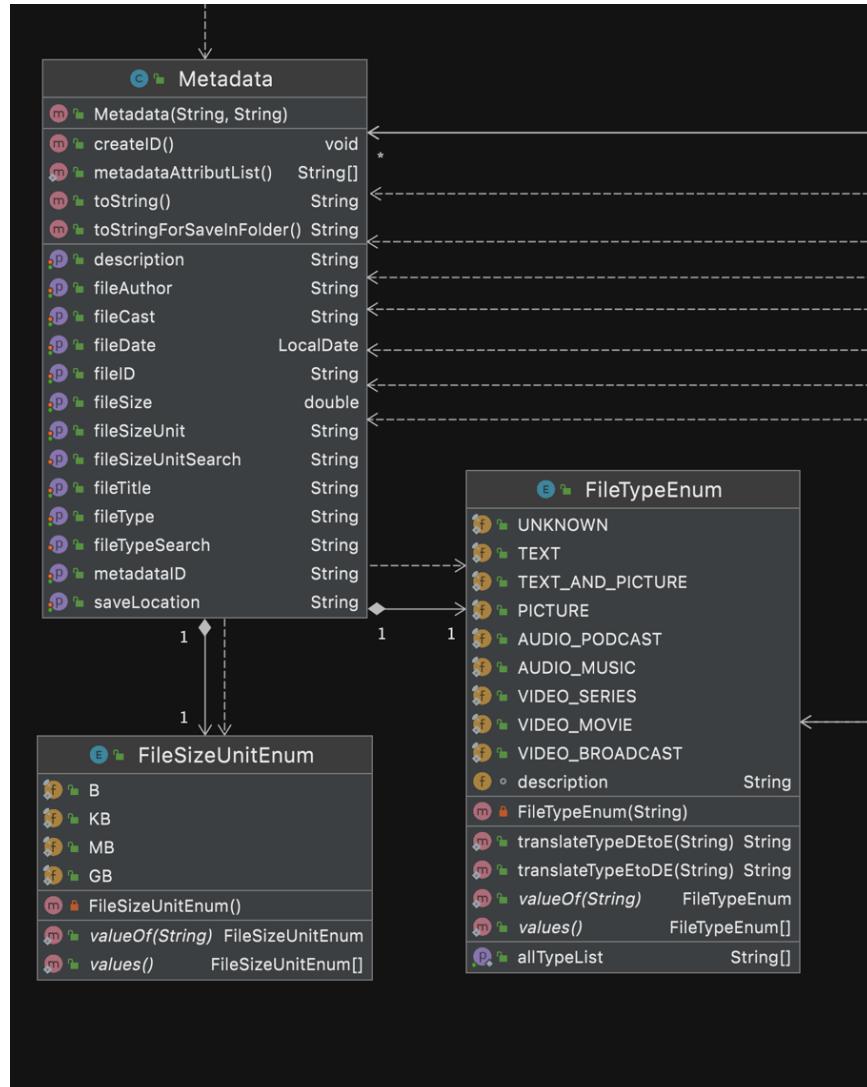


Abbildung 21: Implementiertes Klassendiagramm Ausschnitt 4

Im vierten Ausschnitt ist das Paket «dataRessource» zu sehen, dass die Klassen «Metadata» und die beiden Enum-Klassen beinhaltet. Die «Metadata»-Klasse legt die Objekte fest, die in dieser Anwendung gespeichert, gesucht, upgedatet und gelöscht werden.

### 2.2.3.2.5 Implementiertes Klassendiagramm, Ausschnitt 5

Der letzte Ausschnitt zeigt meine «Messages»-Klasse, die alle JavaFX-Nachrichten beinhaltet, die ich für meine Anwendung geschrieben habe. Diese Nachrichten beinhalten Informations- Fehler- oder Bestätigungs meldungen.

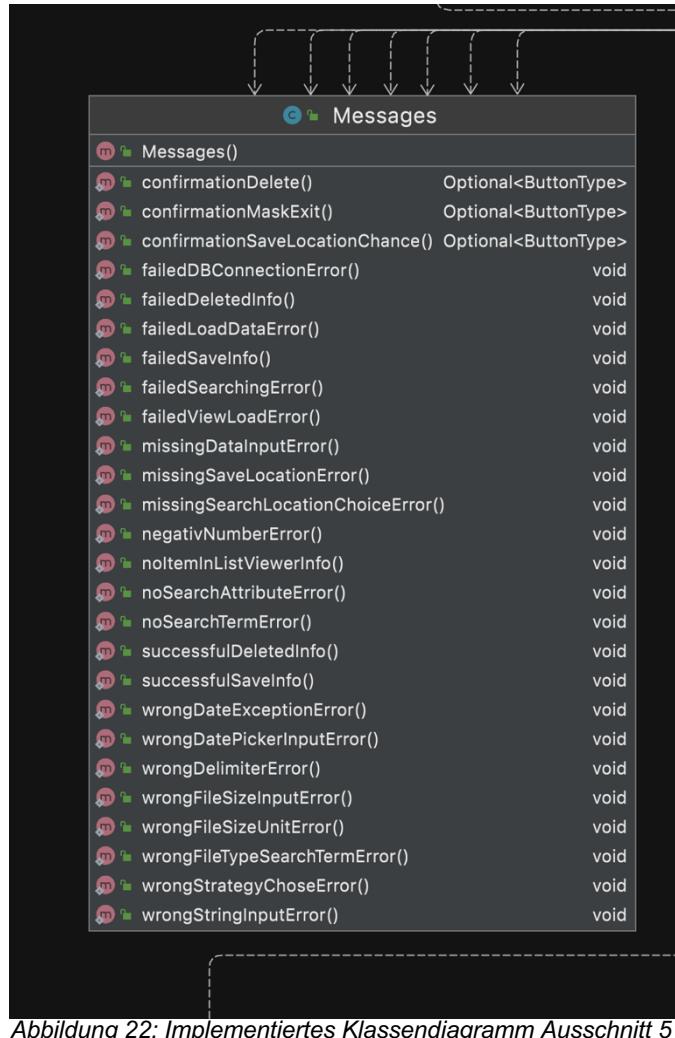


Abbildung 22: Implementiertes Klassendiagramm Ausschnitt 5

#### **2.2.3.2.6 Implementiertes Klassendiagramm, Verbindungen**

Und hier, zum Schluss, das Klassendiagramm mit den Verbindungslien.

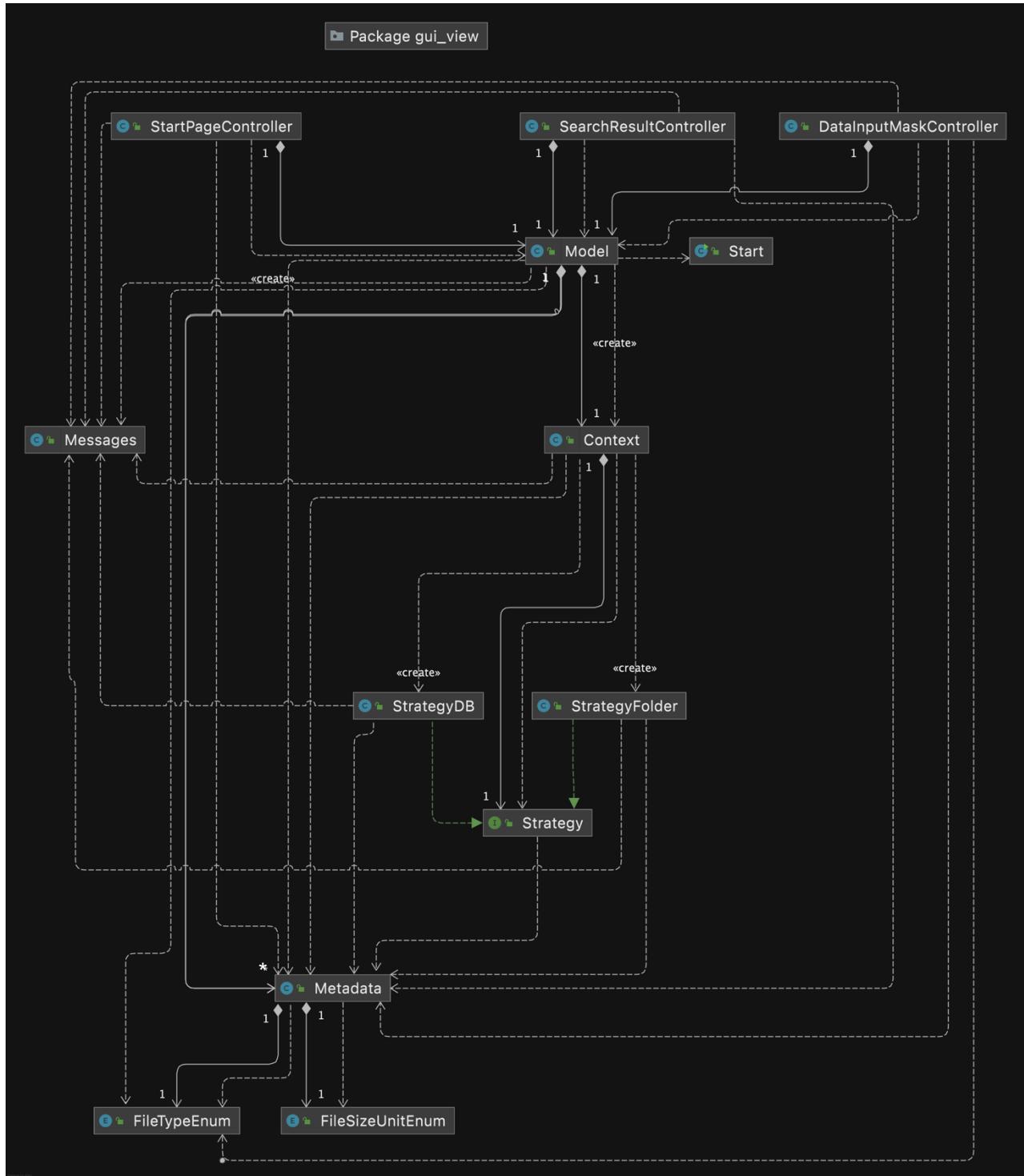


Abbildung 23: Implementiertes Klassendiagramm mit den Beziehungen

## 2.2.4 Dynamische Modelle

Ich habe mich für die Darstellung der dynamischen Modelle, die die Abläufe wiedergeben, für Aktivitätsdiagramme entschieden. Die Diagramme zeigen, in der Reihenfolge die Abläufe für:

- Anwendung schliessen.
- Neue Metadaten hinzufügen und speichern
- Metadaten suche
- Metadaten bearbeiten und speichern oder löschen

### 2.2.4.1 Legende der Aktivitätsdiagramme

Diese fünf Elemente kommen in den Aktivitätsdiagrammen vor. Die Namen erklären was welches Element darstellt.



Abbildung 24: Elemente des Aktivitätsdiagramme

### 2.2.4.2 Aktivitätsdiagramm «Anwendung schliessen»

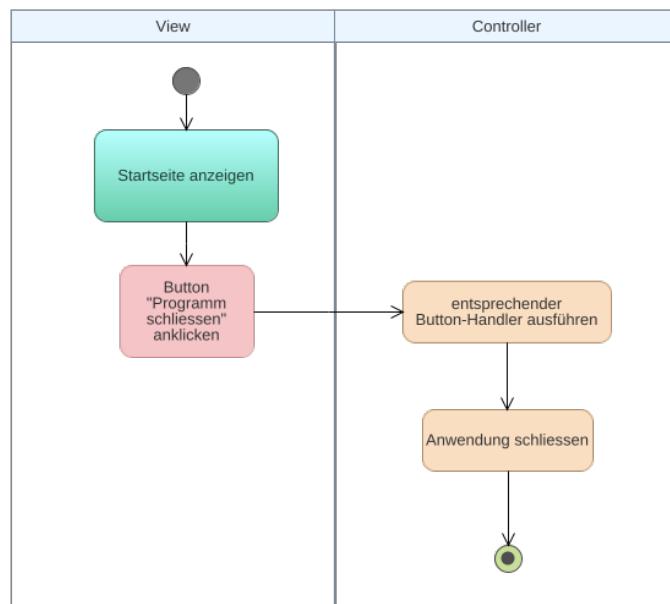


Abbildung 25: Aktivitätsdiagramm «Anwendung schliessen»

### 2.2.4.3 Aktivitätsdiagramm «Neue Metadaten hinzufügen und speichern»

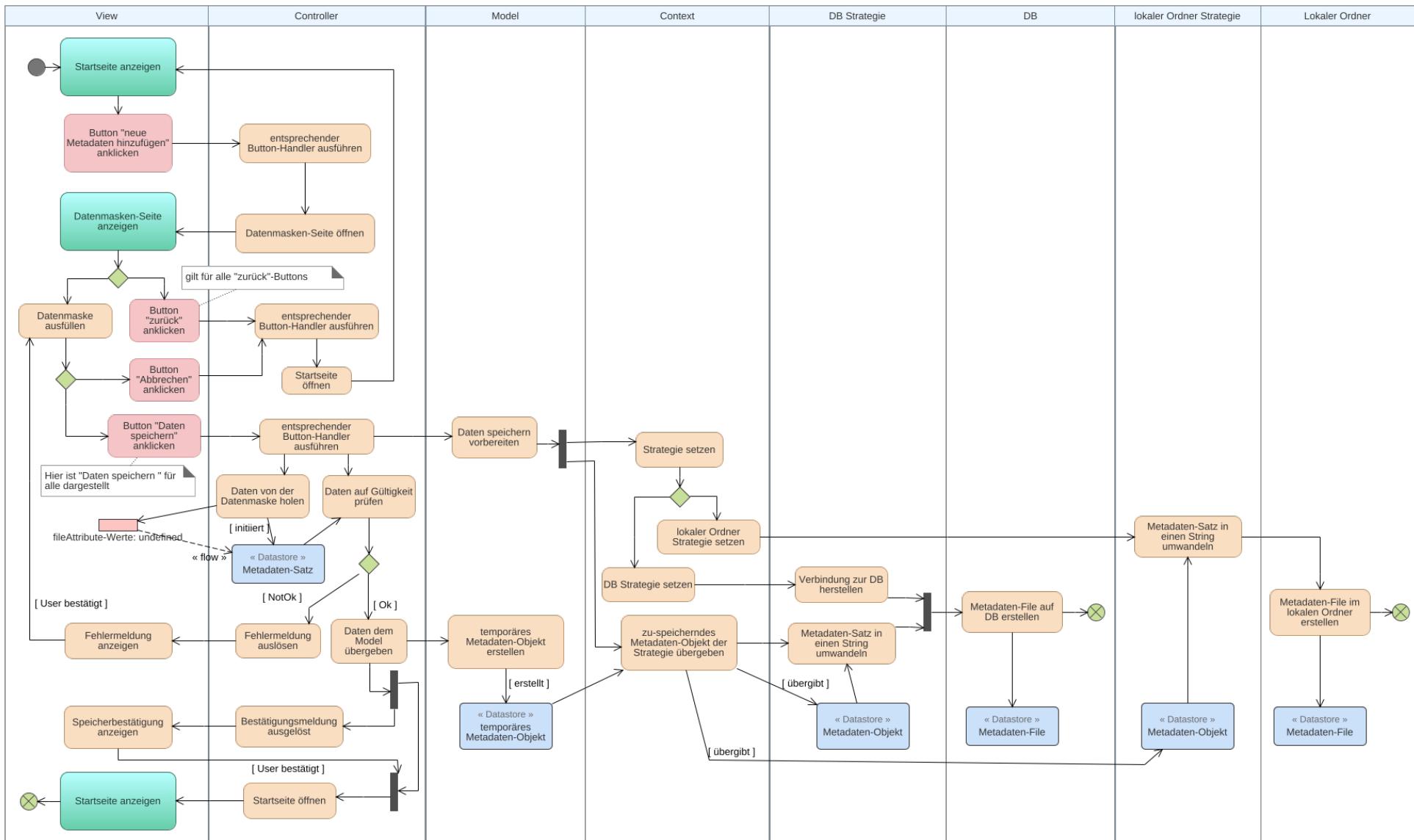


Abbildung 26: Aktivitätsdiagramm «Neue Metadaten hinzufügen und speichern»

## 2.2.4.4 Aktivitätsdiagramm «Metadaten suchen»

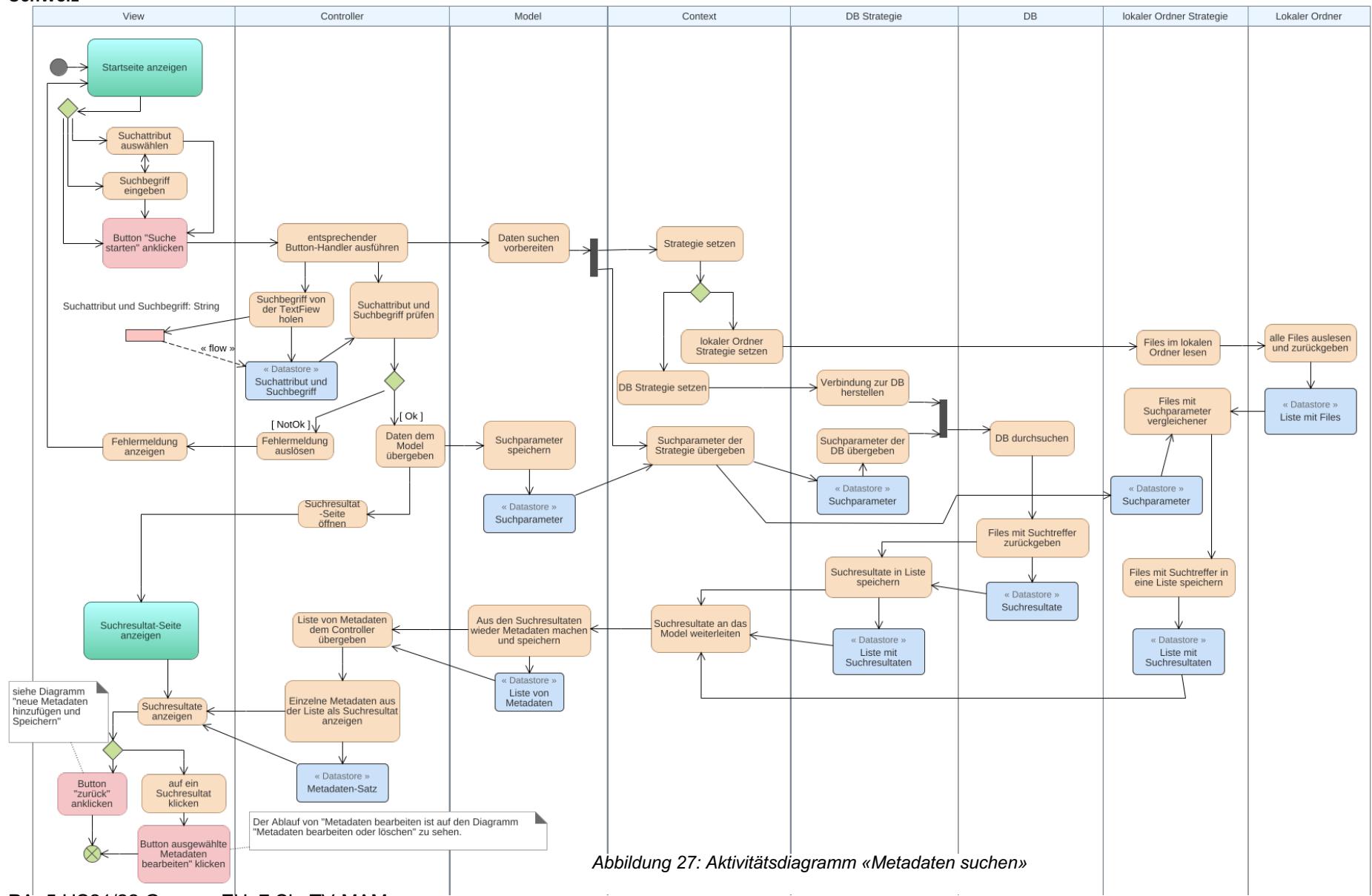


Abbildung 27: Aktivitätsdiagramm «Metadaten suchen»

## 2.2.4.5 Aktivitätsdiagramm «Metadaten bearbeiten und speichern oder löschen»

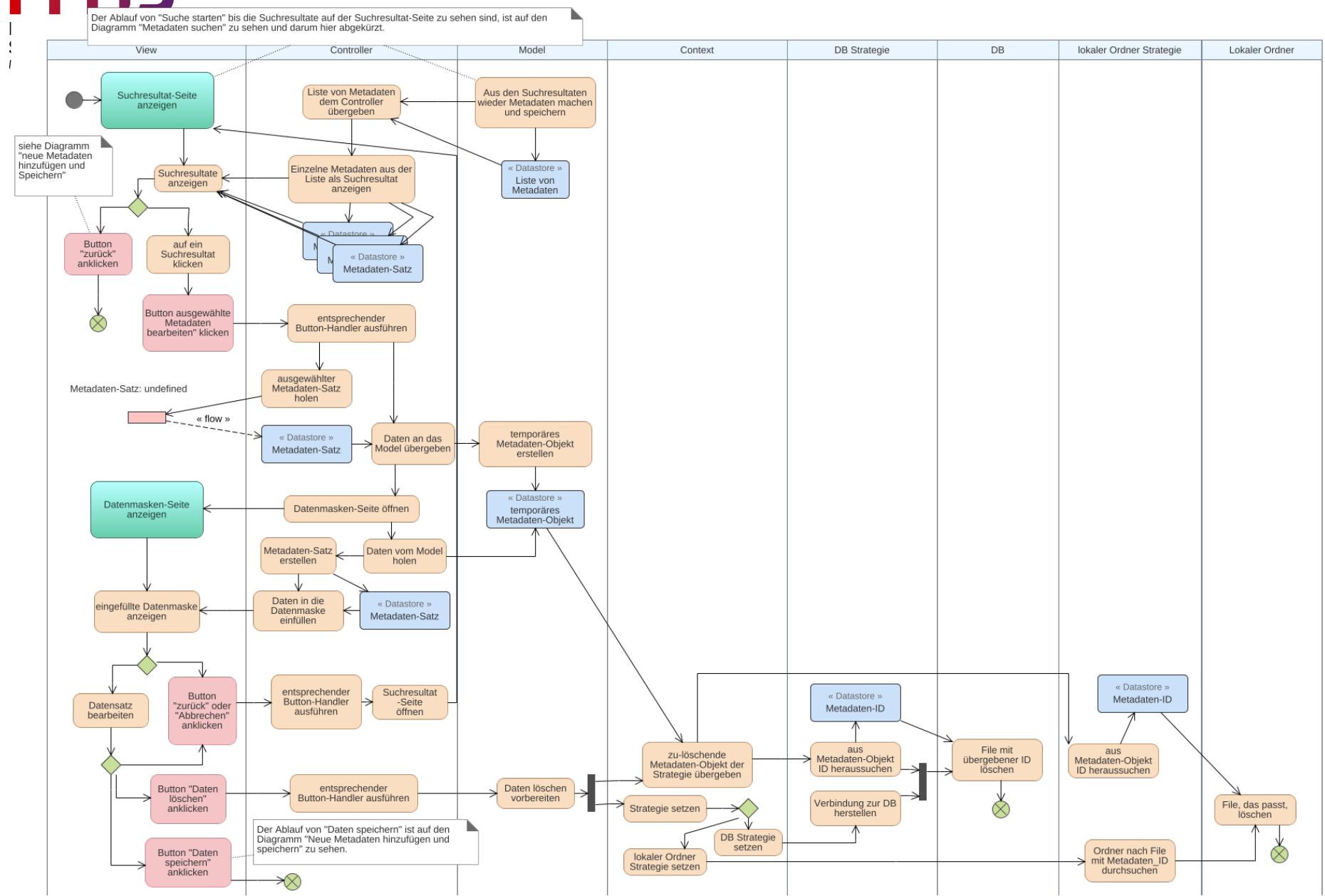


Abbildung 28: Aktivitätsdiagramm «Metadaten bearbeiten und speichern oder löschen»

## 2.2.5 Spezifikation der Bedienoberflächen

### 2.2.5.1 MockUps als Entwurf



Abbildung 30: MockUp der Startseite

Das MockUp zeigt die Metadaten-Maske. Der Titel 'Metadaten - Maske' befindet sich in einem orangefarbenen Balken oben links. Unter dem Titel befindet sich ein 'zurück'-Button. Die Formularfelder sind wie folgt angeordnet: 'Metadaten-ID' (Textfeld), 'File-ID:' (Textfeld), 'File-Titel:' (Textfeld), 'File-Type' (ChoiceBox), 'File-Grösse:' (Textfeld mit EinheitenradioButtons: B, KB, MB, GB), 'File-Erstellungsdatum' (Textfeld mit Datepicker-Icon), 'File-Autor:' (Textfeld), 'File-Besetzung:' (Textfeld), 'Beschreibung:' (Textfeld). Am unteren Rand befindet sich ein Bereich für den Speicherort mit den Optionen 'DB' (ausgewählt), 'Lokaler Ordner' und 'Hinweis-Text zum Speicherort'. Am unteren Rand stehen die Buttons 'Daten speichern', 'Daten löschen' und 'Abbrechen'.

Abbildung 29: MockUp der Metadaten-Maske

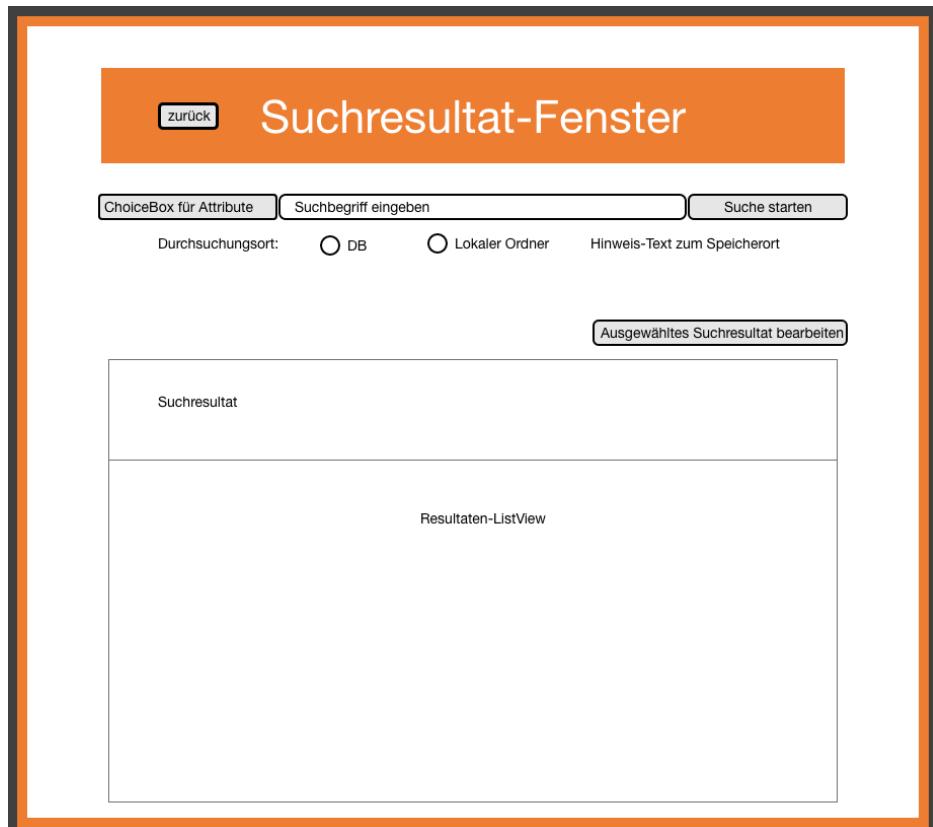


Abbildung 31: MockUp des Suchresultat-Fensters

Die MockUps zeigen den Entwurf der Benutzeroberfläche. Sie galten als Vorbild für die Implementierung. Wie es schlussendlich im fertigen Programm aussieht, zeigen die Screenshots.

### 2.2.5.2 Screenshot vom fertigen Projekt

Für die Erläuterung und Spezifikation der Benutzeroberflächen, benutze ich die nachfolgenden Screenshots. Die roten und weissen Pfeile und die nummerierten, rot-umrandeten Rechtecken sind nicht Teil der Screenshots und dient zur Zuweisung der Informationen.

### 2.2.5.3 Startseite



Abbildung 32: Screenshot der Startseite beim Starten der Applikation

Die 1 gibt den Bereich mit der Suchfunktion an. Hier sind 5 Elemente, mit denen agiert werden kann:

- a ist eine ChoiceBox, mit der das Suchattribut ausgewählt werden kann. Die Abbildung 38 (rechts) zeigt die Auswahl, die erscheint, wenn darauf geklickt wurde.
- b ist ein Textfeld, in das ein Suchbegriff eingegeben werden kann. Dieser Suchbegriff kann aus Buchstaben, Zahlen und Zeichen bestehen. Keine Eingabe ist je nach dem auch möglich.
- c und d sind zwei RadioButton und dienen zur Auswahl des zu-durchsuchenden Ortes.
- Wenn c oder d angeklickt werden, dann erscheint ein Schriftzug unterhalb der RadioButtons. In der Abbildung 39 und 40 sind beide Schriftzüge zu sehen. Es handelt sich dabei um eine detailliertere Information über den Speicherort.

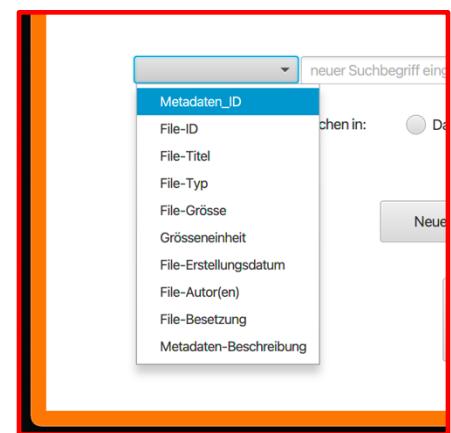


Abbildung 33: Angeklickte ChoiceBox

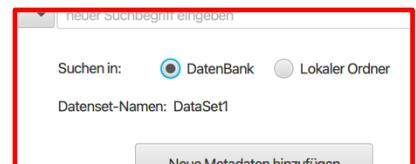


Abbildung 34: Info-Text DB

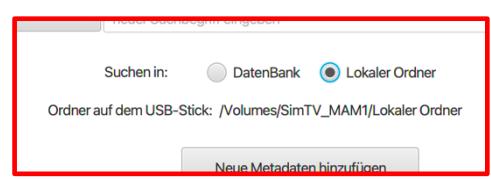


Abbildung 35: Info-Text Lokaler Ordner

- e ist schliesslich der Such-Button, der die Suche startet. Davor kontrolliert die Suchfunktion aber noch drei Dinge:

1. Ist ein Suchattribut ausgewählt worden?
2. Ist ein Ort zum Durchsuchen ausgewählt worden?
3. Passt der Suchbegriff zum Sich-Attribut?

- Wurde kein Suchattribut ausgewählt wurde, wird die Fehlermeldung in Abbildung 41 ausgelöst und ein Hinweis-Text über der ChoiceBox eingeblendet.

- Wurde kein Ort zum Durchsuchen ausgewählt, dann erscheint die Fehlermeldung in Abbildung 42.
- Wurde ein Suchbegriff eingegeben, der nicht zum Suchattribut passt, oder wurde keinen eingegeben und es wird einer erwartet, dann erscheint eine Fehlermeldung,

mit passendem Informationstext. Im Abschnitt 2.2.5.6 ist eine Tabelle, die die Regeln und Fehlermeldungen bei deren Verletzung enthält.

- Wenn e gedruckt wird und es wurde ein Suchattribut ausgewählt, einen gültigen Suchbegriff eingegeben und ein Ort zum Durchsuchen wurde gewählt, dann wird die Suchfunktion ausgeführt. Dann öffnet sich das Suchfenster (siehe Abbildung 43).

Mit der 2 wird die Metadaten-Masken-Seite geöffnet. Dies ist dann in der Abbildung --- zu sehen.

Schlussendlich zeigt die 3 die Wege, mit denen sich die Applikation schliessen lassen. Einmal mit dem Button «Programm beenden» und einmal mit dem (bei mir) roten Punkt im oberen linken Fenstereck.



Abbildung 36: Fehler bei nicht-auswahl eines Suchattributes

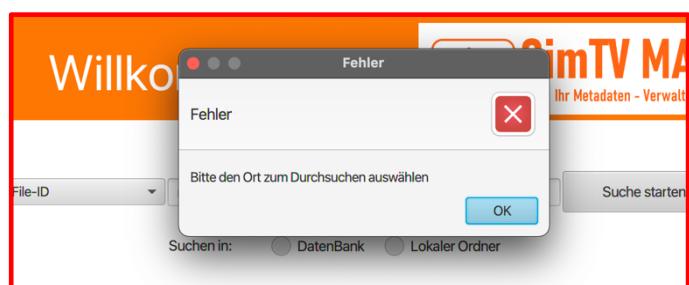


Abbildung 37: Fehler bei nicht-auswahl eines Such-Ortes

## 2.2.5.4 Suchfenster

Als Beispieleingabe habe ich als Suchattribut «File-ID», als Suchbegriff «id» und als Durchsuchungsort «DatenBank» ausgewählt. Das Suchfenster, das sich danach geöffnet hat, sieht folgendermassen aus:

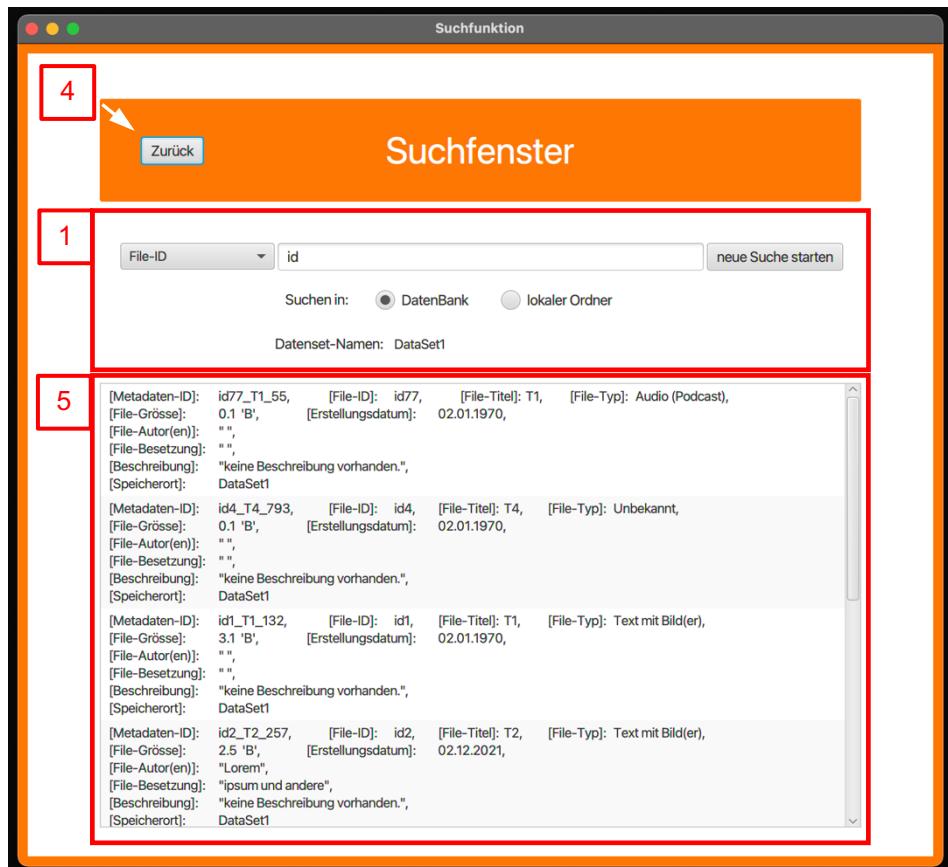


Abbildung 38: Screenshot des Suchfensters

Der Bereich mit der 1 ist derselbe, wie auf der Startseite. Alle 5 Elemente haben dieselbe Funktion wie die anderen 5 und auch die Such-Parameter-Überprüfung und die Fehlermeldungen sind identisch. Daher werde ich hier nicht weiter darauf eingehen.

Die 4 zeigt auf den «zurück»-Button. Wenn der geklickt wird, dann wird die Startseite geöffnet.

Der Bereich 5 zeigt alle Suchresultate an. Alle Attribute mitsamt den Werten sind zu abzulesen. Wenn kein Suchtreffer erfolgte, dann steht in Mitte der Fläche «Keine Treffer gefunden!». Auf die Fläche, auf der die Resultate zu sehen sind, kann geklickt werden und so Suchresultate ausgewählt werden. Durch das Klicken wird ein Button Sichtbar, auf dem «Ausgewählte Metadaten bearbeiten» steht. Ist in Abbildung 44 zu sehen.

Wird auf den Button geklickt, dann öffnet sich die Metadaten-Masken-Seite mit den Werten der Attribute in den entsprechenden Feldern. Dazu später mehr.

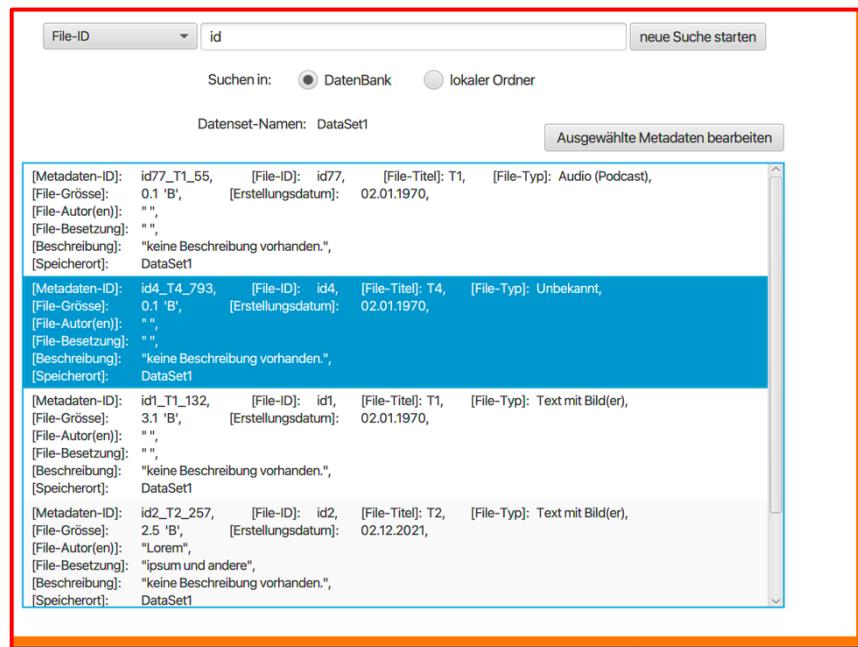


Abbildung 39: Auswahl eines Suchresultates

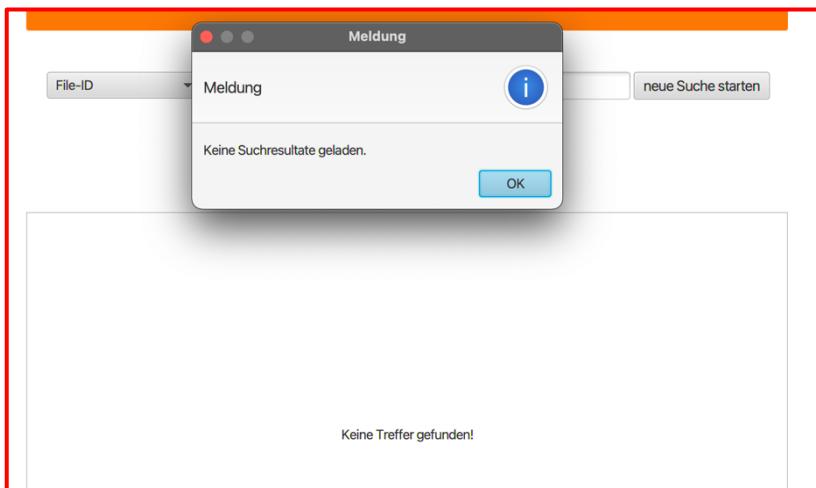


Abbildung 40: Info-Meldung, wenn keine Treffer gefunden wurden und ins Feld geklickt wurde.

Wenn keinen Suchtreffer gemacht wurde, dann kommt eine Informationsmeldung, dass kein Suchresultat geladen ist.

### 2.2.5.5 Metadaten-Maske

Dieses letzte Fenster beinhaltet die Datenmaske. Um auf diese Seite zu kommen, muss man entweder in dem Startfenster auf «Neue Metadaten hinzufügen» klicken oder im Suchfenster

ein Suchresultat auswählen und dann auf «Ausgewählte Metadaten bearbeiten» klicken. Doch die Seiten sehen dann anders aus. Zuerst die 1. Variante über das Startfenster:

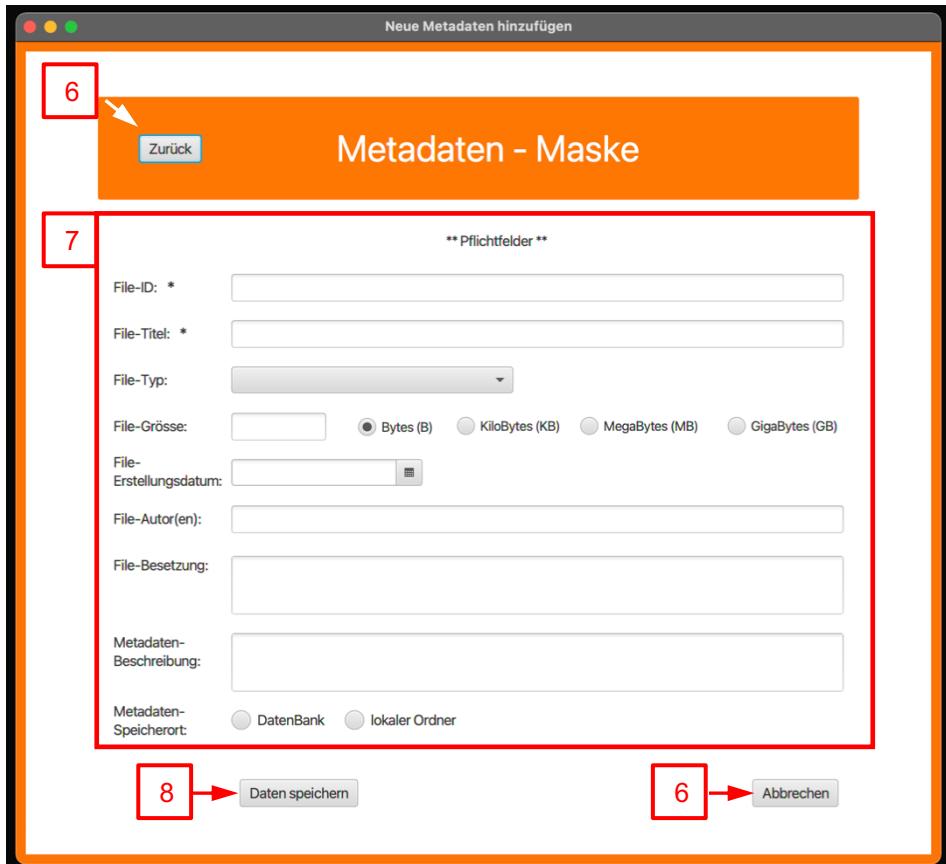


Abbildung 41: Screenshot der Metadaten-Masken-Seite / Variante 1

Die 6 zeigt auf den «zurück»- oder auf den «Abbrechen»-Button. Wenn einer der beiden geklickt wird, kommt eine Meldung, die informiert, dass die ausgefüllten Daten nicht gespeichert werden, wenn die Seite jetzt verlassen wird. Diese Meldung ist in der Abbildung 47 zu sehen. Wird «Ok» geklickt, öffnet sich die Startseite, bei «Abbrechen» bleibt man auf der Metadaten-Masken-Seite.

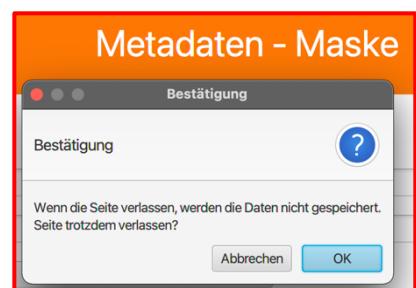


Abbildung 42: Bestätigungsmeldung, vor dem Verlassen der Seite.

Der Bereich mit der 7 ist die eigentliche Datenmaske. Die Einträge werden erst kontrolliert, wenn 8 der «Daten speichern»-Button geklickt wurde. Für die Daten werden von oben nach unten kontrolliert. Als erstes wird überprüft, ob die Pflichtfelder «File-ID» und «File-Titel» ausgefüllt sind. Sind sie es nicht, kommt eine Fehlermeldung und die «\*» bei den beiden Attributen wird rot eingefärbt und verdoppelt, als Hinweis für den User. Die Fehlermeldung ist in der Abbildung 48 zu sehen.

Die anderen Felder haben alle einen Default-Wert und können leer gelassen werden, bis auf den «Metadaten-Speicherort». Dort muss einer ausgewählt werden. Ist kein Speicherort ausgewählt, dann erscheint eine Fehlermeldung. Zu sehen in Abbildung 49.

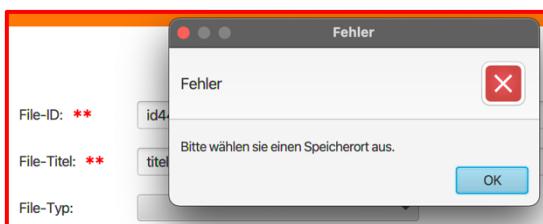


Abbildung 44: Fehlermeldung kein Speicherort

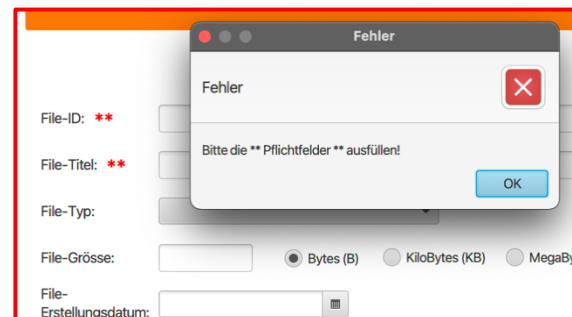


Abbildung 43: Fehlermeldung, bei leerem Pflichtfeld

Für die Überprüfung der anderen Einträge gelten dieselben Regeln, wie bei der Suchfunktion. Deren Regeln und Fehlermeldungen im Abschnitt 2.2.5.6 zu finden sind. Es werden zwar nicht alle gebraucht, da bei «File-Typ» und «Grösseneinheit» keine Falscheingabe getägt werden kann. Aber alle anderen treffen zu.

Wenn auf die RadioButton für den «Metadaten-Speicherort» geklickt wird, erscheint daneben wieder ein Schriftzug mit detaillierteren Informationen über die Speicherorte. Die Informationen entsprechen den Abbildungen 41 und 42.

Wenn alle eingegebenen Daten gültig sind und ein Speicherort ausgewählt wurde, dann lassen sich die Daten mit einem Klick auf **8** dem «Daten speichern»-Button speichern. Wurde das Speicher erfolgreich durchgeführt, dann erscheint eine Informationsmeldung, die das Speichern bestätigt. Zu sehen in der Abbildung 50.

Klick der User auf «Ok», öffnet sich das Startfenster.



Abbildung 45: Bestätigungsmeldung

Schauen wir uns noch die 2. Variante an, bei der wir über das Suchfenster kommen. In der Abbildung 51 ist das ausgewählte Suchresultat zu sehen. Nach den klicken auf den «Ausgewählte Metadaten bearbeiten»-Button öffnet sich folgendes Fenster:

[Metadaten-ID]:	id2_T2_257,	[File-ID]:	id2,	[File-Titel]:	T2,	[File-Typ]:	Text mit Bild(er),
[File-Groesse]:	2.5 'B',	[Erstellungsdatum]:	02.12.2021,				
[File-Autor(en)]:	"Lorem",						
[File-Besetzung]:	"Ipsum und andere",						
[Beschreibung]:	"keine Beschreibung vorhanden.",						
[Speicherort]:	DataSet1						

Abbildung 46: Das ausgewählte Suchresultat

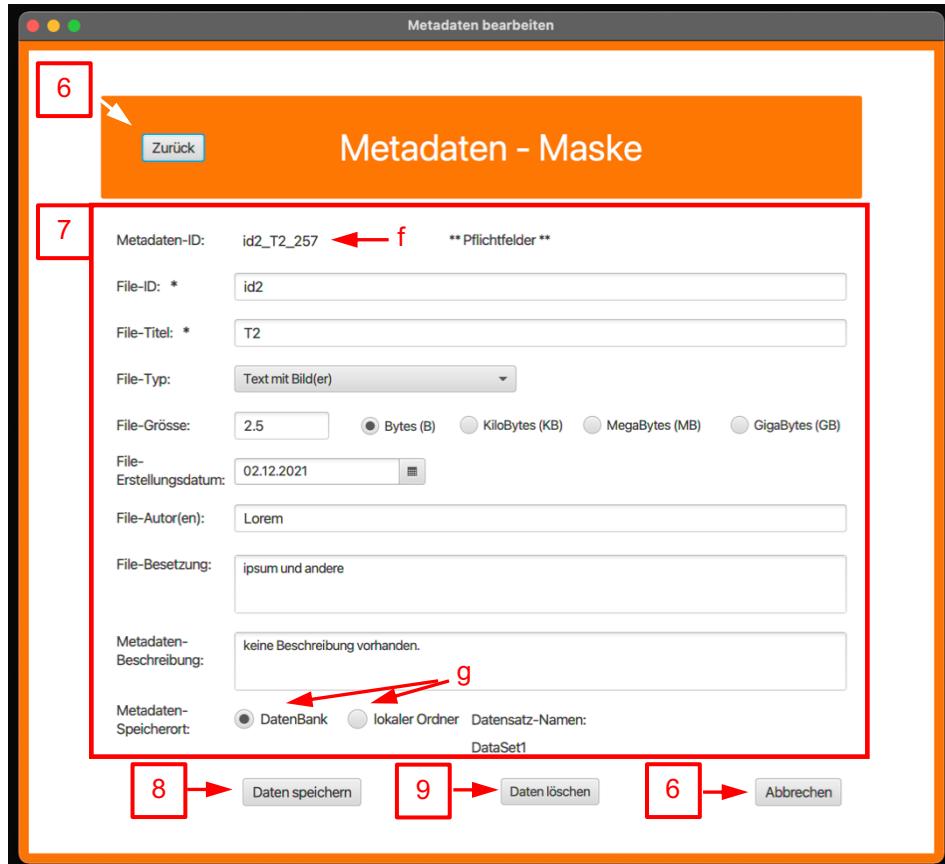


Abbildung 47: Screenshot der Metadaten-Masken-Seite / Variante 2

Die 6 zeigt wieder auf den «zurück»- bzw. auf den «Abbrechen»-Button. Wieder kommt die Bestätigungsmeldung, die in der Abbildung 47 zu sehen ist. Nur das dieses Mal, wenn «Ok» geklickt wurde, das Suchfenster geöffnet wird und die letzten Suchresultate geladen werden. Bei «Abbrechen» bleibt man auf der Metadaten-Masken-Seite.

Im Bereich 7 ist zu sehen, dass die Werte der Attribute in die Felder eingefüllt worden sind. Diese können jetzt bearbeitet werden. Es ist jetzt auch die oberste Zeile, die Metadaten-ID, zu sehen. Da sie automatisch generiert wird, fehlt sie bei der Variante 1. Die Metadaten-ID ist nicht bearbeitbar.

Die Daten von File-ID bis Metadaten-Beschreibung sind nach Herzenslust zu bearbeiten, so lange die Regeln (siehe Tabelle in 2.2.5.6.) eingehalten werden. Selbst die File-ID und der File-Titel können geändert werden. Wenn diese beiden Felder bearbeitet wurden, dann wird beim Speichern eine neue Metadaten-ID erstellt.

Eine Besonderheit ist allerdings das Ändern des Speicherortes bei **g**. Es ist möglich, doch es erscheint eine Warnung, dass das Metadaten-File dann am alten Ort gelöscht wird. Diese Warnung ist in der Abbildung 53 zu sehen. Gelöscht wird erst, wenn auf «Daten speichern» geklickt wird.



Abbildung 48: Bestätigungsmeldung für den Wechsel des Speicherortes

Wenn alle Änderungen abgeschlossen sind, dann kann mit den klicken auf **8** «Daten speichern», die Überprüfung der eingefügten Werte durchgeführt werden. Dies geht gleich von Statten, wie wenn ich neue Metadaten hinzufüge, also Variante 1.

Mit einem Klick auf **9** «Daten löschen» kann ich dieses Metadaten-File aus dem Speicherort löschen. Durch das Klicken auf den Button wird ein Bestätigungs-Fenster angezeigt, ob man die Daten wirklich löschen möchte. Klickt man auf «OK», dann werden die Daten gelöscht. Mit einem Klick auf «Abbrechen» passiert nichts.

Es gibt eine kleine Besonderheit, wenn ich den Speicherort gewechselt habe (bei **7 g**), dann verschwindet der «Daten löschen»-Button. Denn, in dem neuen Speicherort existiert ja dieses File noch gar nicht. Wenn zurückgewechselt wird, bleibt der Button verschwunden.

## 2.2.5.6 Regeln und Fehlermeldungen

Attribute	Regel	Fehlermeldung
«Metadaten_ID», «File-ID», «File-Titel»	Ein leerer Suchbegriff «» ist unzulässig. Darf also nicht leer gelassen werden.	
«Metadaten_ID», «File-ID», «File-Titel», «File-Autor(en)», «File-Besetzung», «Metadaten-Beschreibung»	Ein Komma «,» oder Doppelpunkt «:» im Suchbegriff ist unzulässig	

«File-Typ», «Grössen-einheit»	Alles ist verboten ausser den Eingaben, die in der Fehlermeldung angegeben werden.	 
«File-Grösse»	Alles ausser Zahlen mit einem Dezimalpunkt ist nicht zulässig.	
«File-Erstellungsda-tum»	Alles ausser einem gültigen Datum ist unzulässig.	
«File-Autor(en)», «File-Besetzung»	Es sind keine Zahlen zulässig.	

Tabelle 62: Regeln und Fehlermeldungen

Zum Schluss bleiben noch Fehlermeldungen, die mit den Speicherorten zusammenhängen. Es wird eine Fehlermeldung angezeigt, wenn keine Verbindung zur DB aufgebaut werden konnte oder wenn der USB-Stick, mit dem lokalen Ordner, vom Gerät getrennt ist. Diese Fehlermeldungen haben einen informativen Text, der das Problem erklärt und eine mögliche Lösung vorschlägt.

## 2.2.6 Testkonzept, Tests und Qualitätssicherung

### 2.2.6.1 Manuelle Testszenarien

Die nachfolgende Tabelle zeigt eine Aufstellung der geplanten Testszenarien. Dabei steht die erste Zahl für den Anwendungsfall, der getestet werden soll. Es wird folgende Nummerierung verwendet:

- 1 Neue Metadaten hinzufügen (mit Speichern)
- 2 Metadaten suchen
- 3 Metadaten bearbeiten und Speichern
- 4 Metadaten löschen
- 5 Verbindungsproblem

Test-ID:	Testszenarien:	Beschreibung:
1.1	Neue Metadaten in DB hinzufügen	Test vom Erstellen und Speichern eines neuen Metadaten-Files (DB).
1.2	Neue Metadaten im lokalen Ordner hinzufügen	Test vom Erstellen und Speichern eines neuen Metadaten-Files (Lokaler Ordner).
1.3	Falsche oder fehlende Eingabewerte	Wenn «Daten speichern» geklickt wird und es sind falsche oder fehlende Eingabewerte vorhanden, soll eine Fehlermeldung erscheinen, mit passendem Informationstext.
1.4	Ungültige Werte bei Speicherversuch	Die Daten dürfen nicht gespeichert werden, wenn noch ungültige Eingabewerte vorhanden sind.
1.5	Zurück / Abbruch	Wenn die Metadaten-Masken-Seite verlassen wird, egal ob über «zurück» oder «Abbrechen», so soll eine Bestätigungsabfrage erfolgen.
1.6	Speichermeldung	Nach erfolgreichem oder erfolglosem Speichern soll eine Rückmeldung erscheinen.
2.1	Erfolgreiche Suche	Test einer erfolgreichen Suche. Auch bei Teiltreffern.
2.2	Falscher oder fehlender Eingabewert.	Es soll eine Fehlermeldung erscheinen, mit passendem Informationstext.
2.3	Bei erfolgloser Suche...	Wenn kein Suchtreffer erfolgt ist, soll trotzdem das Suchfenster geöffnet werden und ein entsprechender Hinweis zu lesen sein.
2.4	Suchdauer	Die Suchdauer wird gemessen. (< 3 Sec wird erwartet)
3.1	Suchresultat in die Datenmaske laden	Wenn ein Suchresultat angeklickt wird, soll der Bearbeitungs-Button sichtbar werden. Ausser es gibt keine

		Suchresultate. Wird dieser angeklickt, werden die Daten korrekt in die Felder der Datenmaske geladen.
3.2	Alle Einträge ausser Metadaten-ID...	Es sollen alle Einträge veränderbar sein, ausser die Metadaten-ID.
3.3	Metadaten-ID-Anpassung	Wenn die File-ID oder/und der File-Titel geändert wurde, soll sich auch die Metadaten-ID ändern, nach dem Speichern.
3.4	Wechsel des Speicherorts	Wenn der Speicherort gewechselt werden soll, dann muss eine Informationsmeldung erscheinen, die den User über die Konsequenzen informiert. Das File soll am alten Speicherort gelöscht und am neuen erstellt werden.
4.1	Erfolgreiches Löschen	Test einer erfolgreichen Löschung.
4.2	Fehler beim Löschen	Wenn das Löschen nicht korrekt erfolgt, dann soll eine Fehlermeldung erscheinen.
4.3	Sicherheitsabfrage	Vor dem Löschen soll eine Sicherheitsabfrage erscheinen.
5.1	DB nicht erreichbar	Wenn die DB nicht erreichbar ist, so soll eine Fehlermeldung erscheinen. Das Programm muss weiter bedienbar sein.
5.2	Lokaler Ordner nicht erreichbar	Wenn der USB-Stick nicht erreichbar ist, so soll eine Fehlermeldung erscheinen. Das Programm muss weiter bedienbar

Tabelle 63: Testszenarien

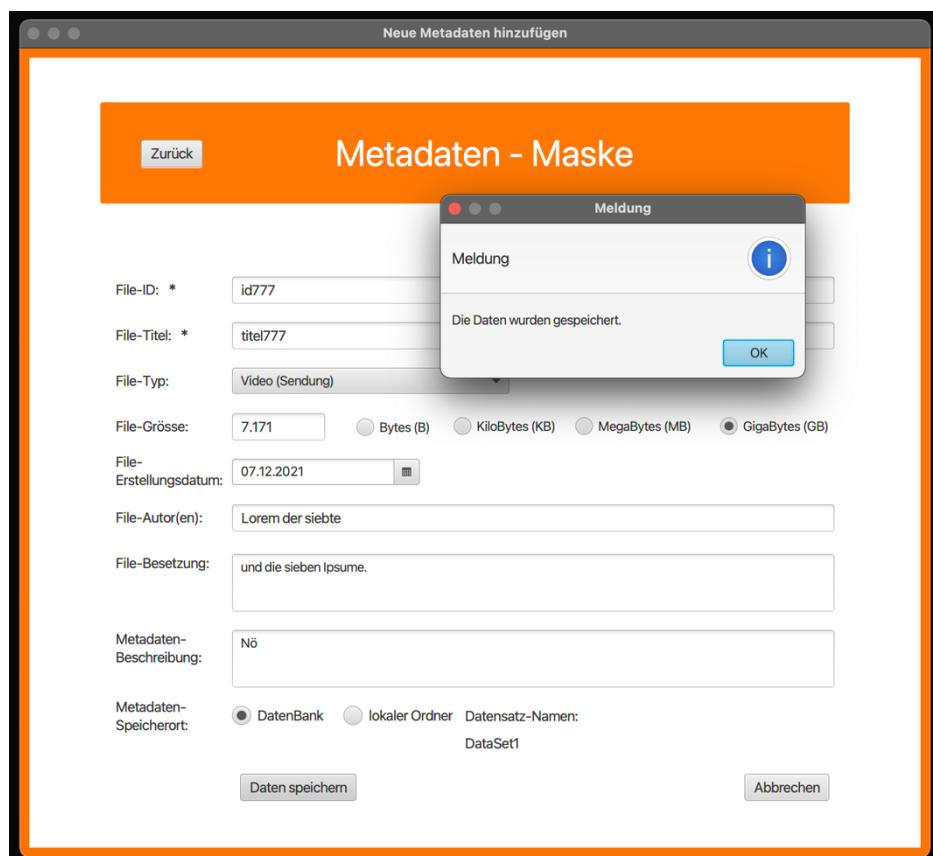
### 2.2.6.2 Testergebnisse

Diese dienen gleichzeitig auch als finale Abnahme.

Test-ID		1.1
<b>Testszenario</b>		Neue Metadaten in DB hinzufügen
<b>Testdatum</b>		20.12.2021
<b>Vorbedingung</b>		Die Applikation muss lauffähig sein und eine Internetverbindung steht.
<b>Ablauf</b>		<ul style="list-style-type: none"> <li>- Programmstart</li> <li>- Klick auf den «Neue Metadaten hinzufügen»-Button</li> <li>- Datenmaske ausfüllen mit gültigen Werten</li> <li>- Klick auf den «Daten speichern»-Button</li> </ul>
<b>Erwartetes</b>		Eine Meldung erscheint, die das Speichern bestätigt.
<b>Ergebnis</b>		In der MongoDB ist der neu-erstellte Eintrag zu sehen.

## Ist- Ergebnis

Die Speicher-Bestätigungsmeldung erscheint:



Der Eintrag ist in der MongoDB zu finden:

```

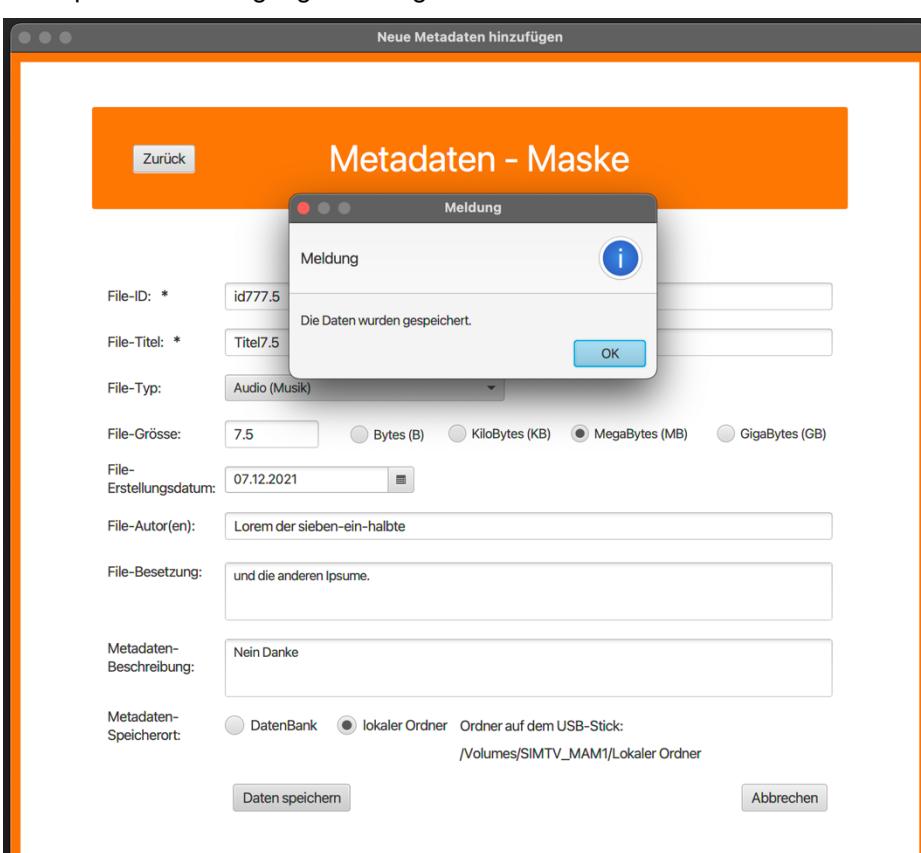
_id: ObjectId("61c198b843ed680052459910")
metadataID: "id777_titel777_858676"
fileID: "id777"
fileTitle: "titel777"
fileType: "VIDEO_BROADCAST"
fileSize: 7.171
fileSizeUnit: "GB"
fileDate: 2021-12-07T00:00:00.000+00:00
fileAuthor: "Lorem der siebte"
fileCast: "und die sieben Ipsume."
description: "Nö"
saveLocation: "DataSet1"

```

## Bemerkung(en)

JUnit-Test: «dataSaveTest()» der StrategyDBTest-Klasse wurde bestanden.

Tabelle 64: Testergebnis 1.1

<b>Test-ID</b>	1.2
<b>Testszenario</b>	Neue Metadaten im lokalen Ordner hinzufügen
<b>Testdatum</b>	20.12.2021
<b>Vorbedingung</b>	Die Applikation muss lauffähig sein und der USB-Stick mit dem Lokalen Ordner ist verbunden.
<b>Ablauf</b>	<ul style="list-style-type: none"> <li>- Programmstart</li> <li>- Klick auf den «Neue Metadaten hinzufügen»-Button</li> <li>- Datenmaske ausfüllen mit gültigen Werten</li> <li>- Klick auf den «Daten speichern»-Button</li> </ul>
<b>Erwartete Ergebnis</b>	<p>Eine Meldung erscheint, die das Speichern bestätigt.  Das Metadatenfile ist im lokalen Ordner zu sehen.</p>
<b>Ist- Ergebnis</b>	<p>Die Speicher-Bestätigungsmeldung erscheint:</p>  <p>Das Metadaten-File ist im lokalen Ordner zu finden:</p>

<b>Bemerkung(en)</b>	JUnit-Test: «dataSaveTest()» der StrategyFolderTest-Klasse wurde bestanden.

Tabelle 65: Testergebnis 1.2

Test-ID		1.3
<b>Testszenario</b>	Falsche oder fehlende Eingabewerte	
<b>Testdatum</b>	20.12.2021	
<b>Vorbedingung</b>	Die Seite «Metadaten-Maske» ist geöffnet.	
<b>Ablauf</b>	<ul style="list-style-type: none"> <li>- Felder mit ungültigen Werten ausfüllen:           <ul style="list-style-type: none"> <li>o Das «File-ID»- oder das «File-Titel»-Feld leer lassen,</li> <li>o Bei «File-Grösse» ein «hallo» eintippen,</li> <li>o Bei «File-Erstellungsdatum» ein «heute» eintippen,</li> <li>o Bei «File-Autor(en)» «Wir: ich, er, sie und es» eintippen.</li> </ul> </li> <li>- Klick auf «Daten speichern»</li> </ul>	
<b>Erwartete Ergebnis</b>	Es erscheint eine Fehlermeldung wegen dem leeren Pflichtfeld. Die anderen Probleme werden so lange ignoriert, bis die beiden Pflichtfelder ausgefüllt sind.	
<b>Ist- Ergebnis</b>	Es erscheint zuerst nur eine Fehlermeldung: die Pflichtfelder müssen ausgefüllt sein.	

Füllt man die beiden Pflichtfelder aus und klickt auf «Daten speichern», dann kommen drei Fehlermeldungen:

1. Die ungültige File-Grösse:

2. Das ungültige Trennzeichen:

3. Das ungültige Datum:

#### Bemerkung(en)

Bis auf den Metadaten-Speicherort spielt es keine Rolle, ob man von der Startseite über den «Neue Daten hinzufügen»-Button oder ob man über das Suchfenster mit dem «ausgewählte Metadaten bearbeiten»-Button auf die Metadaten-

	Masken-Seite gelangt. Die Kontrolle der Eingabefelder geschieht jedes Mal, wenn der Button «Daten speichern» geklickt wird.
--	---

Tabelle 66: Testergebnis 1.3

Test-ID	1.4
Testszenario	Ungültige Werte bei Speicherversuch
Testdatum	20.12.2021
Vorbedingung	Die Seite «Metadaten-Maske» ist geöffnet.
Ablauf	<ul style="list-style-type: none"> <li>- Beide Pflichtfelder ausfüllen</li> <li>- In die anderen Felder mit ungültigen Werten ausfüllen:           <ul style="list-style-type: none"> <li>o Entweder bei «File-Grösse» ein «hallo» eintippen,</li> <li>o Oder bei «File-Erstellungsdatum» ein «heute» eintippen,</li> <li>o Oder bei «File-Autor(en)» «Wir: ich, er, sie und es» eintippen.</li> <li>o Oder bei «File-Besetzung» «du: warst dabei» eintippen.</li> <li>o Oder bei «Metadaten-Beschreibung» «:-) » eintippen.</li> </ul> </li> </ul> <p>Nach jeder Änderung mehrmals (min 3-mal) auf «Daten speichern» klicken. Es soll immer eine der Fehlerhaften Eingaben verbleiben.</p>
Erwartete Ergebnis	Die Daten dürfen nicht gespeichert werden. Es soll immer eine Fehlermeldung erscheinen, solange das Problem besteht.
Ist- Ergebnis	Auch nach mehrfachem Versuch, die Daten fehlerhaft zu speichern, lässt es die Applikation nicht zu.
Bemerkung(en)	Siehe Bemerkung vom 1.3

Tabelle 67: Testergebnis 1.4

Test-ID	1.5
Testszenario	Zurück / Abbruch
Testdatum	20.12.2021
Vorbedingung	Die Seite «Metadaten-Maske» ist geöffnet.
Ablauf	<p>Entweder (Szenario 1a):</p> <ul style="list-style-type: none"> <li>- Die Pflichtfelder und einige weitere Felder korrekt ausfüllen.</li> <li>- Auf «<b>Abbrechen</b>» klicken.</li> </ul> <p>Oder (Szenario 1b):</p> <ul style="list-style-type: none"> <li>- Die Pflichtfelder und einige weitere Felder korrekt ausfüllen.</li> <li>- Auf «<b>zurück</b>» klicken.</li> </ul> <p>Oder (Szenario 2a):</p>

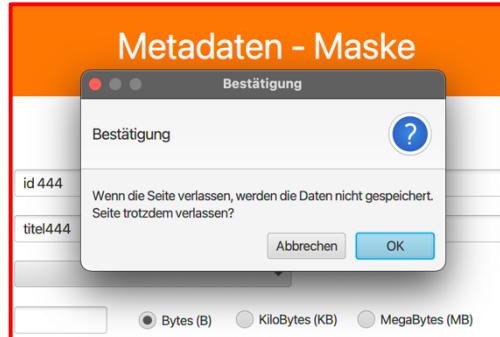
	<ul style="list-style-type: none"> <li>- Die Eingabefelder sind gefüllt mit einem Suchresultat.</li> <li>- Einige Suchresultate werden korrekt verändert.</li> <li>- Auf «<b>Abbrechen</b>» klicken.</li> </ul> <p>Oder (Szenario 2b):</p> <ul style="list-style-type: none"> <li>- Die Eingabefelder sind gefüllt mit einem Suchresultat.</li> <li>- Einige Suchresultate werden korrekt verändert.</li> <li>- Auf «<b>zurück</b>» klicken.</li> </ul>
<b>Erwartete Ergebnis</b>	<ul style="list-style-type: none"> <li>• Ein Fenster öffnet sich, das eine Bestätigung verlangt, dass die Seite, ohne Speichern verlassen wird.</li> <li>• Danach kommt man auf die Seite, von der man auf die Metadaten-Masken-Seite zugegriffen hat.</li> </ul>
<b>Ist- Ergebnis</b>	<p>Dieses Bestätigungsfenster öffnet sich in jedem Szenario, wie erwartet. Das Fenster, dass sich nach einem «Ok» öffnet, ist dasselbe, von dem auf die Metadaten-Masken-Seite zugegriffen wurde.</p> 
<b>Bemerkung(en)</b>	

Tabelle 68: Testergebnis 1.5

Test-ID	1.6
<b>Testszenario</b>	Speichermeldung
<b>Testdatum</b>	20.12.2021
<b>Vorbedingung</b>	Die Seite «Metadaten-Maske» ist geöffnet
<b>Ablauf</b>	<p>Entweder:</p> <ul style="list-style-type: none"> <li>- Gültige Daten werden in die Eingabefelder eingefüllt:           <ul style="list-style-type: none"> <li>o «File-ID» ⇒ «id111»</li> <li>o «File-Titel» ⇒ «Title111»</li> <li>o «File-Typ» ⇒ «Text»</li> <li>o «File-Grösse» ⇒ 1.1 KB</li> <li>o «File-Erstellungsdatum» ⇒ wähle im Picker das heutige Datum aus.</li> <li>o «Metadaten-Speicherort» ⇒ “lokaler Ordner” auswählen</li> </ul> </li> <li>- Klick auf «Daten speichern»</li> </ul> <p>Oder:</p>

	<ul style="list-style-type: none"> <li>- Ein Suchresultat laden</li> <li>- Der «File-Typ» ändern</li> <li>- Klick auf «Daten speichern»</li> </ul>
<b>Erwartete Ergebnis</b>	Eine Meldung, dass die Daten gespeichert wurden.
<b>Ist- Ergebnis</b>	Die Meldung erscheint. Bei beiden Szenarien. (Bild vom 1. Szenario)
<b>Bemerkung(en)</b>	-

Tabelle 69: Testergebnis 1.6

<b>Test-ID</b>	2.1
<b>Testszenario</b>	Erfolgreiche Suche
<b>Testdatum</b>	20.12.2021
<b>Vorbedingung</b>	Die Applikation muss lauffähig sein. Je nachdem welcher Ort durchsucht wird, muss eine Verbindung zu diesem Ort bestehen. Für den Test: zuerst DB dann der lokale Ordner.
<b>Ablauf</b>	Suchattribut auswählen ⇒ «File-ID» Suchbegriff «id» eintippen. Im ersten Durchgang die DB anwählen.

	<p>Klick auf «Suche starten». Im zweiten Durchgang den lokalen Ordner anwählen.</p>
<b>Erwartete Ergebnis</b>	<p>Die Suchfenster-Seite öffnet sich und eine Liste von Treffern erscheint. Auch die, die «id» beinhalten.</p>
<b>Ist- Ergebnis</b>	<p>Die Suchfenster-Seite erscheint und die Treffer werden angezeigt. Zuerst die Treffer in der DB:</p> <p>Dann die Treffer im Lokalen Ordner:</p>

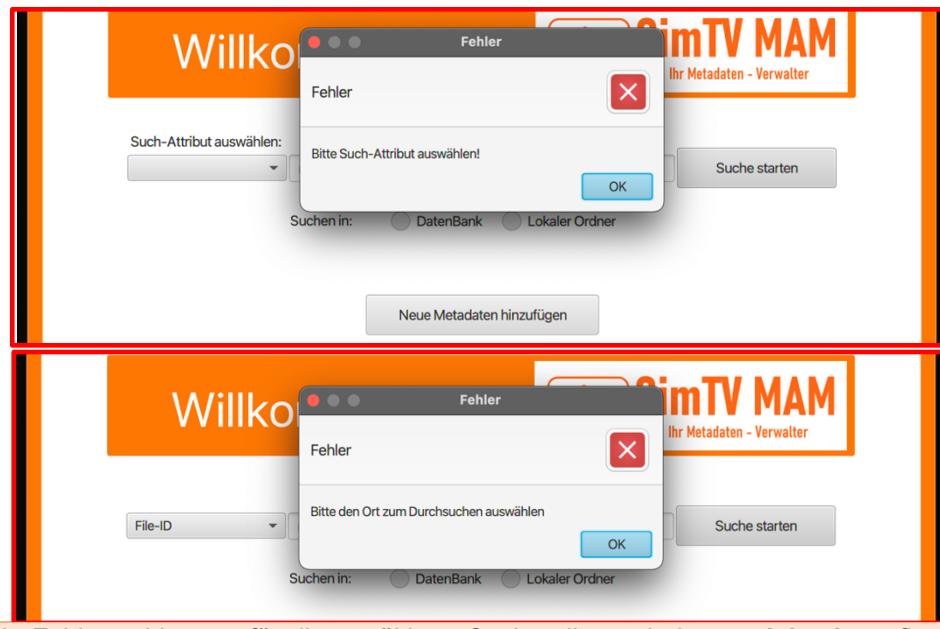
	<p>The screenshot shows a search interface titled "Suchfenster". It has a search bar with "File-ID" selected and "id" entered. Below it, there are search options: "DatenBank" (radio button) and "lokaler Ordner" (radio button). A message indicates the search path: "Ordner auf dem USB-Stick: /Volumes/SIMTV_MAM1/Lokaler Ordner". The results list several entries:</p> <ul style="list-style-type: none"> <li>[Metadaten-ID]: id1_T1_917722, [File-ID]: id1, [File-Titel]: T1, [File-Typ]: Audio (Musik), [File-Grösse]: 0.1 'B', [Erstellungsdatum]: 02.01.1970, [File-Autor(en)]: "", [File-Besetzung]: "", [Beschreibung]: "keine Beschreibung vorhanden.", [Speicherort]: /Volumes/SIMTV_MAM1/Lokaler Ordner/SaveFile_id1_T1_917722_.txt</li> <li>[Metadaten-ID]: id2_T2_170140, [File-ID]: id2, [File-Titel]: T2, [File-Typ]: Unbekannt, [File-Grösse]: 0.1 'B', [Erstellungsdatum]: 02.01.1970, [File-Autor(en)]: "", [File-Besetzung]: "", [Beschreibung]: "keine Beschreibung vorhanden.", [Speicherort]: /Volumes/SIMTV_MAM1/Lokaler Ordner/SaveFile_id2_T2_170140_.txt</li> <li>[Metadaten-ID]: id777_Titel7.5_5163, [File-ID]: id777, [File-Titel]: Titel7.5, [File-Typ]: Video (Film), [File-Grösse]: 7.5 'B', [Erstellungsdatum]: 01.12.2021, [File-Autor(en)]: "Lorem", [File-Besetzung]: "Ipsum", [Beschreibung]: "keine Beschreibung vorhanden.", [Speicherort]: /Volumes/SIMTV_MAM1/Lokaler Ordner/SaveFile_id777_Titel7.5_5163_.txt</li> <li>[Metadaten-ID]: id111_Titel11_986061, [File-ID]: id111, [File-Titel]: Titel111, [File-Typ]: Text, [File-Grösse]: 1.1 'KB', [Erstellungsdatum]: 20.12.2021, [File-Autor(en)]: "", [File-Besetzung]: "", [Beschreibung]: "keine Beschreibung vorhanden.", [Speicherort]: /Volumes/SIMTV_MAM1/Lokaler Ordner/SaveFile_id111_Titel111_986061_.txt</li> </ul>
Bemerkung(en)	JUnit-Test: «searchDataTest()» in der StrategyDBTest-Klasse wurde bestanden. «searchDataTest()» in der StrategyFolderTest -Klasse wurde ebenfalls bestanden.

Tabelle 70: Testergebnis 2.1

Test-ID	2.2
Testszenario	Falscher oder fehlender Eingabewert.
Testdatum	20.12.2021
Vorbedingung	Applikation ist gestartet.
Ablauf	<ul style="list-style-type: none"> <li>- Entweder wird das Suchattribut nicht ausgewählt,</li> <li>- Oder beim Suchattribut «Metadaten-ID» der Suchbegriff nicht eingegeben</li> <li>- Oder kein Ort zum Durchsuchen ausgewählt wurde.</li> <li>- Klick auf «Suche starten»</li> </ul>
Erwartete Ergebnis	Ist nichts ausgefüllt oder angeklickt, kommen die Fehlermeldungen, dass kein Suchattribut ausgewählt wurde und dass kein Ort zum Durchsuchen gewählt wurde. Ob der Suchbegriff zum Suchattribut passt, wird erst im zweiten Schritt geprüft.

### Ist- Ergebnis

Es erscheinen die beiden Fehlermeldungen:



### Bemerkung(en)

Die Fehlermeldungen für die ungültigen Suchattribute sind unter 2.2.5.6. zu finden.

Tabelle 71: Testergebnis 2.2

<b>Test-ID</b>	2.3
<b>Testszenario</b>	Bei erfolgloser Suche...
<b>Testdatum</b>	20.12.2021
<b>Vorbedingung</b>	Applikation ist gestartet.
<b>Ablauf</b>	<ul style="list-style-type: none"> <li>- Bei Suchattribut wird «File-Titel» angewählt.</li> <li>- Als Suchbegriff wird «SoEinLangerTitelExistiertBestimmtNicht» eingegeben</li> <li>- Der Lokale Ordner als Durchsuchungsort ausgewählt.</li> <li>- Klick auf «Suche starten»</li> </ul>
<b>Erwartete Ergebnis</b>	<p>Das Suchfenster öffnet sich.          Im unteren Feld steht «Keine Treffer gefunden»</p>
<b>Ist- Ergebnis</b>	Das Suchfenster öffnet sich und im unteren Feld steht «Keine Treffer gefunden».

	<input type="button" value="Zurück"/> <h3>Suchfenster</h3> <div style="background-color: #f0f0f0; padding: 5px; margin-bottom: 10px;"> <input type="button" value="File-Titel"/> <input type="text" value="SoEinLangerTitelExistiertBestimmtNicht"/> <input type="button" value="neue Suche starten"/> </div> <p>Suchen in: <input checked="" type="radio"/> DatenBank <input type="radio"/> lokaler Ordner</p> <p>Ordner auf dem USB-Stick: /Volumes/SIMTV_MAM1/Lokaler Ordner</p> <div style="border: 1px solid #ccc; height: 100px; margin-top: 10px;"></div> <p style="text-align: center;">Keine Treffer gefunden!</p>
<b>Bemerkung(en)</b>	-

Tabelle 72: Testergebnis 2.3

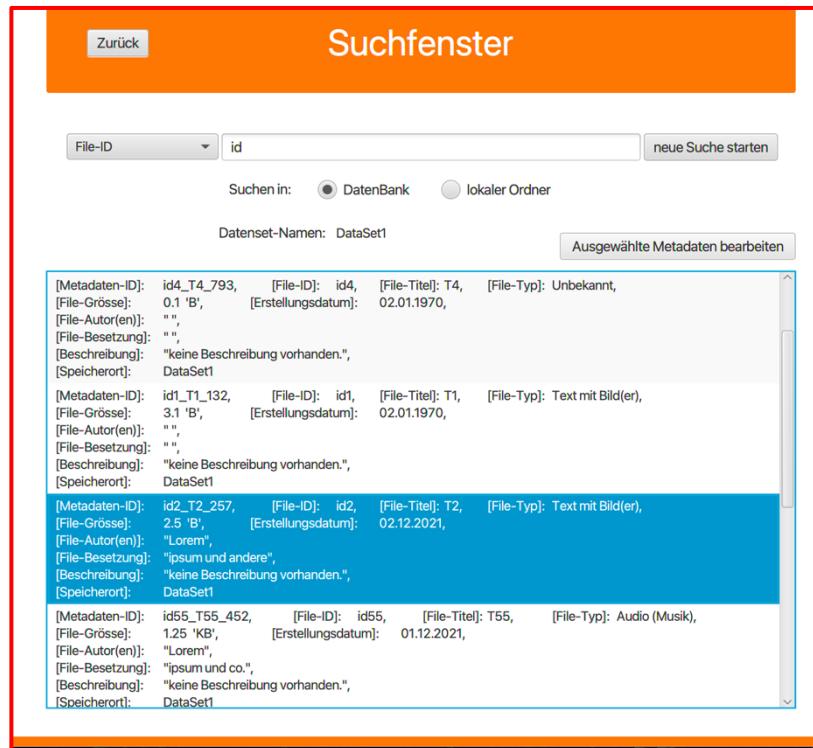
<b>Test-ID</b>		<b>2.4</b>
<b>Testszenario</b>	Suchdauer	
<b>Testdatum</b>	20.12.2021	
<b>Vorbedingung</b>	Eine Suche wie in 2.1 wird vorbereitet	
<b>Ablauf</b>	Die Zeit wird gestartet, sobald auf «Suche starten» geklickt wurde. Wenn die Resultate angezeigt werden, wird gestoppt.	
<b>Erwartete Ergebnisse</b>	Unter 3 Sekunden	
<b>Ist- Ergebnis</b>	Knapp 1 Sekunde	
<b>Bemerkung(en)</b>	2.1 bis 2.4 gilt auch, wenn die Suche vom Suchfenster aus gestartet wird.	

Tabelle 73: Testergebnis 2.4

<b>Test-ID</b>		<b>3.1</b>
<b>Testszenario</b>	Suchresultat in die Datenmaske laden	
<b>Testdatum</b>	20.12.2021	
<b>Vorbedingung</b>	Eine Suche wie in 2.1 wurde erfolgreich durchgeführt. Die Suchresultate sind im Suchfenster zu sehen.	
<b>Ablauf</b>	Ein Suchresultat wird angeklickt Klick auf «Ausgewählte Metadaten bearbeiten»	
<b>Erwartete Ergebnisse</b>	Die Metadaten-Masken-Seite wird geöffnet. Die Werte der Attribute stehen in den Eingabefeldern bis auf die Metadate-ID.	

## Ist- Ergebnis

Wenn ich das nachfolgende Suchresultat auswähle und auf «Ausgewählte Metadaten bearbeiten» klicke, dann erscheint die zweite Abbildung.



The screenshot shows a search interface titled "Suchfenster". At the top, there is a search bar with "File-ID" selected and the value "id". Below it are buttons for "neue Suche starten", "DatenBank" (selected), and "lokaler Ordner". A dataset name "DataSet1" is shown. A button "Ausgewählte Metadaten bearbeiten" is visible. The main area displays a list of metadata entries:

- [Metadaten-ID]: id4\_T4\_793, [File-ID]: id4, [File-Titel]: T4, [File-Typ]: Unbekannt,  
[File-Größe]: 0.1 'B', [Erstellungsdatum]: 02.01.1970,
- [Metadaten-ID]: id1\_T1\_132, [File-ID]: id1, [File-Titel]: T1, [File-Typ]: Text mit Bild(er),  
[File-Größe]: 3.1 'B', [Erstellungsdatum]: 02.01.1970,
- [Metadaten-ID]: id2\_T2\_257, [File-ID]: id2, [File-Titel]: T2, [File-Typ]: Text mit Bild(er),  
[File-Größe]: 2.5 'B', [Erstellungsdatum]: 02.12.2021,  
[File-Autor(en)]: "Lorem",  
[File-Besetzung]: "ipsum und andere",  
[Beschreibung]: "keine Beschreibung vorhanden.",  
[Speicherort]: DataSet1
- [Metadaten-ID]: id55\_T55\_452, [File-ID]: id55, [File-Titel]: T55, [File-Typ]: Audio (Musik),  
[File-Größe]: 1.25 'KB', [Erstellungsdatum]: 01.12.2021,  
[File-Autor(en)]: "Lorem",  
[File-Besetzung]: "ipsum und co.",  
[Beschreibung]: "keine Beschreibung vorhanden.",  
[Speicherort]: DataSet1



The screenshot shows an edit form titled "Metadaten - Maske". It contains fields for metadata entry "id2\_T2\_257":

- File-ID: \*
- File-Titel: \*
- File-Typ:
- File-Größe:   Bytes (B)  KiloBytes (KB)  MegaBytes (MB)  GigaBytes (GB)
- File-Erstellungsdatum:
- File-Autor(en):
- File-Besetzung:
- Metadaten-Beschreibung:

At the bottom, there are buttons for "Daten speichern", "Daten löschen", and "Abbrechen". There are also radio buttons for "DatenBank" (selected) and "lokaler Ordner", and a dropdown for "Datensatz-Namen: DataSet1".

## Bemerkung(en)

-

Tabelle 74: Testergebnis 3.1

Test-ID	3.2
Testszenario	Alle Einträge ausser Metadaten-ID...
Testdatum	20.12.2021
Vorbedingung	Daten sind von den Suchresultaten in die Metadaten-Maske geladen worden.
Ablauf	Alle Daten können verändert werden, ausser die Anzeige der Metadaten-ID.
Erwartete	Es lassen sich alle Daten verändern.
Ergebnis	Bei der Änderung des Speicherortes wird eine Bestätigungsmeldung angezeigt.
Ist- Ergebnis	Es lassen sich alle Daten ändern und bei der Änderung des Speicherortes wird die Bestätigungsmeldung angezeigt.
Bemerkung(en)	-

Tabelle 75: Testergebnis 3.2

Test-ID	3.3
Testszenario	Metadaten-ID-Anpassung
Testdatum	20.12.2021
Vorbedingung	3.1 wurde durchgeführt
Ablauf	Die «File-ID» und der «File-Titel» werden beide mit der Eingabe «halloTest» erweitert. Klick auf «Daten speichern»
Erwartete	Die Daten lassen sich Speichern und es wird das Suchfenster geöffnet.
Ergebnis	Man sieht unter anderem das geänderte File, welches jetzt eine geänderte Metadaten-ID besitzt.
Ist- Ergebnis	Die Speichermeldung wurde angezeigt. Das File hat eine geänderte Metadaten-ID. Der unterste Eintrag hat jetzt statt der Metadaten-ID: «id2_T2_257» neu die Metadaten-ID «id2halloTest_T2halloTest_257».

<b>Bemerkung(en)</b>	<p>JUnit-Test: «updateDataTest()» in der StrategyDBTest-Klasse wurde bestanden.</p> <p>«updateDataTest()» in der StrategyFolderTest -Klasse wurde ebenfalls bestanden.</p>

Tabelle 76: Testergebnis 3.3

Test-ID	3.4
<b>Testszenario</b>	Wechsel des Speicherorts
<b>Testdatum</b>	20.12.2021
<b>Vorbedingung</b>	Daten sind von den Suchresultaten in die Metadaten-Maske geladen worden.
<b>Ablauf</b>	Der Speicherort wird geändert.
<b>Erwartete</b>	Eine Bestätigungsmeldung erscheint.
<b>Ergebnis</b>	Bei «Ok» ist das File im alten Speicherort nicht mehr zu finden.
<b>Ist- Ergebnis</b>	Die Bestätigungsmeldung erscheint und das File wechselt seinen Speicherort.  Hier zuerst vor dem Wechsel:

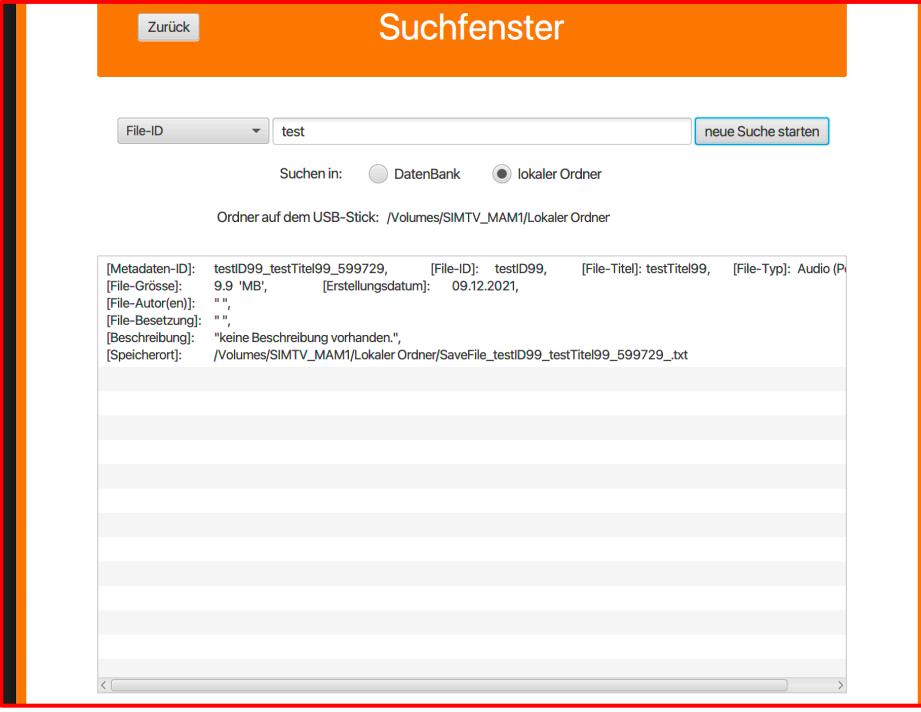
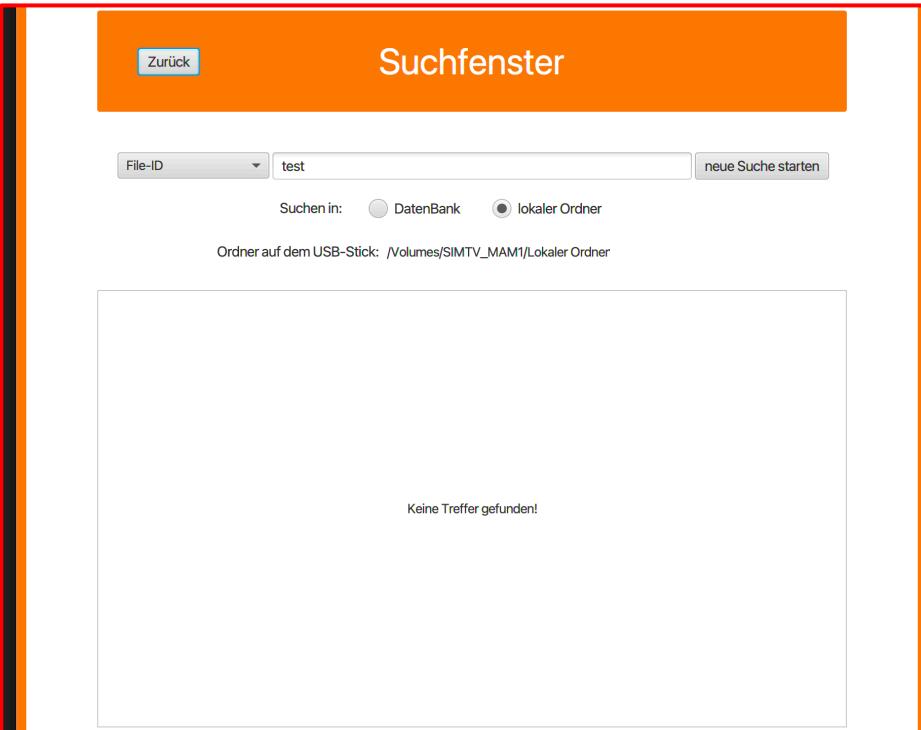
	 <p>The screenshot shows a search interface titled "Suchfenster". The search bar contains "test". Below it, there are search options: "Suchen in: DatenBank" (radio button) and "lokaler Ordner" (radio button). A message indicates "Ordner auf dem USB-Stick: /Volumes/SIMTV_MAM1/Lokaler Ordner". The main area displays a table of search results:</p> <table border="1"> <tbody> <tr> <td>[Metadaten-ID]:</td><td>testID99_testTitel99_599729,</td><td>[File-ID]:</td><td>testID99,</td><td>[File-Titel]:</td><td>testTitel99,</td><td>[File-Typ]:</td><td>Audio (P)</td></tr> <tr> <td>[File-Größe]:</td><td>9.9 'MB',</td><td>[Erstellungsdatum]:</td><td>09.12.2021,</td><td></td><td></td><td></td><td></td></tr> <tr> <td>[File-Autor(en)]:</td><td>" "</td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr> <td>[File-Besetzung]:</td><td>" "</td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr> <td>[Beschreibung]:</td><td>"keine Beschreibung vorhanden."</td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr> <td>[Speicherort]:</td><td>/Volumes/SIMTV_MAM1/Lokaler Ordner/SaveFile_testID99_testTitel99_599729_.txt</td><td></td><td></td><td></td><td></td><td></td><td></td></tr> </tbody> </table>	[Metadaten-ID]:	testID99_testTitel99_599729,	[File-ID]:	testID99,	[File-Titel]:	testTitel99,	[File-Typ]:	Audio (P)	[File-Größe]:	9.9 'MB',	[Erstellungsdatum]:	09.12.2021,					[File-Autor(en)]:	" "							[File-Besetzung]:	" "							[Beschreibung]:	"keine Beschreibung vorhanden."							[Speicherort]:	/Volumes/SIMTV_MAM1/Lokaler Ordner/SaveFile_testID99_testTitel99_599729_.txt						
[Metadaten-ID]:	testID99_testTitel99_599729,	[File-ID]:	testID99,	[File-Titel]:	testTitel99,	[File-Typ]:	Audio (P)																																										
[File-Größe]:	9.9 'MB',	[Erstellungsdatum]:	09.12.2021,																																														
[File-Autor(en)]:	" "																																																
[File-Besetzung]:	" "																																																
[Beschreibung]:	"keine Beschreibung vorhanden."																																																
[Speicherort]:	/Volumes/SIMTV_MAM1/Lokaler Ordner/SaveFile_testID99_testTitel99_599729_.txt																																																
	<p>Und nach dem Wechsel:</p>  <p>The screenshot shows a search interface titled "Suchfenster". The search bar contains "test". Below it, there are search options: "Suchen in: DatenBank" (radio button) and "lokaler Ordner" (radio button). A message indicates "Ordner auf dem USB-Stick: /Volumes/SIMTV_MAM1/Lokaler Ordner". The main area displays a message: "Keine Treffer gefunden!" (No results found!).</p>																																																
	<p>Dafür erscheint das Metadaten-File beim anderen Speicherort.</p>																																																
Bemerkung(en)	-																																																

Tabelle 77: Testergebnis 3.4

Test-ID	4.1
Testszenario	Erfolgreiches Löschen
Testdatum	20.12.2021
Vorbedingung	Ein Suchresultat-Datensatz wurde in die Metadaten-Maske gefüllt, wie bei 3.1.
Ablauf	Klick auf «Daten löschen»
Erwartete Ergebnis	Es erscheint ein Bestätigungsfenster, dass bei «Ok» die Daten löscht und das Suchfenster öffnet. Bei «Abbrechen» werden die Daten nicht gelöscht.
Ist- Ergebnis	Das Bestätigungsfenster erscheint. Bei «Ok» wird der Datensatz gelöscht und das Suchfenster erscheint, bei «Abbruch» geht die Meldung weg und es verändert sich nichts.
Bemerkung(en)	JUnit-Test: «deleteDataTest()» in der StrategyDBTest-Klasse wurde bestanden. «deleteDataTest()» in der StrategyFolderTest -Klasse wurde ebenfalls bestanden.

Tabelle 78: Testergebnis 4.1

Test-ID	4.2
Testszenario	Fehler beim Löschen
Testdatum	20.12.2021
Vorbedingung	Ein Suchresultat-Datensatz wurde in die Metadaten-Maske gefüllt. Wichtig: der Datensatz muss von dem lokalen Ordner sein.
Ablauf	Klick auf «Daten löschen» Wenn das Bestätigungsfenster erscheint, dann von Hand die TXT-Datei, die geladen wurde, löschen. Dann auf «Ok» klicken.
Erwartete Ergebnis	Eine Fehlermeldung soll angezeigt werden, dass die Daten nicht konnte gelöscht werden.
Ist- Ergebnis	Die Fehlermeldung erscheint:

<b>Bemerkung(en)</b>	-

Tabelle 79: Testergebnis 4.2

Test-ID		5.1
<b>Testszenario</b>	DB nicht erreichbar	
<b>Testdatum</b>	20.12.2021	
<b>Vorbedingung</b>	Applikation läuft	
<b>Ablauf</b>	Die Verbindung zum Internet trennen Eine Suche in der DB starten	
<b>Erwartete Ergebnis</b>	Eine Fehlermeldung erscheint, dass die Verbindung zur MongoDB nicht hergestellt werden konnte.	
<b>Ist- Ergebnis</b>	Die Fehlermeldung erscheint: 	

Bemerkung(en)	-
---------------	---

Tabelle 80: Testergebnis 5.1

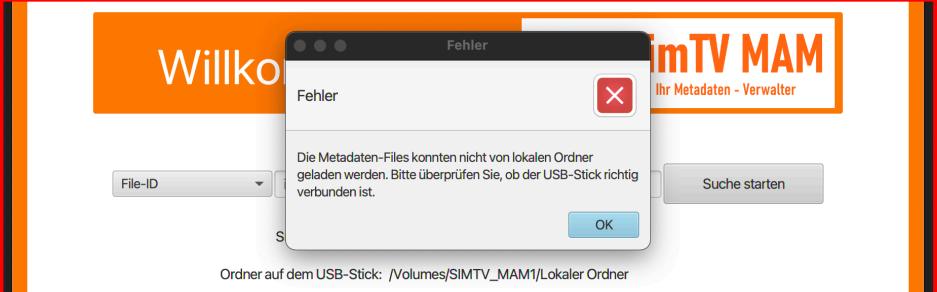
Test-ID	5.2
Testszenario	Lokaler Ordner nicht erreichbar
Testdatum	20.12.2021
Vorbedingung	Applikation läuft
Ablauf	Der USB-Stick auswerfen oder anderweitig die Verbindung trennen. Eine Suche im lokalen Ordner starten
Erwartete Ergebnis	Eine Fehlermeldung, dass der USB-Stick nicht gefunden wurde.
Ist- Ergebnis	Die Fehlermeldung erscheint: 
Bemerkung(en)	-

Tabelle 81: Testergebnis 5.2

### 2.2.6.3 JUnit-Tests

Ich habe für diese Arbeit JUnit5 benutzt. Meine Testklassen prüfen zum Einen Teile der Logik in der Model-Klasse und zum Anderen die Methoden der StrategyDB- und StrategyFolder-Klassen, damit der Datenaustausch mit den Speicherorten gewährleiste wird und getestet werden kann.

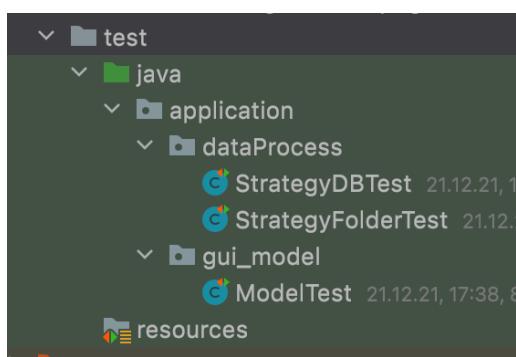


Abbildung 49: Meine Test-Klassen

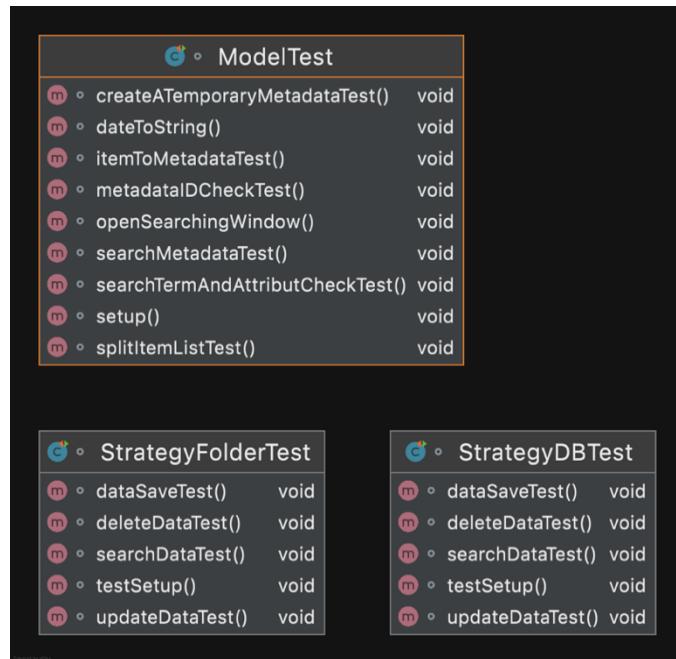


Abbildung 50: Test-Klassen samt Methoden

Alle Tests wurden bestanden:

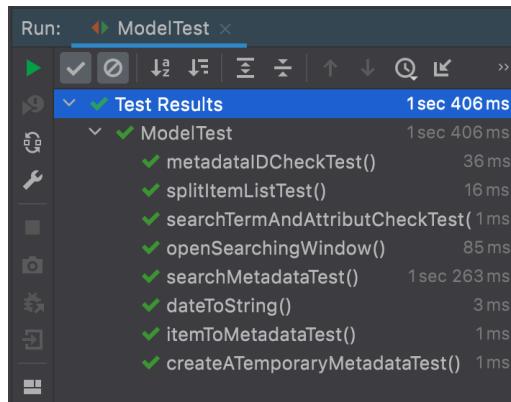


Abbildung 51: bestandene ModelTest-Tests.

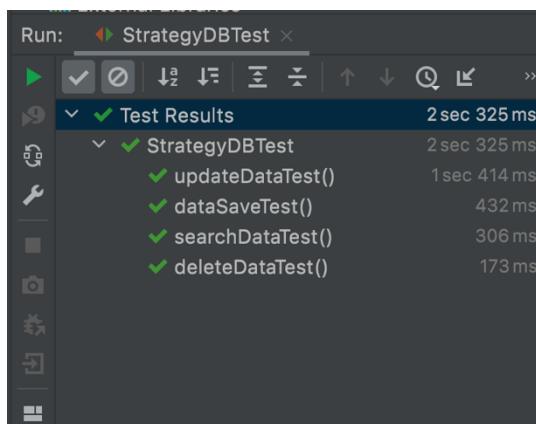


Abbildung 52: bestandene StrategyDBTest-Tests.

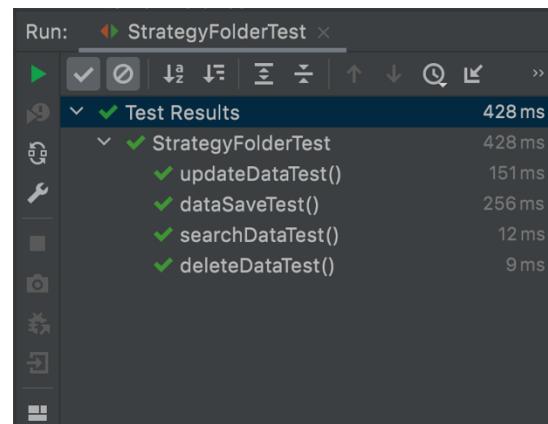


Abbildung 53: bestandene StrategyFolderTest-Tests.

## 2.3 Inbetriebnahme

### 2.3.1 Lieferobjekt

Das Lieferobjekt umfasst die gebundene Dokumentation dieser Projektarbeit und einen USB-Stick. Auf dem USB-Stick ist eine digitale Version dieser Dokumentation als PDF-File, der Abschnitt «Installationsanleitung» in einem separaten PDF-File und einen Ordner mit dem Namen «Lokaler Ordner» indem sich TXT-Dateien befinden. Das sind Testdateien, mit denen die Dozenten die Applikation testen können.

### 2.3.2 Installationsanleitung

#### 2.3.2.1 Voraussetzungen

Damit die «SimTV MAM»-Applikation gestartet werden kann, müssen zwei Bedingungen erfüllt sein:

- Das Programm «IntelliJ IDEA» von JetBrains muss installiert und auf dem neusten Stand sein.
- Das Java SE Development Kit 17.0.1, kurz jdk17, muss auf dem Endgerät, auf dem die Applikation gestartet wird, installiert sein.

#### 2.3.2.2 Schritt 1: Projekt via Git-Repository erstellen

- 1.1. Starten Sie das Programm «IntelliJ IDEA».
- 1.2. Klicken Sie auf «Get from VCS»-Button in der oberen rechten Fensterecke

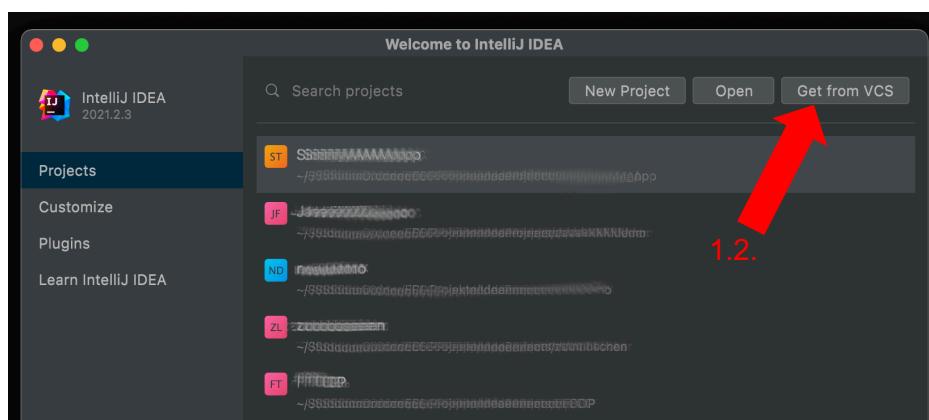


Abbildung 54: IntelliJ Startseite

- 1.3. Geben Sie als URL «[https://git.ffhs.ch/simon.wittker/simtv\\_mam\\_app](https://git.ffhs.ch/simon.wittker/simtv_mam_app)» ein.
- 1.4. Klicken Sie auf «Clone» in der rechten unteren Fensterecke.

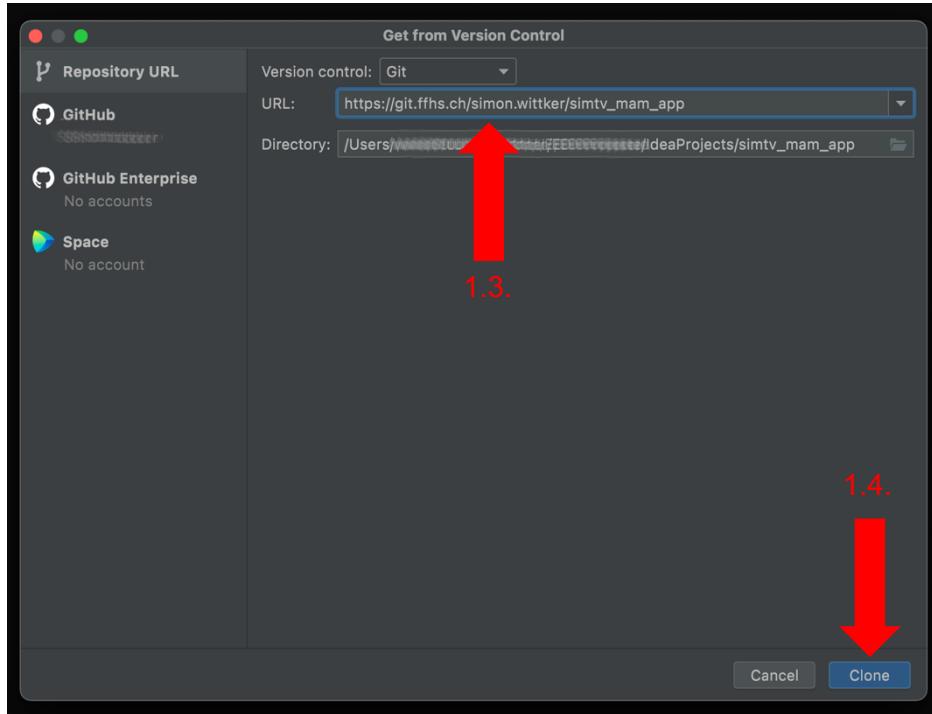


Abbildung 55: Git-Repo einfügen und klonen

- 1.5. Warten Sie einen Moment, bis das Projekt angelegt ist.
- 1.6. Sobald das Projekt angelegt und initiiert ist, erscheint entweder ein Fenster mit dem Titel «Trust Maven Projekt?» oder ein blauer Streifen erscheint über dem geöffneten «README»-File.
- 1.7. Theoretisch sollte die Applikation auch im «Safe Mode» laufen, aber eine Garantie gibt es nur mit einem Klick auf «Trust Projekt».

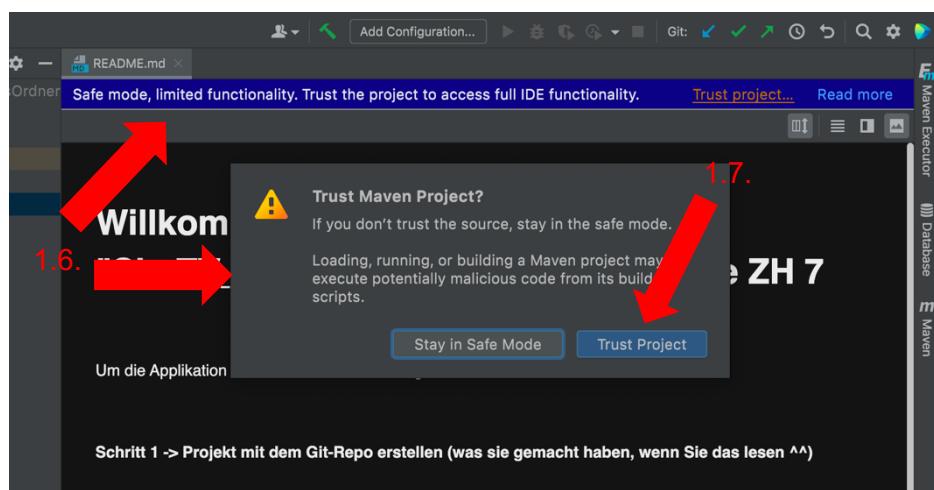


Abbildung 56: Trust Maven Project? - Fenster

- 1.8. In dem «README»-file sind die nächsten Schritte noch einmal aufgeschrieben, zur Hilfe.

### 2.3.2.3 Schritt 2: Java-Version einstellen

- 2.1. Klicken Sie auf den Reiter "File"
- 2.2. Wählen Sie "Project Structure" aus



Abbildung 57: «Project Structure...» öffnen

- 2.3. Unter der Kategorie "Project Settings" das "Project" auswählen
- 2.4. Im Abschnitt "Project SDK:" das "jdk 17.0.1" auswählen  
(Gegebenenfalls muss via "Edit" -> "+" das jdk 17 noch hinzugefügt werden...)
- 2.5. Unter der Kategorie "Project language level:" auch die 17 auswählen

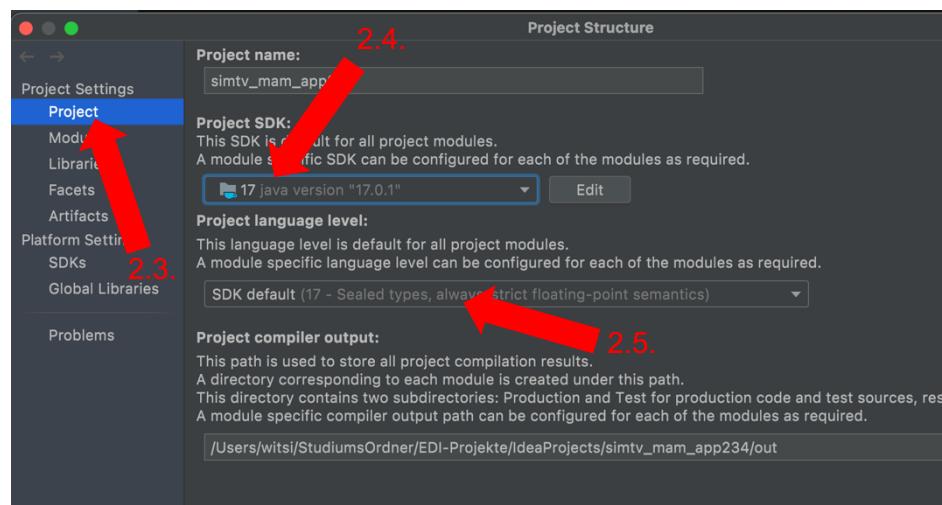


Abbildung 58: Das "Project Structure"-Fenster

- 2.6. Und mit OK das Fenster verlassen.

### 2.3.2.4 Schritt 3: Die Startklasse bestimmen

- 3.1. Klicken Sie auf den Reiter "Run"
- 3.2. Wählen Sie "Edit Configurations..." aus

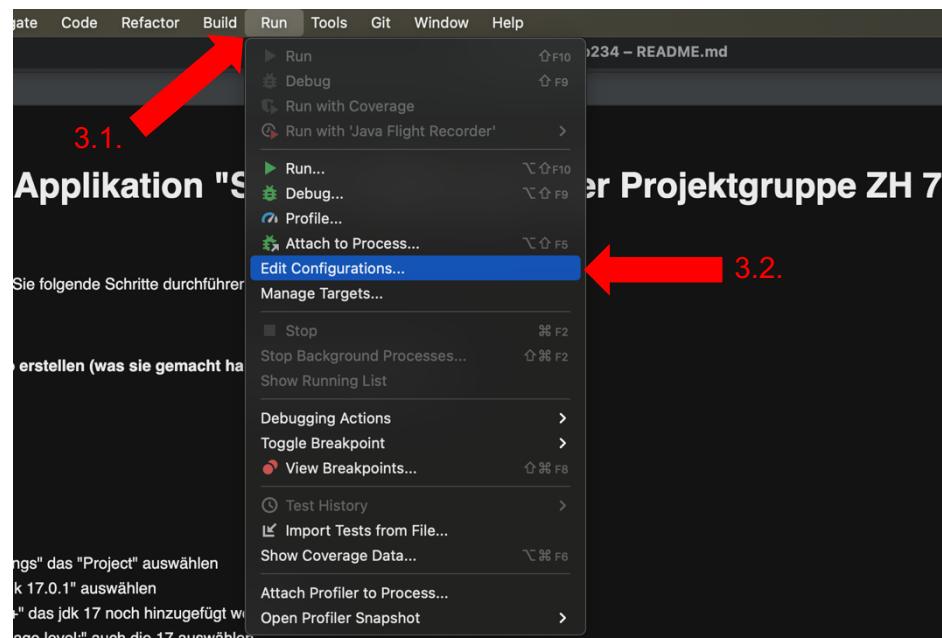


Abbildung 59: "Edit Config..." öffnen

- 3.3. Klicken Sie auf das "+" im oberen linken Fensterrand
- 3.4. Wählen Sie "Applikation" aus

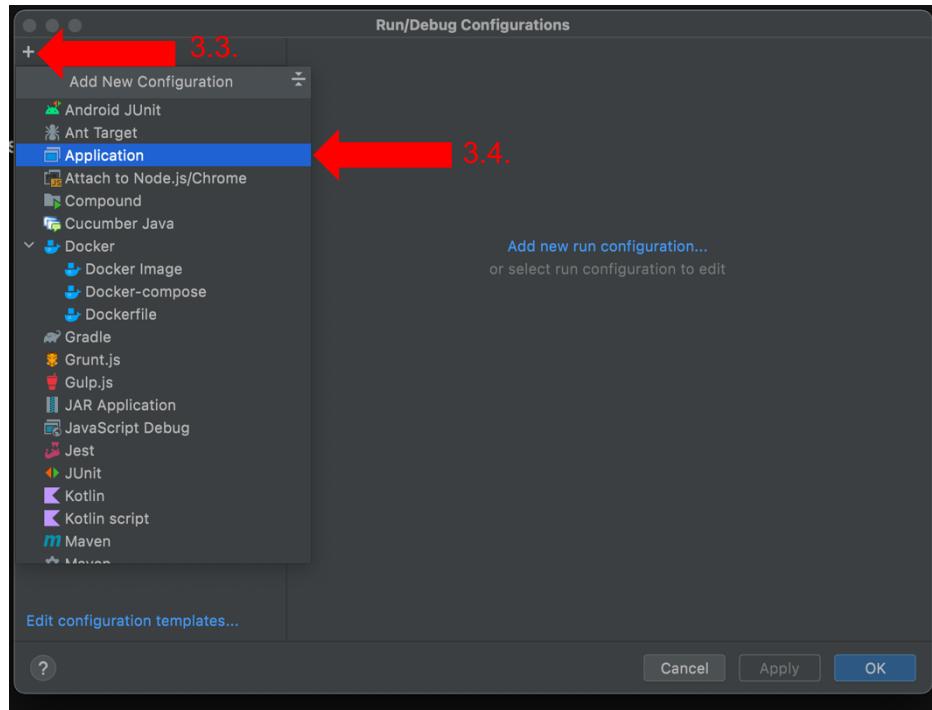


Abbildung 60: neue Applikation initiieren

- 3.5. Schreiben Sie "Start" in das Namensfeld

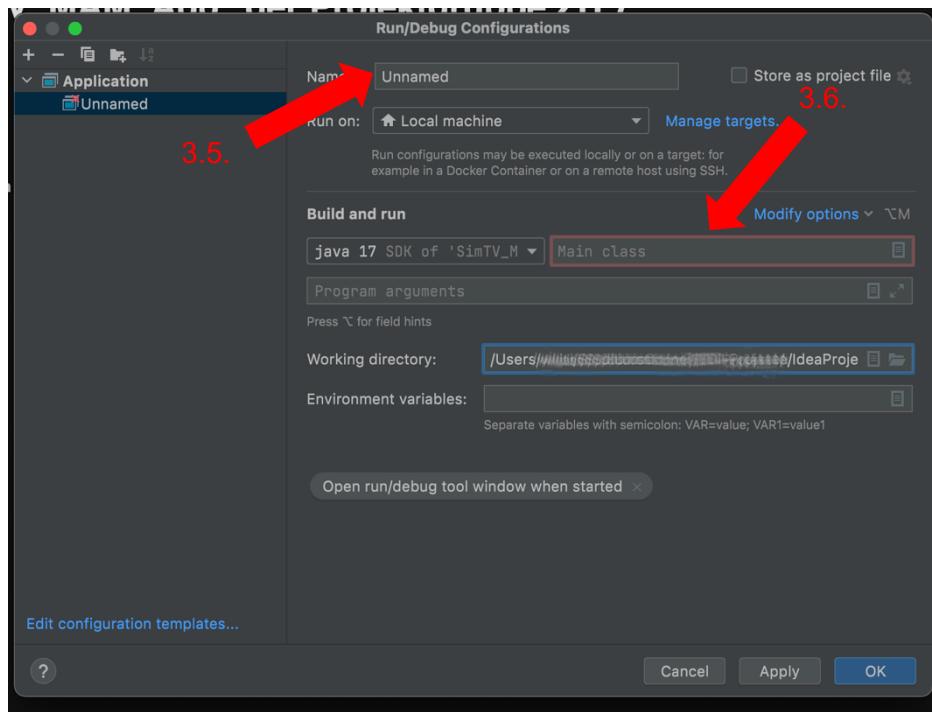


Abbildung 61: Config. ausfüllen

- 3.6. Schreiben Sie "project.simtv\_mam\_app.Start" in das "Main class"-Feld
- 3.7. (Wenn es so aussieht wie auf Abbildung 62) Mit OK bestätigen

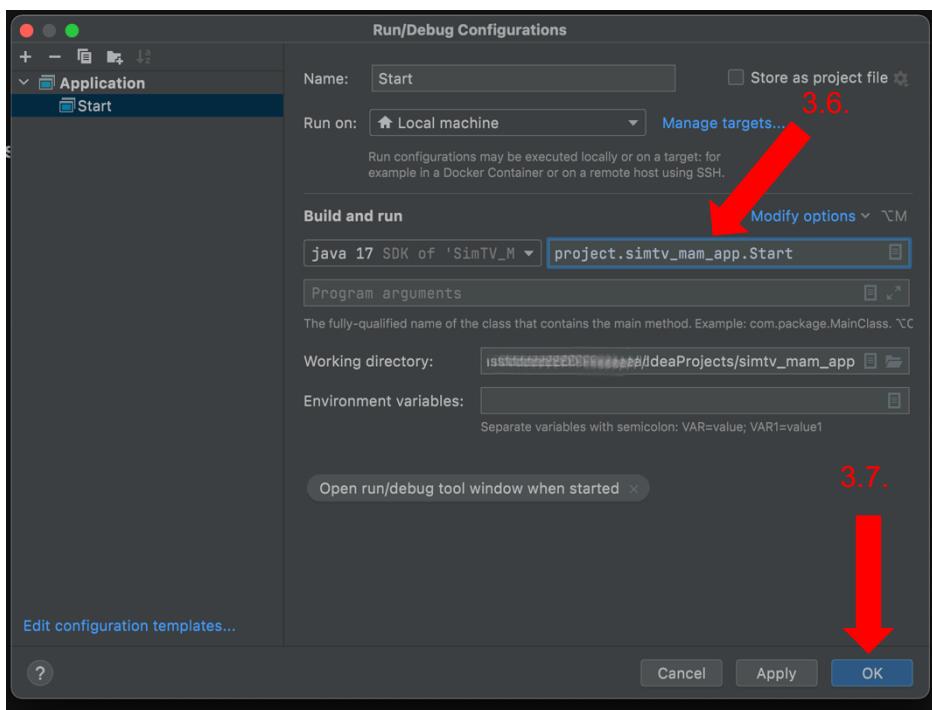


Abbildung 62: fertig ausgefülltes Config.-Fenster

## 2.3.2.5 Schritt 4: Starten!

- 4.1. Die Applikation starten mit Alt+Shift+F10 für Windows-User oder Shift+F10 für Mac-User oder mit Klick auf das grüne-Start-Symbol.

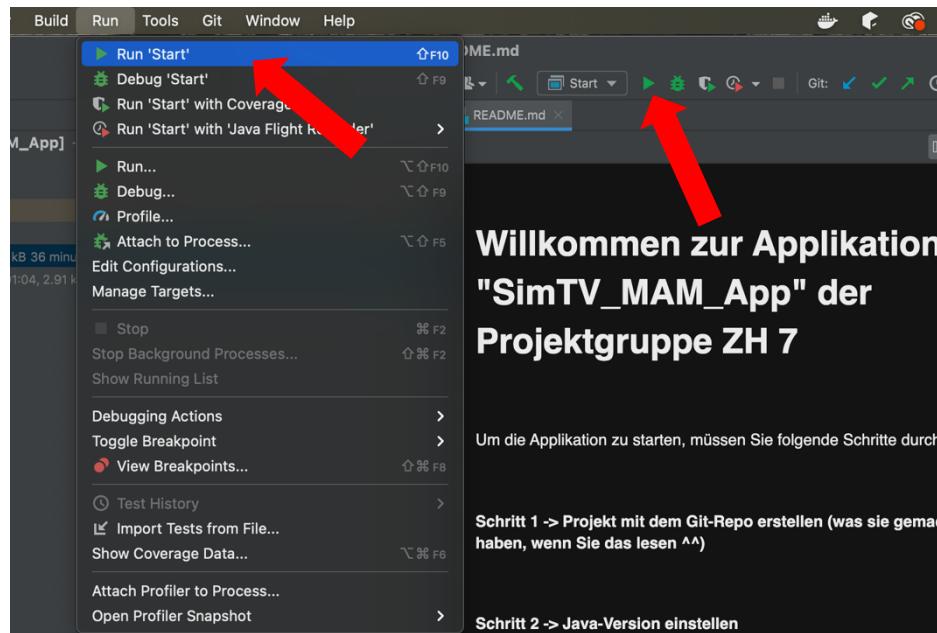


Abbildung 63: Applikation starten

- 4.2. Nach einer kurzen Zeit, in der die IDEA die benötigten Komponenten zusammenstellt, erscheint das Startfenster der Applikation. Herzlichen Glückwunsch.

## 2.3.3 Benutzerhandbuch

### 2.3.3.1 Wozu ist die Applikation?

Der Sinn und Zweck der SimTV MAM-Applikation ist das Speichern und Verwalten von Metadaten. Denn mithilfe der Metadaten können Media-Assets, wie Podcasts oder Folgen einer Fernsehserie, platzsparend und effizient gespeichert und Verwalten werden, während die grossen Media-Asset-Files separat gelagert werden. Mithilfe eines Metadaten-Verwaltungssystems wie dem SimTV MAM ist es in der Zukunft möglich, Sendungen mit Metadaten zu planen und erst am Schluss die dazugehörigen Media-Asset-Files aus dem Speicher zu laden, wenn sie gebraucht werden.

### 2.3.3.2 Start der Applikation

Dem momentanen Zustand der Applikation geschuldet, via «IntelliJ IDEA» und «Run».

### 2.3.3.3 Erste Anwendung, ein neues Metadaten-File anlegen

Wenn die Applikation gestartet ist, Empfängt sie Sie mit der Startseite. Machen wir die ersten Schritte, indem wir ein neues Metadaten-File anlegen. Klicken Sie dazu auf den Button «Neue Metadaten hinzufügen» (siehe der rote Pfeil in der Abbildung 64).

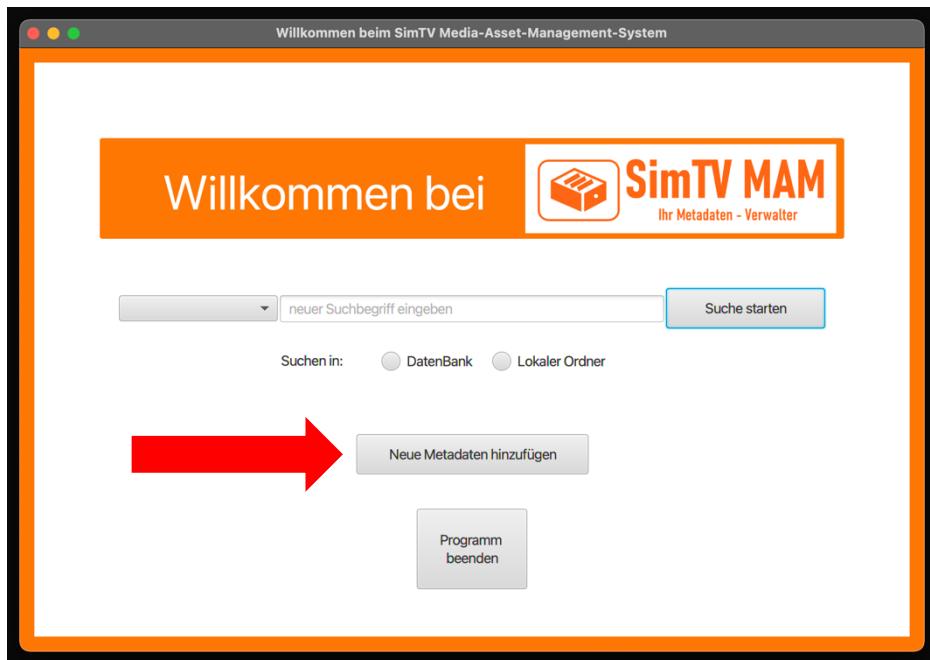


Abbildung 64: Startseite

Durch den Klick auf den Button öffnet sich die Seite mit dem Titel «Metadaten-Maske». Zu sehen in der Abbildung 65. Auf dieser Seite können alle Metadaten-Werte eingetragen werden. In der folgenden Tabelle ist alles Wissenswerte über die einzelnen Einträge zu lesen. Alle Einträge mit einem «File» davor, beziehen sich auf das Media-Asset-File (MAF), dessen Metadaten gespeichert werden.

Attribut	Beschreibung	Erwartete Eingabewerte (Default-Wert DW)	Unzulässige Eingabewerte
<b>File-ID</b>	Die ID des Media-Asset-Files	Buchstaben oder Zahlen (kein Default-Wert, ist ein Pflichtfeld)	Komma, Doppelpunkt oder Leer-lassen
<b>File-Titel</b>	Der Titel des MAF	Buchstaben oder Zahlen (kein Default-Wert, ist ein Pflichtfeld)	Komma, Doppelpunkt oder Leer-lassen
<b>File-Typ</b>	Der Typ des MAF	Einen aus der aufklappbaren Liste aus-suchen (DW: Unbekannt)	Keine falsche Eingabe möglich.
<b>File-Grösse</b>	Die File-Grösse des MAF	Nur Zahlen mit einem Dezimalpunkt (DW: 0.1)	Komma, Doppelpunkt und alles ausser Zahlen
<b>Grössenein-heit</b>	Die Einheit der File-Grösse	Einen der vier RadioButton auswählen (DW: B)	Keine falsche Eingabe möglich.

<b>File-Erstellungsdatum</b>	Das Datum, wann das MAF erstellt wurde	Entweder ein Datum mit dd.mm.yyyy oder es wird eines im Kalender ausgewählt, der durch Anklicken des grauen Buttons rechts von Eingabefeld sichtbar wird. (DW: 01.01.1970)	Buchstaben und Zeichen, ausser den Punkt. Es sind nur Zahlen und Punkte erlaubt.
<b>File-Autor(en)</b>	Der Ersteller des MAF	Buchstaben, Punkte und Semikolons (;) sind erlaubt (DV: « »)	Zahlen, Kommas und Doppelpunkte.
<b>File-Besetzung</b>	Der Cast des MAF	Buchstaben, Satzzeichen wie Punkte und Semikolons (;) sind erlaubt (DV: « »)	Zahlen, Kommas und Doppelpunkte.
<b>Metadaten-Beschreibung</b>	Beschreibung dieser Metadaten	Buchstaben, Zahlen, Satzzeichen wie Punkte und Semikolons (;) sind erlaubt (DV: « »)	Kommas und Doppelpunkte.
<b>Metadaten-Speicherort</b>	Speicherort des Metadaten-Files	Entweder «Datenbank» oder im «lokalen Ordner» (kein DK-Wert)	Nichts von den beiden auszuwählen.
<b>Metadaten-ID (verborgen)</b>	Die eindeutige ID dieses Metadaten-Files	Wird vom Programm erstellt. Besteht aus Buchstaben und Zahlen.	Komma oder Doppelpunkte.

Tabelle 82: Attributtypen-Tabelle

The screenshot shows a Mac OS X style application window titled "Neue Metadaten hinzufügen". The main title bar says "Metadaten - Maske". Inside, there's a header "Metadaten - Maske" with a "Zurück" button. Below it, a note "## Pflichtfelder ##" is followed by several input fields:

- File-ID: \* (text input field)
- File-Titel: \* (text input field)
- File-Typ: (dropdown menu)
- File-Grösse: (text input field with radio buttons for Bytes (B), KiloBytes (KB), MegaBytes (MB), and GigaBytes (GB))
- File-Erstellungsdatum: (date picker input field)
- File-Autor(en): (text input field)
- File-Besetzung: (text input field)
- Metadaten-Beschreibung: (text input field)
- Metadaten-Speicherort: (radio buttons for DatenBank and lokaler Ordner)

At the bottom are two buttons: "Daten speichern" (Save Data) and "Abbrechen" (Cancel).

Abbildung 65: Metadaten-Masken-Seite

Die Minimal-Eingaben, um ein Metadaten-File zu erstellen sind die File-ID, der File-Titel und den Speicherort, den man wählen kann. Für die Speicherung in der Datenbank braucht es allerdings eine Internetverbindung. Der lokale Ordner funktioniert auch ohne Internet, dafür muss aber der USB-Stick, auf dem die Applikation gespeichert war, mit dem Gerät verbunden sein.

Haben Sie alle Eingaben gemacht, die Sie wollten, dann können Sie ihre Daten speichern mit einem Klick auf «Daten speichern». Dabei werden die Eingaben kontrolliert, ob einen Regelverstoss zu finden ist oder nicht. Wenn Ja, dann wird eine Fehlermeldung angezeigt, mit einem kurzen Hinweis-Text, welcher Regelverstoss vorliegt. Wenn Nein, werden die Daten gespeichert und eine Speicherbestätigung wird eingeblendet. Danach wird wieder die Startseite geöffnet.

Falls Sie es sich während der Eingabe der Daten anders überlegt haben und den Vorgang lieber abbrechen wollen, dann können Sie über den «zurück»-Button oben links oder über den «Abbrechen»-Button unten rechts diese Seite verlassen. Die Daten werden allerdings dann nicht gespeichert.

#### 2.3.3.4 Suchfunktion erkunden

Wieder auf der Startseite angekommen machen wir uns auf für unsere nächste Mission, wir wollen ein Metadaten-File suchen. Dazu wählen Sie ein Suchattribut aus, indem Sie auf das aufklappbare Feld links neben dem Suchfeld klicken. Dann schreiben Sie einen Suchbegriff in das Suchfeld in der Mitte. Je nach Suchattribut kann das Suchfeld auch leer gelassen werden. Als Drittes wählen Sie den Ort aus, den sie Durchsuchen wollen, indem sie auf einen der RadioButton klicken. Jetzt kann die Suche gestartet werden. Durch das Klicken auf den «Suche starten»-Button werden das Suchattribut und der Suchbegriff überprüft, ob sie gültig sind. Wenn Ja, dann wird die Suche gestartet, wenn Nein, dann erscheint eine informative Fehlermeldung. In der Tabelle im vorherigen Abschnitt ist eine Übersicht über die erlaubten Eingaben. Diese gelten nicht nur für das Erstellen von Metadaten-Files, sondern auch für das Suchen.

In der Abbildung 66 sind diese vier Schritte bis zum Starten einer Suche noch einmal bildhaft dargestellt. Wenn alle Eingaben gültig sind, dann öffnet sich ein neues Fenster: das Suchfenster. Indem werden die Suchresultate dargestellt.

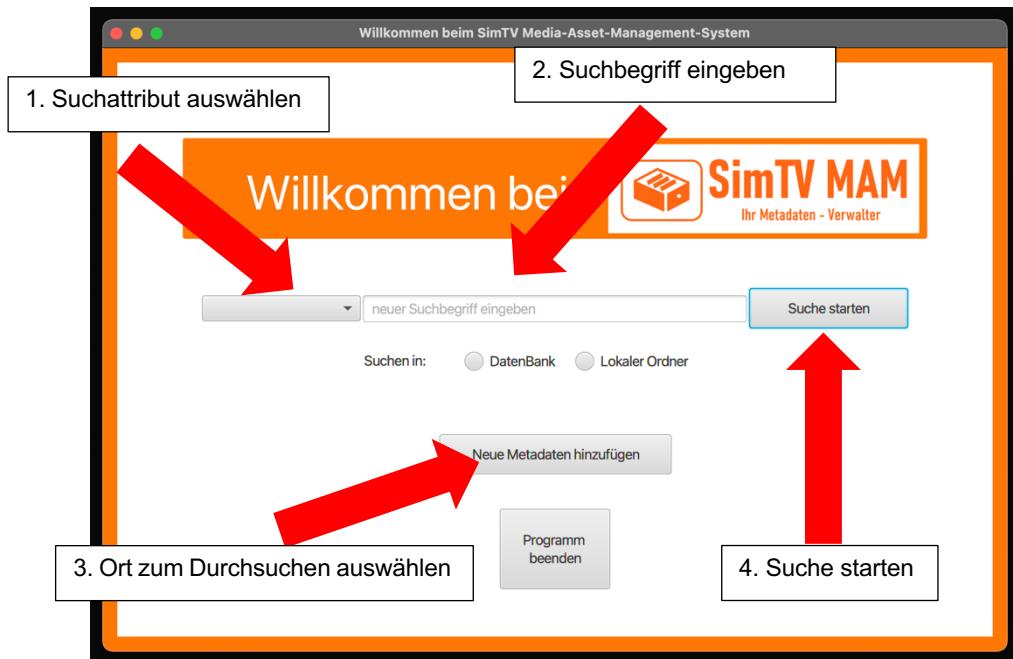


Abbildung 66: Suchvorgang initiieren

Das Suchfenster könnte aussehen, wie die Abbildung 61. Gesucht wurde mit dem Suchattribut «File-ID», dem Suchbegriff «id» und in der Datenbank.

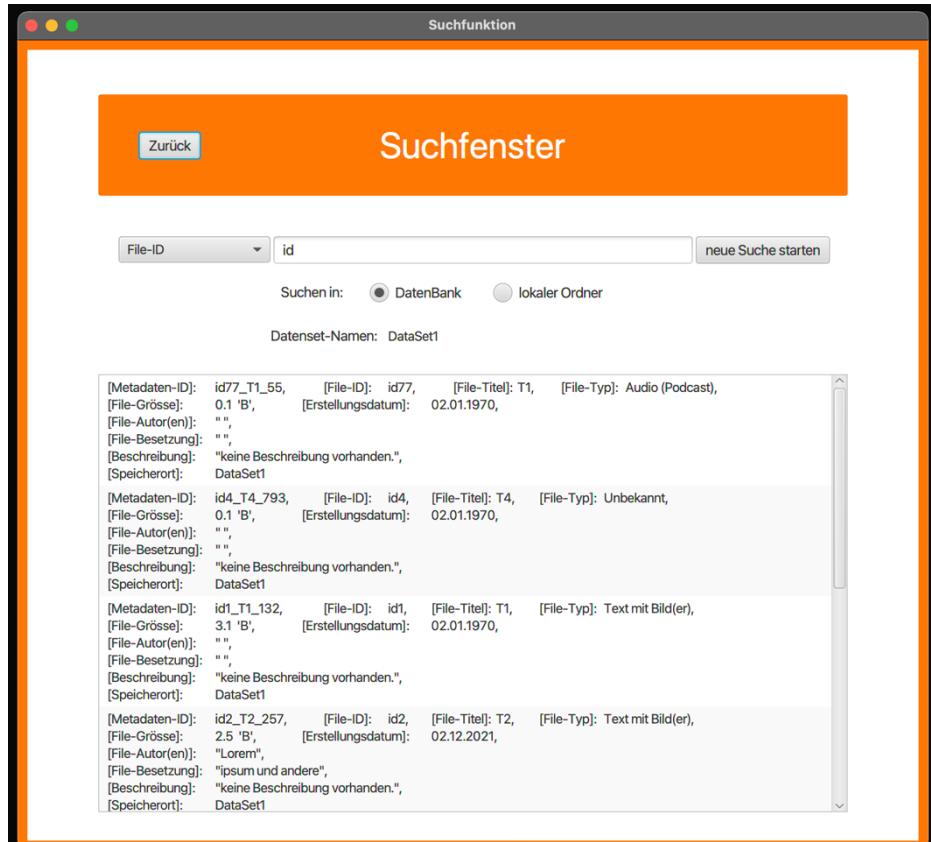


Abbildung 67: Suchfenster mit Suchresultaten

Von hier aus können Sie eine neue Suche starten, mit den gleichen Vorbereitungen wie von der Startseite aus. Die Suchresultate werden sofort unten angezeigt. Dann würde ich sagen: Mission erfolgreich 😊.

#### 2.3.3.5 Das Programm kann noch mehr...

Von dem Suchfenster aus kann nicht nur eine neue Suche gestartet werden. Sondern die Suchresultate können auch bearbeiten werden. Dazu klicken Sie auf ein Suchresultat, wie in der Abbildung 68.

Es erscheint ein neuer Button: der «Ausgewählte Metadaten bearbeiten»-Button. Er macht genau was darauf steht. Klicken Sie auf ihn. Dann öffnet sich die Seite mit der Metadaten-Maske.

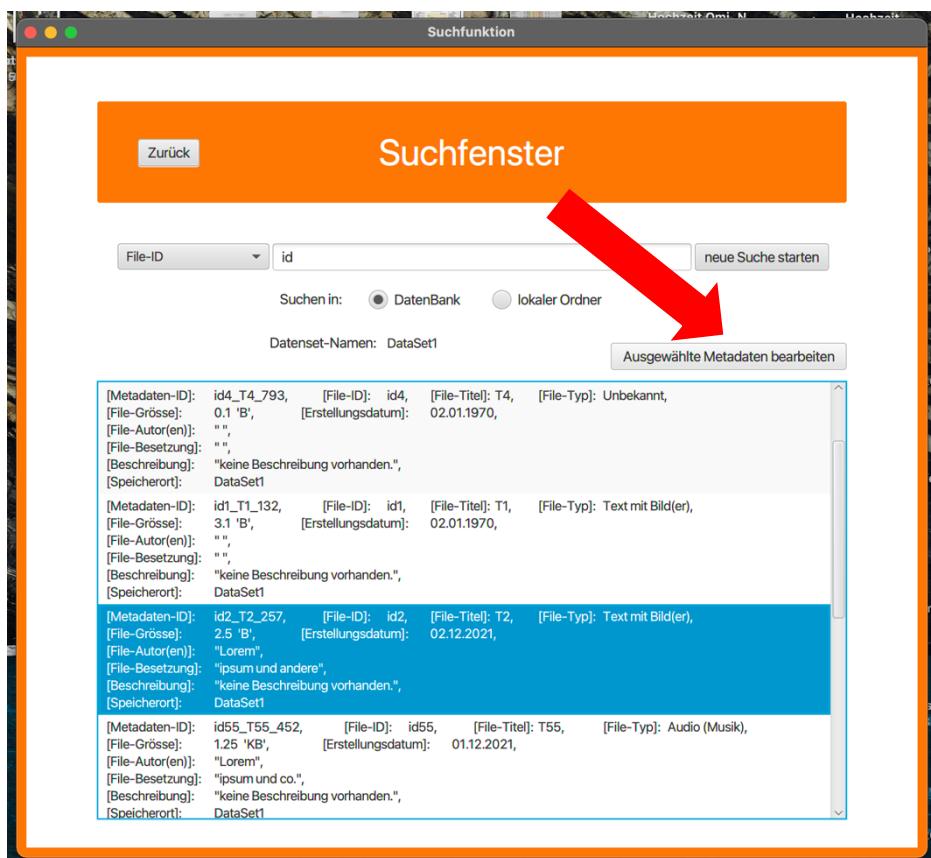


Abbildung 68: Suchfenster mit angewähltem Suchresultat

Diesmal sind die Eingabefelder aber nicht leer. Die Metadaten-Werte sind eingetragen. Diese können Sie jetzt beliebig bearbeiten und verändern. Wird dann wieder auf «Daten speichern» geklickt, werden die Eingaben überprüft, ob eine Regel verletzt wurde, wie bei dem Erstellen

eines neuen Metadaten-Files. Wird keine Regel verletzt, dann werden die Änderungen gespeichert und das Suchfenster mit den Suchresultaten wird wieder geöffnet.

### 2.3.3.6 Und zum Schluss...

Die letzte Funktion, die diese Applikation zu bieten hat, ist das Löschen von Metadaten-Files. Dazu müssen Sie auch ein Suchresultat anklicken. Auf der Metadaten-Masken-Seite befindet sich unten, in der Mitte, ein Button mit der Aufschrift «Daten löschen». Mit diesem werden das Metadaten-File, das in die Datenmaske eingetragen wurde, gelöscht. In der Abbildung 69 ist der Button gut zu sehen.

The screenshot shows a window titled "Metadaten bearbeiten" with a sub-section title "Metadaten - Maske". The form contains the following fields:

- Metadaten-ID: id2\_T2\_257 \*\* Pflichtfelder \*\*
- File-ID: \* id2
- File-Titel: \* T2
- File-Typ: Text mit Bild(er)
- File-Größe: 2.5 Bytes (B) KiloBytes (KB) MegaBytes (MB) GigaBytes (GB)
- File-Erstellungsdatum: 02.12.2021
- File-Autor(en): Lorem
- File-Besetzung: ipsum und andere
- Metadaten-Beschreibung: keine Beschreibung vorhanden.
- Metadaten-Speicherort:  DatenBank  lokaler Ordner  DataSet

At the bottom, there are three buttons: "Daten speichern" (Save data), "Daten löschen" (Delete data, highlighted with a red arrow), and "Abbrechen" (Cancel).

Abbildung 69: Datenmaske mit Metadaten

Viel Spass mit der «SimTV MAM»-Applikation!

## 3 Code

### 3.1 Code-Richtlinien

Folgende Coderichtlinien gelten für dieses Projekt:

#### 3.1.1 Richtlinien für die Namensgebung:

- CamelCase-Schreibweise verwenden
- Packages und Variablen fangen immer mit Kleinbuchstaben an
- Klassen fangen immer mit Grossbuchstaben an
- Konstanten werde nur mit Grossbuchstaben geschrieben
- Lesbare und sinnvolle und konsistente Benennung verwenden, Kürzel vermeiden
- Klassen beschreiben, was sie machen
- Die Exceptiones sollen mit aussagekräftigen Namen benannt werden
- Der Code wird in Englisch verfasst mit Ausnahme der Javadoc-Beschreibungen und Kommentaren.
- Test-Klassen und -Methoden enden mit dem Wort «Test».

#### 3.1.2 Richtlinien für die Programmieren

- Keine Code-Duplikationen, ganz nach dem DRY-Prinzip (Don't Repeat Yourself)
- KISS-Prinzip anwenden.
- Pro Zeile nur eine Zuweisung oder eine geschweifte Klammer (Ausnahme bei Getter und Setter-Methoden).
- Pro Zeile max. 110 Zeichen verwenden.
- Auskommentierter Code wird vor Abgabe gelöscht
- Der Rückgabewert «null» ist zu vermeiden
- Statische Attribute sind zu vermeiden
- Eine Methode sollte eine Aufgabe erfüllen, wenn möglich nicht mehrere.
- Methoden sollen eine Maximallänge von ca. 50 Zeilen, wenn möglich nicht überschreiten
- Klassen sollte eine Maximallänge von ca. 1000 Zeilen, wenn möglich nicht überschreiten

#### 3.1.3 Richtlinien für die Code-Dokumentation

- Kurze aussagekräftige Kommentare verwenden.

- Vermeiden von Kommentaren, die nur den Ablauf beschreiben.
- Öffentliche und geschützte Methoden sind zwingend zu beschreiben.
- Private Methoden können beschreiben werden, müssen aber nicht.
- Möglicher Rückgabewert «null» ist immer zu beschreiben.
- Javadoc sinnvoll verwenden

### 3.1.4 Begründung

Die Richtlinien in der Namensgebung sind rein für die Lesbarkeit und eine bessere Übersicht gegeben. Die Richtlinien für die Programmierung und die Code-Dokumentation sind neben der Lesbarkeit und die bessere Übersicht auch für ein Verhindern von «Bad Smells», zur Förderung der Code-Qualität und zur Wiederverwendbarkeit.

## 4 Abschlussbericht

### 4.1 Evaluationsbericht

#### 4.1.1 Übersicht über den Projektverlauf

Uiii, dieses Projekt war eine anhaltende Berg-und-Tal-Fahrt. Begonnen mit einem optimistischen Start zu Dritt mit guten Vorsätzen und strikten Vorgaben und jetzt am Schluss alleine, teilweise mit der Aufgabe sehr gefordert, wenn nicht zeitweise überfordert.

#### 4.1.2 Ziele erreicht?

Die Ziele P01 bis P07 wurden erfüllt. Das sind sechs Muss- und ein Kann-Ziel. In Worten bedeutet das:

P01: «Metadaten erstellen, einsehen, bearbeiten und löschen» ist erfüllt.

P02: «Funktionierendes GUI» ist erfüllt.

P03: «Metadaten in einer DB speichern wieder laden» ist erfüllt.

P04: «Metadaten in ein Ordner speichern wieder laden» ist erfüllt.

P05: «Funktionierende Suchfunktion» ist erfüllt.

P06 und P07: «Suchfunktion terminiert unter 3 Sekunden (\*)» ist erfüllt.

\* Einzige Ausnahmen sind: Keine Internetverbindung oder keine Verbindung zum Ordner.

#### 4.1.3 Schwierigkeiten? → Ja

Allerdings gab es Schwierigkeiten. Begonnen mit dem Zeitpunkt, an dem meine Teammitglieder das Projekt verlassen haben und ich die gesamte Projekt-Management-Planung überarbeiten musste, über die Schwierigkeiten im Software-Architektur-Entwurf bis hin zu den Problemen während der Programmierung.

Die meisten dieser Schwierigkeiten traten allerdings durch meine fehlende Erfahrung auf und nicht durch das Verlassen der Teammitglieder. Viele Artefakte musste ich doppelt machen, weil erst im Nachhinein klar wurde, dass das so nicht funktioniert. Das erklärt auch meine vielen Überstunden.

Ein Beispiel dafür ist die Umsetzung des Model-View-Controller-Pattern. Bis ich herausgefunden hatte, wie ich das als Code umsetzen kann, hatte ich die Klassendiagramme und

Aktivitätsdiagramme schon angefertigt. Als ich verstand, wie ich das Pattern benutzen kann, stimmten meine Diagramme nicht mehr und ich musste sie abändern.

#### 4.1.4 Sonnen- und Schattenseiten

Was gut lief, war die Überarbeitung des Projektmanagements, nach dem Gespräch mit Herrn Winiger. Weil ich dadurch eine klare Vorstellung bekam, was zu tun war.

Auch die Verwendung der Entwurfs-Pattern Strategy und Singleton liefen sehr gut, weil sie genau für mein Problem die Lösung hatten.

Das Programmieren der GUI an und für sich lief gut. Besonders als mir klar war, wie das MVC-Pattern umzusetzen ist. Aber bis ich IntelliJ so weit hatte, meine JavaFX-Bibliothek zu akzeptierte und das GUI zu laden, kostete es mich Zeit und Nerven.

Der DB-Datenaustausch verlief einiges besser als erwartet. Dafür habe ich mit dem Datenaustausch des lokalen Ordners fast die Zähne ausgebissen.

Am schlechtesten verlief das Programmieren der JUnit-Tests. Ich habe es bis zum Schluss nicht hinbekommen die JavaFX-Elemente in die Tests miteinzubeziehen. Auch die Fehlermeldungen, die durch falsche Eingabewerte aktiviert und dem User angezeigt werden, haben das Tests-schreiben unmöglich gemacht (mit meinen Kenntnissen). Die Schuld dafür kann ich aber nur mir selber geben, da ich es versäumt habe, während dem Programmieren der Applikation die Tests parallel zu implementieren.

#### 4.1.5 Mein Fazit

Im grossen Ganzen hat mit diese Projektarbeit sehr viel gebracht. Für mich war es sehr hilfreich das Projektmanagement, das Software-Engineering und das Programmieren persönlich zu planen und durchzuführen. Besonders deshalb, weil mir dadurch das Zusammenspiel dieser Disziplinen deutlich geworden ist und wie wichtig methodisches Vorgehen, gute Planung und dementsprechende Umsetzung ist. Trotz allen Schwierigkeiten, Schlaflosen Nächten und Überstunden hat dieses Projekt doch Spass gemacht und war die Mühe wert, denn dadurch bin ich um viele Erfahrungen reicher ☺.

## 5 Literaturverzeichnis

- [1] "Media Asset Management – Project Consult", Kampffmeyer/Risse, <https://www.project-consult.de/Files/Media%20Asset%20Management%20200704.pdf>
- [2] Handbuch Projektmanagement, Spring Verlag 2019, 4. Auflage
- [3] <http://blog.axxg.de/model-view-controller-mit-javafx/>
- [4] <https://www.philippauer.de/study/se/design-pattern/strategy.php>

e

## 6 Abbildungsverzeichnis

Abbildung 1: Phasenmodell nach "Handbuch Projektmanagement" .....	4
Abbildung 2: Projektstrukturplan .....	12
Abbildung 3: Netzplan.....	13
Abbildung 4: Domänenmodell.....	35
Abbildung 5: UseCase-Diagramm .....	36
Abbildung 6: Model-View-Controller-Pattern (Quelle [3]).....	43
Abbildung 7: Strategy-Pattern (Quelle [4]).....	44
Abbildung 8: Verteilungsschicht.....	45
Abbildung 9: Kontextabgrenzung.....	45
Abbildung 10: Bausteinsicht Gesamtsystem .....	46
Abbildung 11: Bausteinsicht GUI .....	46
Abbildung 12: Bausteinsicht Datenoperator .....	47
Abbildung 13: Geplantes Klassendiagramm Übersicht .....	47
Abbildung 14: Einteilung für die Detailansichten .....	48
Abbildung 15: Klassendiagramm Detailansicht links .....	49
Abbildung 16: Klassendiagramm Detailansicht rechts.....	50
Abbildung 17: Implementiertes Klassendiagramm Übersicht .....	51
Abbildung 18: Implementiertes Klassendiagramm Ausschnitt 1 .....	52
Abbildung 19: Implementiertes Klassendiagramm Ausschnitt 2 .....	53
Abbildung 20: Implementiertes Klassendiagramm Ausschnitt 3 .....	54
Abbildung 21: Implementiertes Klassendiagramm Ausschnitt 4 .....	55
Abbildung 22: Implementiertes Klassendiagramm Ausschnitt 5 .....	56
Abbildung 23: Implementiertes Klassendiagramm mit den Beziehungen.....	57
Abbildung 24: Elemente des Aktivitätsdiagramme .....	58
Abbildung 25: Aktivitätsdiagramm «Anwendung schliessen».....	58
Abbildung 26: Aktivitätsdiagramm «Neue Metadaten hinzufügen und speichern».....	59
Abbildung 27: Aktivitätsdiagramm «Metadaten suchen» .....	60
Abbildung 28: Aktivitätsdiagramm «Metadaten bearbeiten und speichern oder löschen» .....	61
Abbildung 30: MockUp der Startseite .....	62
Abbildung 29: MockUp der Metadaten-Maske.....	62
Abbildung 31: MockUp des Suchresultat-Fensters.....	63
Abbildung 32: Screenshot der Startseite beim Starten der Applikation .....	64
Abbildung 33: Angeklickte ChoiceBox .....	64
Abbildung 34: Info-Text DB.....	64
Abbildung 35: Info-Text Lokaler Ordner.....	64

Abbildung 36: Fehler bei nicht-auswahl eines Suchattributes .....	65
Abbildung 37: Fehler bei nicht-auswahl eines Such-Ortes .....	65
Abbildung 38: Screenshot des Suchfensters .....	66
Abbildung 39: Auswahl eines Suchresultates .....	67
Abbildung 40: Info-Meldung, wenn keine Treffer gefunden wurden und ins Feld geklickt wurde.....	67
Abbildung 41: Screenshot der Metadaten-Masken-Seite / Variante 1.....	68
Abbildung 42: Bestätigungsmeldung, vor dem Verlassen der Seite.....	68
Abbildung 43: Fehlermeldung, bei leerem Pflichtfeld .....	69
Abbildung 44: Fehlermeldung kein Speicherort.....	69
Abbildung 45: Bestätigungsmeldung .....	69
Abbildung 46: Das ausgewählte Suchresultat .....	69
Abbildung 47: Screenshot der Metadaten-Masken-Seite / Variante 2.....	70
Abbildung 48: Bestätigungsmeldung für den Wechsel des Speicherortes .....	71
Abbildung 49: Meine Test-Klassen .....	92
Abbildung 50: Test-Klassen samt Methoden .....	93
Abbildung 51: bestandene ModelTest-Tests. ....	93
Abbildung 52: bestandene StrategyDBTest-Tests.....	93
Abbildung 53: bestandene StrategyFolderTest-Tests. ....	93
Abbildung 54: IntelliJ Startseite .....	94
Abbildung 55: Git-Repo einfügen und klonen .....	95
Abbildung 56: Trust Maven Project? - Fenster .....	95
Abbildung 57: «Project Structure...» öffnen .....	96
Abbildung 58: Das "Project Structure"-Fenster.....	96
Abbildung 59: "Edit Config..." öffnen.....	97
Abbildung 60: neue Applikation initiieren .....	97
Abbildung 61: Config. ausfüllen .....	98
Abbildung 62: fertig ausgefülltes Config.-Fenster.....	98
Abbildung 63: Applikation starten .....	99
Abbildung 64: Startseite.....	100
Abbildung 65: Metadaten-Masken-Seite.....	101
Abbildung 66: Suchvorgang initiieren .....	103
Abbildung 67: Suchfenster mit Suchresultaten .....	103
Abbildung 68: Suchfenster mit angewähltem Suchresultat .....	104
Abbildung 69: Datenmaske mit Metadaten.....	105
Abbildung 70: Phasenmodell nach "Handbuch Projektmanagement" .....	137

## 7 Tabellenverzeichnis

Tabelle 1: Glossar und Abkürzungen .....	v
Tabelle 2: Projektziele .....	2
Tabelle 3: Projektphasen .....	5
Tabelle 4: Musskriterien.....	5
Tabelle 5: Lösungsausprägungen .....	6
Tabelle 6: Wunschkriterien .....	6
Tabelle 7: Punktevergabe .....	6
Tabelle 8: Präferenzmatrix.....	7
Tabelle 9: Bewertungsbeschreibung .....	7
Tabelle 10: Nutzwertanalyse Backend .....	7
Tabelle 11: Nutzwertanalyse GUI .....	7
Tabelle 12: Nutzwertanalyse DB .....	8
Tabelle 13: Risikoeinschätzung .....	9
Tabelle 14: Risikomatrix .....	9
Tabelle 15: Massnahmentabelle.....	9
Tabelle 16: Problemtabelle .....	10
Tabelle 17: Meilensteine .....	12
Tabelle 18: Terminplanung mit Reihenfolge der Arbeitspakete.....	16
Tabelle 19: Tabelle 1 des Terminplanes vom 28.08 bis 11.11 .....	19
Tabelle 20: Tabelle 2 des Terminplanes vom 15.11. bis zum 14.01 .....	20
Tabelle 21: Arbeitspaket 1.1 .....	21
Tabelle 22: Arbeitspaket 2.1 .....	21
Tabelle 23: Arbeitspaket 2.2 .....	22
Tabelle 24: Arbeitspaket 2.3 .....	22
Tabelle 25: Arbeitspaket 2.4 .....	22
Tabelle 26: Arbeitspaket 2.5 .....	23
Tabelle 27: Arbeitspaket 2.6 .....	23
Tabelle 28: Arbeitspaket 2.7 .....	23
Tabelle 29: Arbeitspaket 2.8 .....	24
Tabelle 30: Arbeitspaket 3.1 .....	24
Tabelle 31: Arbeitspaket 3.2 .....	25
Tabelle 32: Arbeitspaket 3.3 .....	25
Tabelle 33: Arbeitspaket 3.4 .....	25
Tabelle 34: Arbeitspaket 3.5 .....	26
Tabelle 35: Arbeitspaket 3.6 .....	26
Tabelle 36: Arbeitspaket 3.7 .....	26

Tabelle 37: Arbeitspaket 3.8 .....	27
Tabelle 38: Arbeitspaket 4.1 .....	27
Tabelle 39: Arbeitspaket 4.2 .....	28
Tabelle 40: Arbeitspaket 4.3 .....	28
Tabelle 41: Arbeitspaket 4.4 .....	28
Tabelle 42: Arbeitspaket 4.5 .....	29
Tabelle 43: Arbeitspaket 4.6 .....	29
Tabelle 44: Arbeitspaket 4.7 .....	29
Tabelle 45: Arbeitspaket 4.8 .....	30
Tabelle 46: Arbeitspaket 5.1 .....	30
Tabelle 47: Arbeitspaket 5.2 .....	30
Tabelle 48: Arbeitspaket 5.3 .....	31
Tabelle 49: Arbeitspaket 5.4 .....	31
Tabelle 50: Arbeitspaket 5.5 .....	32
Tabelle 51: Arbeitspaket 5.6 .....	32
Tabelle 52: Arbeitspaket 5.7 .....	32
Tabelle 53: Stakeholder-Tabelle .....	33
Tabelle 54: Funktionale Anforderungen .....	34
Tabelle 55: Nicht-Funktionale Anforderungen .....	35
Tabelle 56: Qualitätsanforderungen .....	35
Tabelle 57: UseCase 1 .....	37
Tabelle 58: UseCase 2 .....	38
Tabelle 59: UseCase3 .....	39
Tabelle 60: UseCase 4 .....	40
Tabelle 61: UseCase 5 .....	41
Tabelle 62: Regeln und Fehlermeldungen .....	72
Tabelle 63: Testszenarien .....	74
Tabelle 64: Testergebnis 1.1 .....	75
Tabelle 65: Testergebnis 1.2 .....	77
Tabelle 66: Testergebnis 1.3 .....	79
Tabelle 67: Testergebnis 1.4 .....	79
Tabelle 68: Testergebnis 1.5 .....	80
Tabelle 69: Testergebnis 1.6 .....	81
Tabelle 70: Testergebnis 2.1 .....	83
Tabelle 71: Testergebnis 2.2 .....	84
Tabelle 72: Testergebnis 2.3 .....	85
Tabelle 73: Testergebnis 2.4 .....	85
Tabelle 74: Testergebnis 3.1 .....	86

Tabelle 75: Testergebnis 3.2 .....	87
Tabelle 76: Testergebnis 3.3 .....	88
Tabelle 77: Testergebnis 3.4 .....	89
Tabelle 78: Testergebnis 4.1 .....	90
Tabelle 79: Testergebnis 4.2 .....	91
Tabelle 80: Testergebnis 5.1 .....	92
Tabelle 81: Testergebnis 5.2 .....	92
Tabelle 82: Attributtypen-Tabelle .....	101

## 8 Anhang

### 8.1 Akzeptanzerklärung

Ich, Simon Wittker bezeuge hier mit meiner Unterschrift, dass ich vollkommen hinter dieser Projektarbeit, der Dokumentation und dem Quellcode stehe. Ausserdem bezeuge ich, dass ich diese Projektarbeit selbstständig durchgeführt, die Dokumentation selbst verfasst und den Quell-Code selbst programmiert habe. Mit Ausnahme der Teile der Dokumentation, an denen meine ehemaligen Gruppenmitglieder Sabrina Künzi und Billy Utermann mitgewirkt haben, als sie noch Teil der Projektgruppe waren.

Datum 23.12.2021

Simon Wittker



## 8.2 Entscheidungs-Log

Da ich seit dem 04.11.2021 allein an diesem Projekt arbeite, führe ich dieses Entscheidungs-Log als Ersatz für die Protokolle der Teamsitzungen. Die Zeit, die ich dafür brauche, wird unter «Projektmanagement» und «Projektcontrolling 1, 2» verbucht.

### 8.2.1 Übersicht

EL-Nr.	Kurzbeschreibung	Fachgebiet
1	Namensänderung des Projektes	PM
2	Überarbeitung des Projektmanagement (PM)	PM
3	Keine «Frontend und Backend»-Lösung	PM
4	Überarbeitung des Requirement Engineering	SWE
5	Überarbeitung des GUI	SWE
6	Anpassung des UseCase-Diagramms und Beschreibung	SWE
7	Als GUI ein Model-View-Controller-Pattern benutzen	SWE
8	Für die Datenaufbereitung ein Strategy -Pattern benutzen	SWE
9	Benötigte Komplexibilität durch neue Klasse «Owner»	SWE
10	Missverständnis bezüglich Komplexibilität	SWE
11	Model als Singleton, Entwurf wird überarbeitet.	SWE
12	Zeitplan anpassen	PM
13	«FileType» als «Enum» programmiert	OOP
14	«FileSizeUnit» als «Enum» initialisieren	OOP
15	IntelliJ-Projekt neu initialisieren	OOP
16	JavaFX-Projekt statt Java-Projekt	OOP
17	GUI wird nicht «responsive»	OOP
18	Methoden auslagern	OOP
19	«Altes» und «neues» Klassendiagramm	OOP
20	JavaFX-Alert-Nachrichten statt Exceptiones oder Hinweise in Labels	OOP
21	Messenger-Klasse	OOP
22	Strategy-Variable	OOP
23	Sortierfunktion wird weggelassen	OOP
24	Grosser Test-Knorz	OOP

## 8.2.2 Die einzelnen Entscheidungen

### Kurzbeschreibung: Namensänderung des Projektes

EL-Nr. 1	Datum: 06.11.21	Fachgebiet: PM
----------	-----------------	----------------

#### Problem:

Da ich jetzt allein das Projekt «SBS-TV MAM» allein durchführe, ist «SBS», was für «Sabrina, Billy, Simon» steht, nicht mehr passen

#### Lösungsentscheid:

Der Projektname wird in «SimTV MAM» umgewandelt. Das Projektziel, ein MAM-System zu machen, bleibt.

#### Konkrete Auswirkung:

Anpassung der Dokumentation und allen Erwähnungen von «SBS-TV» auf «Sim-TV»

### Kurzbeschreibung: Überarbeitung des Projektmanagement (PM)

EL-Nr. 2	Datum: 06.11.21	Fachgebiet: PM
----------	-----------------	----------------

#### Problem:

Da ich jetzt allein dieses Projekt durchführe, haben sich auch der Projektumfang verändert, auf (min.) 150 h. Dadurch haben sich die Projektziele und so ziemlich jedes Artefakt, dass wir schon haben, verändert.

#### Lösungsentscheid:

Das Projektmanagement wird komplett überarbeitet und auf den Umfang meiner geforderten Arbeitsstunden zugeschnitten.

#### Konkrete Auswirkung:

Erstellen eines Soll-Konzeptes, Projektziele werden überarbeitet, MVP wird angepasst, Grobanforderungen werden angepasst, PSP wird angepasst, ein Netzplan wird erstellt, Terminplanung wird überarbeitet, Muss- und Soll-Kriterien werden angepasst.

### Kurzbeschreibung: Keine «Frontend und Backend»-Lösung.

EL-Nr. 3	Datum: 06.11.21	Fachgebiet: PM
----------	-----------------	----------------

#### Problem:

Als Gruppe hatten wir uns für ein Web-Frontend entschieden. Doch allein ist das innerhalb von 150 h nicht durchführbar.

**Lösungsentscheid:**

Statt Frontend und Backend wird eine Java Applikation mit Java-GUI geplant.

**Konkrete Auswirkung:**

Nutzwertanalyse, welches GUI am besten Umsetzbar ist.

**Kurzbeschreibung: Überarbeitung des Requirement Engineering**

EL-Nr. 4	Datum: 06.11.21	Fachgebiet: SWE
----------	-----------------	-----------------

**Problem:**

Da ich im Projektmanagement die Projektziele, das MVP, die Grobanforderungen und daraus folgend die Muss- und Sollkriterien geändert habe, hat das Auswirkungen auf die schon vorhandenen Anforderungen.

**Lösungsentscheid:**

Die vorhandenen Artefakte des Requirement Engineering werden überarbeitet.

**Konkrete Auswirkung:**

Anpassen der Analysedokumente, konkret der Stakeholder-Tabelle, der Systemidee, die Produktanforderungen sowie des Domänenmodells.

**Kurzbeschreibung: Überarbeitung des GUI.**

EL-Nr. 5	Datum: 06.11.21	Fachgebiet: SWE
----------	-----------------	-----------------

**Problem:**

Da ich das Projekt allein fertig mache und ich den Projektnamen geändert habe, stimmt das Logo nicht mehr.

**Lösungsentscheid:**

Ich erstelle ein neues Logo und überarbeite die MockUp's, die mir als Vorlage dienen werden. (So pragmatisch und zeitsparend wie möglich). Dazu entscheide ich, dass die Applikation auf Deutsch angezeigt wird.

**Konkrete Auswirkung:**

Logo wird ersetzt, Farben in der Doku angepasst, MockUp's werden geändert.

**Kurzbeschreibung: Anpassung des UseCase-Diagramms und Beschreibung.**

**EL-Nr. 6**

**Datum:** 06.11.21

**Fachgebiet:** SWE

**Problem:**

Durch die Anpassung der Projektziele und der Grobanforderungen stimmt das UseCase-Diagramm nicht mehr, weil siehe EL-Nr. 3.

**Lösungsentscheid:**

UseCase-Diagramm müssen überarbeitet werden.

**Konkrete Auswirkung:**

UseCase-Diagramm mitsamt der Beschreibung wird überarbeitet.

**Kurzbeschreibung: Als GUI ein Model-View-Controller-Pattern benutzen**

**EL-Nr. 7**

**Datum:** 10.11.21

**Fachgebiet:** SWE

**Problem:**

**Fehlende Zeit und fehlende Erfahrung bei der GUI-Programmierung.**

**Lösungsentscheid:**

Ich benutze das Model-View-Controller-Pattern und erhoffe mir dabei eine Zeitersparnis.

**Konkrete Auswirkung:**

Für die Architektur des GUI's verwende ich das Model-View-Controller-Pattern.

**Kurzbeschreibung: Für die Datenaufbereitung ein Strategy -Pattern benutzen**

**EL-Nr. 8**

**Datum:** 10.11.21

**Fachgebiet:** SWE

**Problem:**

Die Erweiterbarkeit des Systems soll gewährleistet werden.

**Lösungsentscheid:**

Ich benutze das Strategy-Pattern für die Datenaufbereitung und erhoffe mir dabei eine einfache Erweiterbarkeit.

**Konkrete Auswirkung:**

Für die Architektur der Datenaufbereitung verwende ich das Strategy -Pattern und entwerfe es dementsprechend.

**Kurzbeschreibung: Benötigte Komplexität durch neue Klasse «Owner»**

**EL-Nr. 9**

**Datum:** 10.11.21

**Fachgebiet:** SWE

**Problem:**

Im ersten Statusmeeting wurde von Roman Fischer gesagt, dass er mindestens eine n-zu-m-Beziehung sehen möchte, dass das Projekt die nötige Komplexibilität aufweist.

**Lösungentscheid:**

Ich baue die neue Klasse «Owner» ein, die eine n-zu-m-Beziehung zu den Metadaten hat.

**Konkrete Auswirkung:**

Klassendiagramm, Bausteinsicht, UseCase-Diagramm und Beschreibung wurden angepasst und die Anforderungen ergänzt.

**Kurzbeschreibung: Missverständnis bezüglich Komplexibilität**

**EL-Nr. 10**

**Datum:** 11.11.21

**Fachgebiet:** SWE

**Problem:**

Im zweiten Statusmeeting habe ich das mit der Komplexibilität durch die n-zu-m-Beziehung angesprochen und wurde von Roman Fischer korrigiert. Eine 1-zu-n-Beziehung ist ausreichend. Daher hat mein Projekt jetzt eine höhere Komplexibilität als nötig.

**Lösungentscheid:**

Ich mach die Änderungen vom EL-Nr. 9 (Benötigte Komplexibilität durch neue Klasse «Owner») rückgängig.

**Konkrete Auswirkung:**

Bei dem Klassendiagramm, der Bausteinsicht, dem UseCase-Diagramm und den Beschreibungen wurden die Änderungen rückgängig gemacht und die Anforderungen korrigiert.

**Kurzbeschreibung: Model als Singleton, Überarbeitung des Entwurfes**

**EL-Nr. 11**

**Datum:** 11.11.21

**Fachgebiet:** SWE

**Problem:**

Für die konkrete Umsetzung des MVC-Patterns für die GUI, orientiere ich mich an einer Implementierungsvorlage von «<http://blog.axxg.de/model-view-controller-mit-javafx/>». Pro View erstelle ich einen Controller, der die Elemente der View anspricht, Daten einsetzt und ausliest. Dazu eine einzige Model-Klasse, die die Daten von den Controllern speichert und allfällige Hilfsfunktionen bereitstellt. Wichtig ist, dass die Model-Klasse keine direkte

Verbindung zu einer View hat, nur über den Controller. Da jeder Kontroller auf das Model zugreifen will und Daten dort Speichern und Laden will, kann es nicht zwei Model-Instanzen geben.

#### Lösungsentscheid:

Um sicherzustellen, dass es nur eine Model-Instanz gibt, verwende ich das Singleton-Pattern. Das Model übernimmt außerdem die Rolle als Schnittstelle zwischen Context und Controller.

#### Konkrete Auswirkung:

Die Editor-Klasse braucht es dadurch nicht mehr. Das Klassendiagramm und die Bausteinansicht werden überarbeitet. Die Model-Klasse wird mit dem Singleton-Pattern entworfen.

#### Kurzbeschreibung: Zeitplan anpassen

EL-Nr. 12	Datum: 11.11.21	Fachgebiet: PM
-----------	-----------------	----------------

#### Problem:

In dem Statusmeeting wurde ich darauf hingewiesen, dass in dem PSP einige erforderliche Artefakte nicht aufgeführt sind. Dazu bekam ich den Tipp, dass ich die Suchfunktionen der Java-Bibliotheken nutzen kann (und soll) statt diese selbst zu implementieren.

#### Lösungsentscheid:

Ich kontrolliere meinen Zeitplan, denn dort habe ich eine detailliertere Übersicht über alle Artefakte des Projektes. Dazu verteile ich die Stunden, die ich für die Suchfunktion geplant habe auf andere Programmierarbeiten.

#### Konkrete Auswirkung:

Zeitplan angepasst und fehlende Artefakte hinzugefügt (z.B. Evaluationsbericht)

#### Kurzbeschreibung: «FileType» als «Enum» programmiert

EL-Nr. 13	Datum: 11.11.21	Fachgebiet: OOP
-----------	-----------------	-----------------

#### Problem:

Momentan habe ich den FileType so geplant, dass ich beim Initialisieren jedes Mal den Typ schreiben muss und dazu kann es zu Schreibfehler kommen.

#### Lösungsentscheid:

Ich programmiere eine «Enum» mit Konstanten, die den Typ beschreiben.

### Konkrete Auswirkung:

Programmieren eine FileType-Enum Klasse, in dem ich die Konstanten festlege und beschreibe. Dazu wird die Auswahl des File-Typs mit einer ChoiceBox getroffen, statt eines TextFields. Im Klassendiagramm auch so ergänzt. Die Zeit wird unter dem Arbeitspaket Nr.4.6 «Dokumentation à jour halten (Ph 4)» verbucht.

### Kurzbeschreibung: «FileSizeUnit» als «Enum» initialisieren

EL-Nr. 14	Datum: 11.11.21	Fachgebiet: OOP
-----------	-----------------	-----------------

#### Problem:

Geplant war, dass bei der Eingabe die Grösseneinheit (z.B. B, KB, MG, GB) eingegeben werden sollte. Aber da jedes Mal beim Initialisieren die Einheit geschrieben werden müsste, verkompliziert das die ganze Sache unnötig und Schreibfehler können auftreten.

#### Lösungsentscheid:

Ich programmiere eine «Enum» mit Konstanten, die den Grösseneinheit beschreiben.

### Konkrete Auswirkung:

Programmieren eine FileSizeUnit-Enum Klasse, in dem ich die Konstanten festlege und beschreibe. Dazu wird die Auswahl des File-Typs mit vier RadioButton getroffen, statt eines TextFields. Im Klassendiagramm auch so ergänzt. Die Zeit wird unter dem Arbeitspaket Nr.4.6 «Dokumentation à jour halten (Ph 4)» verbucht.

### Kurzbeschreibung: IntelliJ-Projekt neu initialisieren

EL-Nr. 15	Datum: 16.11.21	Fachgebiet: OOP
-----------	-----------------	-----------------

#### Problem:

Beim Programmieren kam es zu Problemen mit der Version. Ich wollte mit Maven arbeiten, doch nach dem Update von JetBeans funktionierte mein Programm nicht mehr, da IntelliJ die von mir verwendeten Maven-Plugins nicht mehr unterstützt.

#### Lösungsentscheid:

Ich initialisiere ein neues IntelliJ-Projekt mitsamt Git-Repo.

### Konkrete Auswirkung:

Neues IntelliJ-Projekt erstellen, die Klassen rüber kopieren, bestehendes Git-Repo löschen, neues Git-Repo erstellen und wieder verknüpfen.

## Kurzbeschreibung: JavaFX-Projekt statt Java-Projekt

**EL-Nr. 16** | **Datum:** 22.11.21 | **Fachgebiet:** OOP

### Problem:

Als ich mit der GUI beginnen wollte, erstelle ich ein JavaFXGui-File, das als Beispiel dienen sollte. Aber mein IntelliJ-IDEA fand die Klasse für JavaFX nicht.

### Lösungsentscheid:

Nach langer Suche und Probieren, habe ich mich entschieden, statt ein «normales» Java-Projekt, ein JavaFX-Projekt zu erstellen.

### Konkrete Auswirkung:

JavaFX -Projekt erstellen, die Klassen rüber kopieren, bestehendes Git-Repo löschen, neues Git-Repo erstellen und wieder verknüpfen.

## Kurzbeschreibung: GUI wird nicht «responsive»

**EL-Nr. 17** | **Datum:** 23.11.21 | **Fachgebiet:** OOP

### Problem:

Die Elemente des erstellten GUIs passen sich nicht an die Fenstergrösse an. Das Problem könnte mit mehr Aufwand beseitigt werden. Die Funktionalität ist nicht betroffen.

### Lösungsentscheid:

Aus Zeitmangel und Rückstand halte ich das GUI schlicht und verzichte auf «responsibility».

### Konkrete Auswirkung:

Das GUI wird sich nicht an die Fensterbreite und -höhe anpassen.

## Kurzbeschreibung: Methoden auslagern

**EL-Nr. 18** | **Datum:** 23.11.21 | **Fachgebiet:** OOP

### Problem:

Verschiedene Methoden, z.B. «openStartWindow()» und «openSearchWindow()», werden von verschiedenen Controllern aufgerufen und es entsteht daher eine Codeduplikation.

### Lösungsentscheid:

Ich lagere die Methoden aus, die in mehreren Controllern aufgerufen werden.

### Konkrete Auswirkung:

Diese Methoden werden in die Model-Klasse ausgelagert, um Codeduplikationen zu verhindern. Das Klassenmodell wird aber nicht angepasst. -> Grund EL-NR. 19.

### Kurzbeschreibung: «Altes» und «neues» Klassendiagramm

EL-Nr. 19	Datum: 23.11.21	Fachgebiet: SWE
-----------	-----------------	-----------------

#### Problem:

Viele Methoden werden, aufgrund des «eine Methode für eine Aufgabe»-Prinzips, aufgespalten und umbenannt. Um Codeduplikation zu vermeiden, werden auch Methoden ausgelagert -> siehe EL-NR. 18. Das Klassendiagramm stets zu ändern und die Namen einzeln anpassen kostet viel Zeit und hat nichts mit Programmieren zu tun.

#### Lösungsentscheid:

Da ich den Punkten aus der Arbeitsmappe gerecht werden will, werde ich das geplanten Klassendiagramm (mit den Änderungen EL-Nr. 10 – 14) in die Doku als «geplantes Klassendiagramm» und dann das schlussendlich fertig implementierte Klassendiagramm, erstellt mit IntelliJ, als «implementierte Klassendiagramm» in die Doku aufnehmen und die Abweichungen kurz erläutern.

Dadurch erhoffe ich mir einen Zeitgewinn, da ich nicht nach jeder (Namens-)Änderung das Klassendiagramm anpassen muss.

#### Konkrete Auswirkung:

Das jetzige Klassendiagramm als «geplantes Klassendiagramm» abspeichern und zum Schluss, wenn die Applikation ausliefer-bereit ist, dass «implementierte Klassendiagramm» erstellen.

### Kurzbeschreibung:

#### JavaFX-Alert-Nachrichten statt Exceptiones oder Hinweise in Labels

EL-Nr. 20	Datum: 23.11.21	Fachgebiet: OOP
-----------	-----------------	-----------------

#### Problem:

Wenn der User die Anwendung falsch bedient, dann möchte ich Ihn darauf hinweisen, was er falsch gemacht hat und wie er seine Falsch-Eingabe beheben kann.

#### Lösungsentscheid:

Damit ich nicht die Warnhinweise und Fehlermeldungen in ein extra Label hineinschreiben muss, benutze ich die JavaFX-Alert-Klasse.

**Konkrete Auswirkung:**

Ich erstelle für jeden Fehler, jedes Falschverhalten oder Falscheingabe eine aussagekräftige Nachricht als JavaFX-Alert, damit der User informiert wird.

**Kurzbeschreibung: Messenger-Klasse**

EL-Nr. 21	Datum: 29.11.21	Fachgebiet: OOP
-----------	-----------------	-----------------

**Problem:**

Da es viele Wege gibt die Anwendung falsch zu benutzen und ich für jedes eine Fehlermeldung/Bestätigung und co. einblende, ist die Zahl der Nachrichten gestiegen und diese Nachrichten brauche ich an mehreren Orten und sie nehmen in den Klassen Platz weg.

**Lösungsentscheid:**

Ich erstelle eine eigene Klasse mit den Nachrichten, die erscheinen.

**Konkrete Auswirkung:**

Erstellung einer «Messages»-Klasse, die die javafx-Alert enthältet. Damit ich die Nachrichten Objekt-los aufrufen kann, mache ich alle Methoden «static».

**Kurzbeschreibung: Strategy-Variable**

EL-Nr. 22	Datum: 29.11.21	Fachgebiet: OOP
-----------	-----------------	-----------------

**Problem:**

Ich habe mich beim Software-Entwurf für das Strategy-Pattern entschieden und muss jetzt irgendwie der Context-Klasse mitteilen, welche Strategie gefahren wird.

**Lösungsentscheid:**

Ich erstelle eine Variable in der Model-Klasse, die ich den Methoden für die Strategie mitgebe.

**Konkrete Auswirkung:**

Erstellung der Strategie-Variable namens «strategyVar», die einen Integer-Wert mitliefert, der anzeigt, welche Strategie gewählt wird.

## Kurzbeschreibung: Sortierfunktion wird weggelassen

EL-Nr. 23 | Datum: 07.12.21 | Fachgebiet: OOP

### Problem:

Terminrückstand plus die Sortierfunktion wurde noch nicht implementiert.

### Lösungsentscheid:

Ich habe mich entschieden, die Sortierfunktion wegzulassen.

### Konkrete Auswirkung:

Keine Sortierfunktion implementiert.

## Kurzbeschreibung: Grosser Test-Knorz

EL-Nr. 24 | Datum: 13.12.21 | Fachgebiet: OOP

### Problem:

1. Der Plan war, während dem Programmieren die nötigen Tests zu schreiben. Leider blieb es bei dem Plan.
2. Dazu kommt, dass ich es nicht fertig gebracht habe für die Klassen, die auf JavaFX-Elemente zugreifen, gültige Tests zu schreiben. Jedes Fehlverhalten löst eine JavaFX-Alert-Meldung aus. Ich habe es nicht geschafft, dem Test beizubringen auf diese zu achten. Stattdessen gibt er unerwartete Fehlermeldungen aus.

### Lösungsentscheid:

1. Die wichtigsten Methoden werden mit Tests abgedeckt, so gut es geht. Dazu zählen vor allem die Strategy-Klassen (StrategyDB und StrategyFolder). In der Model-Klasse so gut wie möglich.
2. Die JavaFX-Klassen werden nicht mit Tests abgedeckt. Stattdessen werden manuelle Testszenarien eingesetzt.

### Konkrete Auswirkung:

Einige wenige Testklassen wurden erstellt, die das nötigste Testen. Manuelles Testszenario wird durchgeführt. Beim nächsten Projekt besser nach «tdd» arbeiten!

## 8.3 Protokolle

### 8.3.1 Protokoll vom Telefonmeeting vom 13.09.21

Thema: Projektthema genauer definieren

Billy: Anwesend

Sabrina: Verhindert

Simon: Anwesend (Protokollführer)

Unser Projektthema, dass wir am 28.08.21 grob definiert hatten, war die Erstellung eines Media Asset Management (MAM) - Systems. Uns war von Anfang an bewusst, dass wir nicht ein ganzes MAM-System «aus dem Boden stampfen» können. Dieses Meeting war dafür gedacht, nach einer kurzen Recherchezeit, einen möglichst praktisch Umsetzbaren Abriss zu beschließen, den wir als Projektarbeit, in der zur Verfügung-stehenden Zeit, auch umsetzen können.

Wir haben uns entschlossen, dass wir eine Basis für ein mögliches MAM-System kreieren werden, mit besonderem Fokus auf eine Erweiterbarkeit und Ausbaufähigkeit. Konkret ist die Idee, dass wir nicht mit den Assets arbeiten, sondern mithilfe der Metadaten agieren, die dann mit dem entsprechenden Asset verlinkt ist. Unser System sollte also folgende Funktionen haben:

- Das Hinzufügen (und Speichern) von Metadaten,
- Das Suchen von Metadaten,
- Das Ändern (und Speichern) von Metadaten,
- Das Löschen von Metadaten.

Unser Kunde wird darüber hinaus nicht der Arbeitgeber von Billy sein, sondern eine fiktive Firma.

8488 Turbenthal den 14.09.21

Simon Wittker

### **8.3.2 Protokoll vom Teamsmeeting vom 23.09.21**

Billy: Anwesend

Sabrina: Anwesend (Protokollführerin)

Simon: Anwesend

Zuerst haben wir die bestehenden Gantt Diagramm und Domänenmodell angeschaut.

Planung für alles oder nicht? Unnötig, weil klar ist was wir machen wollen.

Domänenmodell erklärt (Security, Customer Service, CRM werden wir nie machen). Zeiteinteilung wie lange jeder Teil gehen würde in Theorie mit unendlich Zeit sollten wir noch machen.

Edit Tools, Cut Programm, Tags setzen, wie Metadaten bearbeiten

MDM je nach Hersteller sehr verschieden, warum neu machen, wenn man alte Teile einbinden kann (z.B. Cutting Programm) Wir wollen eine einfache Software machen, Ziel Modulbauweise, wie würde weiter gehen, wie spielen Dinge zusammen.

Wir haben besprochen, dass die Dozenten irgendwie nicht richtig verstanden haben, was wir machen wollen. Aber wir haben uns darauf geeinigt, dass die Arbeit in erster Linie für uns Sinn ergeben muss.

Besprechung des Gantt Diagrammes:

Frontend Backend, Zeiten teilweise nur 1 Tag, nicht alle Daten sind im Frontend, Use Case Diagramm Verteilung, Testkonzept, EER Diagramm, Verteilungsdiagramm, Sequenz, Datenmodellierung. Wir haben alle anstehenden Blöcke verteilt. Zur Verteilung siehe Gant Diagramm. Domänenmodell in Gant, hinzugefügt.

Als letztes haben wir besprochen wer im Moment Projektleiter sein will. Sabrina stellt für nächstes Meeting Dokumente zusammenführen und bereitet Sitzung vor.

Nächster Termin Mittwoch den 29.09.21 um 20:00.

8353 Elgg, 26.09.21

Sabrina Küenzi

### **8.3.3 Protokoll vom Teams-Meeting vom 13.10.21**

Billy: Anwesend

Sabrina: Anwesend (Protokollföhrerin)

Simon: Anwesend

Agenda:

- Alle: Kurz Status aktuelle Tasks
  - o Billy: Ressourcenplanung gemacht, todo sind nachwievor Risikoanalyse/Nutzwertanalyse
- Billy: Ressourcenplanung zeigen. Fehlt etwas? Gantt? Oder Excel?
- Billy: Rückmeldung Mock Ups
- Nächster Meilenstein definieren
- Sabrina: Testkonzept und Fragen zur Dokumentation

Sabrina weiss nicht, was sie an den Frontend Use Cases von Simon verbessern soll → Simon ändert noch ein paar Sachen an denen morgen.

Sabrina: Testfälle ok wie in der präsentierten Vorlage aus dem Internet? Sind verschiedene Searchkeywords 1 Testfall? Simon und Billy finden nicht. 1 Testfall mit verschiedenen Varianten.

Sendungen: wir machen noch ein Knopf «Sendungen», wo wir Sendungen verwalten können. Dann kann jede Media/Assets in einer Sendung eingeteilt werden (Dropdown-Menü).

Sabrina erstellt Testfälle farbig, nach FFHS-Farben. Sabrina findet, wir sollten in der Arbeit ein bisschen auf den Farben beachten. Simon und Billy sind einverstanden.

Sabrina: Ziele sind gut bei der präsentierten Vorlage (Problemlösung inkl. Wertung). Sabrina übernimmt dies und auch die Risikoanalyse. Die Ziele wurden im Statusmeeting bemängelt, wir sollten diese noch verbessern.

Gantt mit Excel machen (Sabrina) (mit anderem Programm doch nicht gut), so dass es gut lesbar ist am Schluss. (evtl. 2-teilig)

Billy übernimmt Schnittstellenbeschreiben und Backend Use Cases

Mock Ups:

- Direkt einfache Suche
- Liste von Suchresultaten
- Dropdowns für Filetyp (Movie, Picture, Document) und Sprache

Wir machen ein richtiges Literaturverzeichnis am Ende der Dokumentation.

Konzeptionsphase soll bis am 23.10.2021 beendet sein.

Nächste Sitzung am 25.10.2021 am Abend.

8353 Elgg, 14.10.21

Sabrina Küenzi

### **8.3.4 Protokoll vom Telefonmeeting vom 28.10.2021**

Sabrina: Anwesend

Simon Anwesend (Protokollführer)

Agenda:

- Beide: Kurz Status aktueller Task.
- Beide: Ausstieg Billy, wie weiter?
- Beide: Ausblick.

Simon: Frontend-Use-Cases angepasst, Mock-Up's angepasst.

Sabrina: Hatte keine Zeit, und nach dem Ausstieg von Billy keine Lust, an ihren Tasks zu arbeiten. Wird sich aber diese Woche Zeit nehmen, Risikoplanung und Testfälle noch nachzu-arbeiten.

Billys Ausstieg hat uns beide schockiert. Sabrina überliess es Simon, ob wir weitermachen oder auch aufhören. Simon will die Projektarbeit fortsetzen. Sabrina zieht mit.

Sabrina wird Risikoplanung und Testfälle bis in einer Woche machen.

Simon übernimmt die Backend-Use-Cases von Billy sowie die Schnittstellen-beschreibung.

Simon wird Alexander Winiger kontaktieren, bezüglich der Probleme und Unklarheiten bezüg-lich des Projektumfangs und der Zeitplanung, die sich durch den Abschied von Billy ergeben haben.

Nächstes Treffen: Sonntag, 6.11.21 um 20:00

8488 Turbenthal den 28.10.21

Simon Wittker

## **8.3.5 Statusberichte und Protokolle der Statusmeetings**

### **8.3.5.1 Statusbericht 1**

# **SBS-TV MAM**

**Statusbericht 1**

**von**

**Sabrina Küenzi, Billy Utermann und Simon Wittker**  
**(Gruppe 7)**

**Zürich, HS 21/22**

## 1.1.1 Ausgangslage

### 1.1.1.1 Ist-Situation

SBS-TV ist ein regionaler Fernsehsender in der Romandie. In einem Fernsehunternehmen müssen viele technische Systeme zusammenarbeiten und die Prozesse sind komplex. Beispiele davon sind Programmplanungs-, Workflowautomatisierung- und Reportingssysteme. Alle basieren aber auf einer zentralen Datenbank, die alle zur Verfügung stehende Medienobjekte und deren Metadaten beinhaltet.

Anstatt eine existierende Software zu kaufen, hat SBS-TV vor ungefähr 20 Jahren ein eigenes System mit der Software FileMaker entwickelt. Mit dem Aufkommen vom Internet und den vielen verschiedenen neuen Medienformate (Podcast, Social Media, Website, usw.) kommt nun dieses System an seine Grenzen, weshalb SBS-TV den Auftrag gegeben hat, das System neuzudenken und vom Grund auf wieder aufzubauen. Der erste Schritt soll ein erweiterbares System sein, um alle Medienobjekte samt Metadaten zu verwalten. Ein sogenanntes Media Assets Management System.

### 1.1.1.2 Definition Media Assets Management System

*“Das Media Asset Management sorgt für die Aufnahme, Speicherung, Kategorisierung, Indizierung und Bereitstellung von Media Assets mit der Prämisse, eine hohe Bearbeitungsgeschwindigkeit und eine optimale Wiederverwendung zu ermöglichen. Ein Media Asset Management System verwaltet beliebige Inhalte über Metadaten und Datenbanken.“<sup>1</sup>*

## 1.1.2 Projektziele und Inhalte

### 1.1.2.1 Projektziele

In diesem Projekt wollen wir (Sabrina Küenzi, Simon Wittker und Billy Utermann) ein einfaches, erweiterbares MAM-System entwickeln. Folgende Projektziele haben wir identifiziert:

ID	Ziel	Muss/Kann
P01	Am 23.12.2021 muss das Projekt fertig und lauffähig sein.	M
P02	Das Team erfüllt die Vorgaben aus der Arbeitsmappe weitgehend, damit der Lerneffekt des Moduls am höchsten ist.	M

<sup>1</sup> Siehe “Media Asset Management – Project Consult” Kampffmeyer/Risse

P03	Das Team lernt durch die Rollenverteilung verschiedene Aufgaben wahrzunehmen, auch wenn sie den einzelnen Mitgliedern weniger liegen.	M
P04	Das Team zeigt die Fähigkeit, ein anspruchsvolles Projekt mit unbekannten Teammitglieder erfolgreich zu realisieren	M
P05	Der Auftraggeber kann die Erkenntnisse des Projekts brauchen und darauf aufbauen.	K

Eine detailliertere Liste der Anforderungen ist im Kapitel 2 enthalten.

### 1.1.2.2 Minimum Viable Product (MVP)

Wir wollen kein marktfertiges Produkt liefern, sondern ein durchgedachtes und erweiterbares System, das eine einfache Verwaltung von Medienobjekte samt Metadaten ermöglicht.

Im folgenden Abschnitt sind die Grobanforderungen gelistet.

### 1.1.2.3 Grobanforderungen

- Das System hat eine intuitive und ansprechende Multi-Plattform Bedienoberfläche und kann mit einer Maus bedient werden.
- Metadaten von Media Assets hinzufügen, löschen und bearbeiten.
- Media Assets kategorisieren und archivieren.
- Das System ermöglicht eine schnelle erweiterte Suche.
- Das System soll einfach erweiterbar sein.
- Das System soll grösstenteils mit Java entwickelt werden.
- Das neue System sollte möglichst selbst entwickelt sein und wenn, dann schweizerische Firmen als Partner bevorzugen.

### 1.1.2.4 Abgrenzung

- Das System implementiert keine Zugriffs- und Berechtigungsfunktion, sondern ist lediglich Passwortgeschützt.
- Das System speichert keine Medienobjekte, sondern nur Metadaten. Für die Speicherung von Medienobjekten wird je nach Typ nur eine URL oder eine ID von einem bestehenden Dienst gespeichert. Die Speicherung, Konvertierung und Bereitstellung von solchen grossen Dateien würden den Rahmen unseres Projekts sprengen. Zudem verfügt SBS-TV schon über einen externen Partner für diesen Teil des Systems und sieht da keinen Grund für eine Veränderung.

- Die Bedienoberfläche wird vorläufig nicht für mobile Geräte optimiert.

## **1.1.3 Rahmenbedingungen**

### **1.1.3.1 Technische Vorgaben (FFHS)**

Als Programmiersprache ist hauptsächlich, d.h. zu min 51%, Java zu verwenden. Die Regeln und Sprachmittel der objektorientierten Programmierung sollen beachtet werden.

Die Nutzung von weiteren Sprachen ist möglich. Für unser Projekt werden wir ein Web-Frontend mit HTML und JavaScript entwickeln, sowie die SQL-Sprache für Datenbankanfragen benutzen.

### **1.1.3.2 Technische Ressourcen**

Das Frontend des Systems sollte mit jedem aktuellen Browser abrufbar sein. Das Backend des Systems wird mit Java 11 programmiert und soll mit einer aktuellen Version des Java Runtime Environment ausführbar sein.

## **1.1.4 Eckwerte**

### **1.1.4.1 Projektdauer**

Das Projekt startet offiziell am 28.08.2021 mit einem Kick-Off-Event in der Gleisarena in Zürich und endet mit der finalen Abgabe am 23.12.2021 und der Schlusspräsentation am 14.01.2022.

### **1.1.4.2 Aufwand**

Das Projekt muss von den Teammitgliedern in einer Gesamtarbeitszeit von ca. 450 Stunden abgeschlossen sein. Dies entspricht ungefähr ein Arbeitstag pro Woche pro Person. Dies sollte allen von Anfang an klar sein, und ist nicht zu vernachlässigen.

### **1.1.4.3 Phasentermine**

Dieses Projekt verwendet das Phasenkonzept vom Lehrmittel «Handbuch Projektmanagement»<sup>2</sup>. Wir passen die Phasen sinnvoll auf das Projekt an.

---

<sup>2</sup> Handbuch Projektmanagement, Springer Verlag 2011



Abbildung 70: Phasenmodell nach "Handbuch Projektmanagement"

Für unseres Projekt haben wir uns für das Wasserfallmodell entschieden, ein lineares (nicht iteratives) Vorgehensmodell, dass sich für dieses Projekt mit vorgegebenen Anforderungen und Leistungen präzise beschreiben lässt.

Nr.	Phase	Beschreibung
1	Beauftragung	Auftrag bekommen, Planung der nächsten Phase, grobe Schätzungen, Projektantrag freigeben.
2	Initialisierung	"Im Rahmen der Initialisierungsphase müssen verbindliche Aussagen zu Machbarkeit, Risiken und Stakeholdern erarbeitet werden. Wesentliche Grundlagen dazu sind die Analyse der aktuellen Situation sowie klar vereinbarte Ziele und die Formulierung der Anforderungen an das Resultat des Projekts, z. B. an das zu entwickelnde Produkt."
3	Konzeption	"In der Konzeptphase werden zu den Anforderungen Lösungsvarianten ausgearbeitet. Für die ausgewählte Lösungsvariante erfolgt eine detaillierte Ausarbeitung des Lösungskonzeptes. Basierend auf dem Lösungskonzept erfolgt die Detailplanung des Projektes."
4	Realisierung	"In der Realisierungsphase werden die Pläne aus der Konzeptphase verwirklicht."
5	Einführung/Abchluss	Übergabe organisieren, Abschlussbericht und Schlussrechnung erstellen. Projektdokumentation ergänzen, Projektbeurteilung und «Lessons Learned» verarbeiten. Evaluierungsbericht erstellen.

Meilenstein	Name	Start	Ende
1	Beauftragung	28.08.2021	19.09.2021

2	Initialisierung	20.09.2021	03.10.2021
3	Konzeption	04.10.2021	31.10.2021
4	Realisierung	01.11.2021	05.12.2021
5	Einführung/Abschluss	06.12.2021	23.12.2021

#### 1.1.4.4 Rollenverteilung

Wir sehen in unserem Projekt 3 Hauptrollen: Projektmanager, Architekt/Requirement Engineer und Entwickler/Tester. Jede Rolle soll einmal jedem Teammitglied zugeteilt sein. Die Zuteilung ist wie folgt geplant, wird aber je nach Interessen/Stärken eventuell ein bisschen abweichen:

Person	Rolle	Phase
Billy Utermann	Projektleiter	Beauftragung
Simon Wittker	Architekt / RE	Initialisierungsphase
Sabrina Küenzi	Entwicklerin / Testerin	
Sabrina Küenzi	Projektleiterin	Konzeption
Billy Utermann	Architekt / RE	
Simon Wittker	Entwickler / Tester	
Simon Wittker	Projektleiter	Realisierung
Sabrina Küenzi	Architektin / RE	
Billy Utermann	Entwickler / Tester	Abschluss

#### 1.1.5 Grundlagen

Es gibt ein bestehendes System, welches den heutigen Anforderungen nicht mehr entspricht. Ein externer Partner hat in den letzten 2 Jahren eine neue Webseite für den Fernsehsender realisiert. Dieser Partner hat schon viel Arbeit in der Modellierung der Daten investiert, und daran können wir uns eventuell orientieren, bzw. bei schon existierenden Überlegungen anfangen.

#### 1.1.6 Kurzbeschreibung des Gesamtstatus

**Termine:** Projektidee und grobe Anforderungen sind da. Die Beauftragungsphase wird somit sehr wahrscheinlich Ende Woche nach Plan abgeschlossen sein können.

**Lieferobjekte:** Wir hätten gerne schon eine Risikoanalyse, einen Lösungsansatz (inkl. Nutzwertanalyse) und ein grobes Projektstrukturplan bis zum Statusmeeting 1 realisiert, aber die Zeit war zu knapp, und die Organisation innerhalb der Gruppe schwieriger als geplant, da wir

uns vorher nicht gekannt haben. → Massnahme: Bis Ende Woche klare Kommunikationswege, Verantwortungen und wöchentliche Meetings planen.

Simon (Architekt/RE) hat schon Anforderungen und Use-Cases angefangen, dort sind wir gut in der Zeit drin.

**Andere Probleme:** Krankheitsfall bei erster Sitzung, keine grosse Wirkung aufs Projekt

**Vorgehensmodell:** Phasenmodell: Wasserfallmodell

**Nächste Schritte:** Kommunikationswege definieren / Projektstrukturplan / Risikoanalyse / Lösungsansatz / Erster Prototypanfang

### 8.3.5.2 Protokoll vom Statusmeeting vom 15.09.21

Billy: Anwesend

Sabrina: Anwesend (Protokollführerin)

Simon: Anwesend

Urs-Martin Künzi: Anwesend

Alexander Winiger: Anwesend

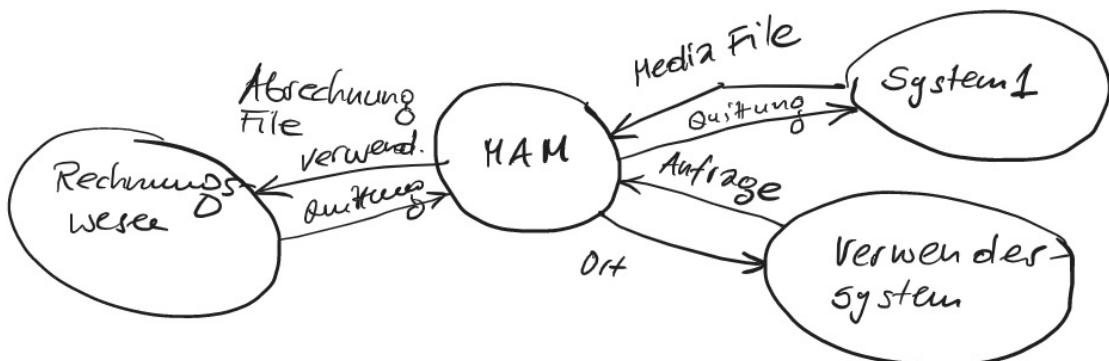
Roman Fischer: Anwesend

Die ersten 10 Minuten hat Billy unseren Statusbericht vorgestellt und Simon einen Use Case.

Wir wurden gebeten den Statusbericht nach dem Meeting hochzuladen.

Wir haben gefragt, wie es bezüglich Literaturangaben mit den Vorgaben aussieht, es gibt offenbar keine Vorgaben.

Die Dozenten meinten sie seien sich etwas unsicher in Bezug auf den Umfang unseres Projekts. Diese Grafik wurde von Herrn Winiger gezeichnet:



Es wurden Fragen bestellt bezüglich Prototyps, Schnittstellen, Identifikation, Stichwortsuche, Dateneinschränkung und Domänenmodell.

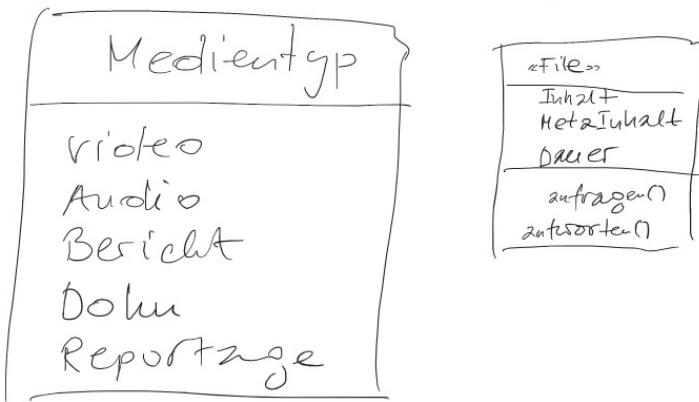
Unsere Projektziele wurden bemängelt, da diese nicht funktionaler Art sind, ausserdem wurden wir gebeten Kann-Ziele zu definieren (falls wir zu viel Zeit hätten).

Die Meilensteine sollen wir arithmetisch herleiten, Arbeitspakete verteilen, Annäherung an den Projektstrukturplan sowie Schätzungen für Zeiten machen und später mit ist Zeiten verglichen.

Es wurde vorgeschlagen Entwickler und Tester als Rollen zu trennen.

**Statusmeeting 1**

Und es wurde eine weitere Grafik gezeichnet:



Wir sollen uns überlegen, welcher kleine machbare Teil gemacht werden soll, dieser soll durchdacht und erweiterbar sein.

Wir wurden gebeten ein Domänenmodell zu erstellen, dieses soll so erstellt werden, als wenn wir unendlich Zeit hätten. Danach sollen Einschränkungen folgen.

Danach wurde noch der Termin für das nächste Statusmeeting festgelegt: Der 10.11.21 um 20:00.

Es wurde noch festgelegt das Meilensteine nicht mit Statusmeetings übereinstimmen müssen. Als letztes wurde noch gesagt, dass der Statusbericht noch ergänzt werden soll, mit den Sachen, die uns vom Statusmeeting geblieben sind.

8353 Elgg, 15.09.21

Sabrina Küenzi

**8.3.5.3 Statusbericht 2**

**SBS-TV**

**SimTV MAM**

**Statusbericht 2**

**Semesterarbeit im Modul PA\_5**

**von**

**Sabrina Küenzi, Billy Utermann und Simon Wittker**

**Gruppe\_ZH\_7; Schweiz; HS 21/22**

## Gesamtstatus

Nachdem zwei Gruppenmitglieder die Gruppe verlassen haben, habe ich, Simon Wittker, das Projekt den neuen Gegebenheiten angepasst. Da das Projektmanagement punkto Lieferobjekt einige Lücken hinterlassen hat, habe ich das Projektmanagement komplett überarbeitet und auf die neue Ein-Mann-Gruppe angepasst.

Auch der Name habe ich geändert, von SBS-TV MAM (SBS stand für Sabrina, Billy und Simon) in SimTV MAM (Sim wie Simon).

Nach anfänglichen Überlegungen, ob ich nicht auch «die Flinte ins Korn werfen» soll, habe ich durch ein Gespräch mit Herr Winiger neue Motivation erhalten, die Projektarbeit «durchzuziehen» und abzuschliessen.

Vom Zeitplan her bin ich jetzt aber ca. 3 Tage hinterher. Als Massnahme habe ich aus den anfänglichen 8 Stunden pro Woche, 16 Stunden gemacht, bis ich wieder im Zeitplan bin.

## Status Lieferobjekte

Folgende Lieferobjekte sind schon fertig:

- Projektauftrag
- PSP
- Meilensteinplanung
- Netzplan
- Arbeitspakete
- Terminplanung
- Domänenmodell
- Nutzwertanalysen
- UseCase
- Schnittstellenbeschreibung
- Verteilungsschicht

## Sonstiges

- Kontextabgrenzung
- Bausteinsicht
- Klassendiagramm

Es fehlen noch:

- Ein Dynamisches Modell
- Testkonzept
- Spezifikation der Bedienoberflächen
- Coderichtlinien
- Der gesamte Quellcode
- Die Unit-Test
- Testprotokolle
- Evaluierungsbericht
- Handbuch

## Nächste Schritte

Für die nächste Arbeitseinheit stehen das Aktivitätsdiagramm, das Testkonzept und die restlichen Architekturentscheide auf dem Plan. Dann ist die Konzeptionsphase abgeschlossen.

Für die Realisierungsphase werde ich ca. 10 volle Arbeitstage brauchen, wenn alles «glatt» läuft mit der Implementierung.

Bis zum Versand der gebundenen Projektdokumentation und dem Quellcode benötige ich ca. 2 volle Arbeitstage.

### 8.3.5.4 Reflektionsprotokoll von Statusmeeting 2

#### Was habe ich von dem Statusmeeting mitgenommen?

- Es wird das bewertet, was abgegeben wurde! Was alles von den Artefakten, die wir zu Dritt angefertigt haben, übernommen wurde, was geändert und was weggelassen wurde, das ist mir überlassen. Nur die Stunden, die eingesetzt wurden, müssen ersichtlich sein.
- Der Nachweis, dass ich die 150 Stunden in das Projekt investiert habe, soll im Zeitplan gut ersichtlich sein. Das heisst, dass die Stunden, die ich ins Projekt gesteckt habe, als wir noch zu dritt sind, auch ausgewiesen werden müssen. Am besten der alte Ressourcenplan mitschicken!
- Da ich die Entscheidungen allein Treffe, muss ich eine Art «Entscheidungsprotokoll» führen, das als Ersatz für Protokolle der Teamsitzungen fungieren kann.
- Wenn eine Aufgabe mit einer Java-Bibliotheken lösbar ist, dann darf und soll ich sie auch benutzen.
- Die Applikation am besten als «Dockerfile» bzw. einer Docker-Instanz abgeben.
- Neben dem Projektumfang wurde auch die Befragungszeit verkürzt. Von 30 Min. als Dreierteam auf 10 Min. Die Präsentationszeit beträgt nach wie vor gleich viel.
- Methodisches Vorgehen ist das A und O!
- Kontrolliere, ob alle geforderten Lieferartefakte in Planung sind!

Turbenthal 10.11.21

Simon Wittker

**Sonstiges**

### 8.3.6 Sonstiges

#### 8.3.6.1 Alter Ressourcenplan vom SBS-TV

#### SBS-TV MAM: Ressourcenplanung

<b>Projektstart</b>	28.08.21
<b>Projektende</b>	23.12.21
<b>Projektleiter</b>	Billy, Sabrina, Simon
<b>Aktualisiert am</b>	2.11.

Nr.	Arbeitspaket	DONE	SOLL-Aufwand in Personenstunden	SOLL-Aufwand Billy	IST-Aufwand Billy	SOLL-Aufwand Simon	IST-Aufwand Simon	SOLL-Aufwand Sabrina	IST-Aufwand Sabrina	IST-Aufwand in Personenstunden	Abweichung
1	Beauftragung		9.00	3.00	3.00	3.00	3.00	3.00	3.00	9.00	0.00
1.1	Auftrag definieren	x	3.00	1.00	1.00	1.00	1.00	1.00	1.00	3.00	0.00
1.2	Team kennenlernen	x	3.00	1.00	1.00	1.00	1.00	1.00	1.00	3.00	0.00
1.3	Organisation / Kommunikation	x	3.00	1.00	1.00	1.00	1.00	1.00	1.00	3.00	0.00
2	Initialisierung		71.00	33.00	16.00	22.00	32.00	16.00	4.00	52.00	19.00
2.1	Projektziele definieren	x	6.00	2.00	4.00	4.00	4.00			8.00	-2.00
2.2	<i>Projektplanung</i>										
2.2.1	Projektstrukturplan	x	6.00	3.00	3.00		3.00	3.00		6.00	0.00
2.2.2	Netzplan		4.00	4.00			4.00			4.00	0.00

**Sonstiges**

2.2.3	Meilensteine planen / definieren	x	1.00	1.00						0.00	1.00
2.2.4	Ressourcenplanung	x	6.00	6.00	6.00					6.00	0.00
2.2.5	Grobanforderungen	x	8.00	4.00		4.00	4.00			4.00	4.00
2.3	Status-quo + Systemgrenzen definieren		2.00	2.00						0.00	2.00
2.4	Nutzwertanalyse vom Projekt		4.00	4.00						0.00	4.00
2.5	Domänenmodell	x	6.00			6.00	6.00			6.00	0.00
2.6	Rollen definieren verteilen	x	1.00	1.00	2.00					2.00	-1.00
2.7	Statusbericht 1	x	3.00	1.00	1.00	1.00	1.00	1.00	1.00	3.00	0.00
2.8	Reflektion Statusbericht 1	x	1.00					1.00	1.00	1.00	0.00
2.9	Risikoanalyse		4.00					4.00		0.00	4.00
2.10	Teamsitzungen	x	9.00	3.00		3.00	2.00	3.00	2.00	4.00	5.00
2.11	Dokumentation à jour halten		10.00	2.00		4.00	8.00	4.00		8.00	2.00
2.12	Projektleitung		8.00	8.00	8.00					8.00	0.00
3	Konzeption		81.00	35.00	0.00	26.00	18.00	24.00	1.00	19.00	
3.1	Use Cases Frontend	x	8.00			7.00	4.00	1.00	1.00	5.00	3.00
3.2	Mock Ups Frontend	x	8.00			8.00	6.00			6.00	2.00
3.3	Backend Nutzwertanalyse		4.00	4.00						0.00	4.00
3.4	Datenmodellierung DB		4.00	4.00						0.00	4.00
3.5	Use Cases Backend	x	4.00	4.00			6.00			6.00	-2.00
3.6	Schnittstellenbeschreibung		4.00	4.00						0.00	4.00
3.7	Diagramme erstellen										
3.7.1	EER Diagramm		4.00	4.00						0.00	4.00
3.7.2	Verteilungsdiagramm		4.00	4.00						0.00	4.00
3.7.3	Klassendiagramm		4.00			4.00				0.00	4.00

**Sonstiges**

3.7.4	Sequenzdiagramm	4.00	4.00						0.00	4.00
3.8	Testkonzept erstellen	4.00					8.00		0.00	4.00
3.9	Teamsitzungen	9.00	3.00		3.00		3.00		0.00	9.00
3.10	Dokumentation à jour halten	12.00	4.00		4.00	2.00	4.00		2.00	10.00
3.11	Projektleitung	8.00					8.00		0.00	8.00
<b>4</b>	<b>Realisierung</b>	<b>210.00</b>	<b>69.00</b>	<b>0.00</b>	<b>72.00</b>	<b>0.00</b>	<b>69.00</b>	<b>0.00</b>	<b>0.00</b>	
4.1	<i>Entwicklungsumgebung einstellen</i>									
4.1.1	Git Repo und Berechtigung	1.00	1.00						0.00	1.00
4.1.2	Server und Deployment Skripts	4.00	4.00						0.00	4.00
4.1.3	Code Richtlinien definieren	4.00					4.00		0.00	4.00
4.1.4	Umgebung lokal installieren	3.00	1.00		1.00		1.00		0.00	
4.2	<i>Packages Backend</i>									
4.2.1	"Server" programmieren	24.00	12.00		12.00					
4.2.2	MetadatenEditor programmieren	16.00	8.00		8.00					
4.2.3	Datenbank nach EER erstellen	24.00	12.00		12.00				0.00	24.00
4.2.4	Suchfunktion programmieren	16.00			16.00					
4.3	<i>Components Frontend</i>									
4.3.1	Website HTML + CSS	16.00					16.00		0.00	16.00
4.3.2	Website JS	24.00					24.00			
4.4	Existierende Daten migrieren/erstellen	8.00	8.00						0.00	8.00
4.5	Tests programmieren	24.00	8.00		8.00		8.00			
4.6	Statusmeeting 2	3.00	1.00		1.00		1.00		0.00	3.00
4.7	Statusbericht Reflexion	1.00					1.00		0.00	1.00
4.8	Teamsitzungen	18.00	6.00		6.00		6.00		0.00	18.00

**Sonstiges**

4.9	Dokumentation à jour halten	24.00	8.00		8.00		8.00		0.00	24.00
4.10	Projektmanagement	16.00			16.00				0.00	16.00
5	Einführung / Abschluss	77.00	23.00	0.00	23.00	0.00	31.00	0.00		
5.1	Handbuch für Benutzer	24.00	8.00		8.00		8.00		0.00	24.00
5.2	Projektdokumentation fertigstellen	12.00	4.00		4.00		4.00		0.00	12.00
5.3	Druck der Dokumentation	3.00	1.00		1.00		1.00		0.00	3.00
5.4	Versand	3.00	1.00		1.00		1.00		0.00	3.00
5.5	Schlusspräsentation vorbereiten	12.00	4.00		4.00		4.00		0.00	12.00
5.6	Präsentation	3.00	1.00		1.00		1.00		0.00	3.00
5.7	Nachbesprechung	3.00	1.00		1.00		1.00		0.00	3.00
5.8	Teamsitzungen	9.00	3.00		3.00		3.00		0.00	9.00
5.9	Projektleitung	8.00					8.00		0.00	8.00
	Total Stunden									
	Billy Utermann		163.00	19.00					19.00	
	Simon Wittker				146.00	53.00			53.00	
	Sabrina Küenzi						143.00	8.00	8.00	

SOLL

Gesamtaufwand in Personenstunden 448.00

IST

80.00

### 8.3.6.2 Alte MockUps von SBS-TV



## Sonstiges

Filetitel: Titel 1	Produzent: Til Eulenspiegel	Cast: TKKG
Filetyp: Video	Erstellungsjahr: 2018	Tags: 1:35'; 3:45'
Filetitel: Titel 4	Produzent: Alfred Hitchcock	Cast: ???
Filetyp: Video	Erstellungsjahr: 2020	Tags:
Filetitel: Titel 5	Produzent: Unbekannt	Cast: Unbekannt
Filetyp: Video	Erstellungsjahr: 2017	Tags: 5:11'; 5:45'

## Sonstiges

