

## Anforderungsdokument

### 1. Stakeholder

Der Medianizer ist für verschiedene Endbenutzer einsetzbar. Zum einen ist er für Privatpersonen geeignet, die eine Unterstützung bei der Organisation ihrer Film- und/oder CD-Sammlung wünschen. Dies ist umso relevanter, je größer die Sammlung ist. Darüber hinaus eignet sich die Anwendung auch für kleinere Unternehmen, die mit Medien handeln. Besonders relevant sind hier Bereiche wie Second Hand oder An- und Verkauf, da hier i.d.R. keine andere Übersicht, wie etwa ein Sortiment besteht. Für alle Endbenutzer gilt, dass sie eine intuitive Bedienung, eine klare Darstellung der Daten und eine zuverlässige Speicherung ohne online-Anwendungen erwarten.

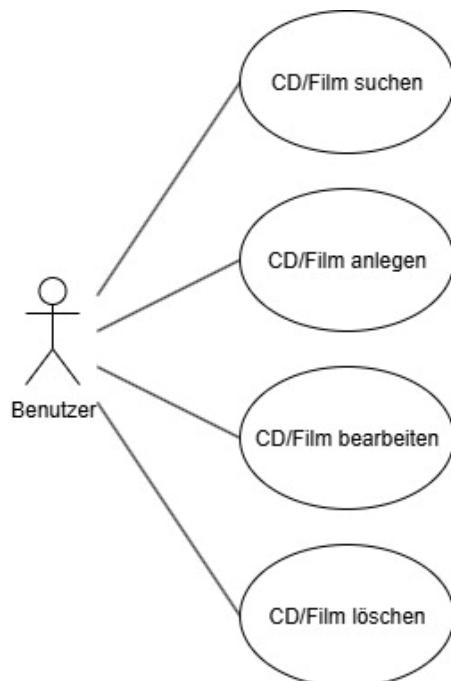
### 2. Funktionale Anforderungen

Als Benutzer möchte ich Filme mit Titel und CDs mit Titel und Interpret suchen können, um in einem neuen Fenster eine übersichtliche Ausgabe der vorhandenen Einträge in Tabellenform mit den Spalten Titel, ggf. Interpret (bei CDs) und Anzahl zu erhalten.

Als Benutzer möchte ich einen Eintrag in der Tabelle auswählen und ihn löschen oder die Felder Titel, Anzahl und ggf. Interpret (bei CDs) bearbeiten können.

Als Benutzer möchte ich, wenn bei der Suche keine Ergebnisse für eine Ausgabe gefunden wurden, direkt an ein Fenster weitergeleitet werden, das meine Eingaben übernimmt und mir die Möglichkeit bietet diese um die Anzahl meiner Exemplare zu erweitern und meinen Eintrag zu speichern.

Es folgt ein entsprechendes UML-Use-Case-Diagramm:



### 3. Nichtfunktionale Anforderungen

Neben den grundlegenden Funktionen gibt es weitere Anforderungen an das Projekt. Die Benutzeroberfläche soll klar strukturiert, intuitiv und leicht verständlich sein. Die Anwendung muss sich stabil ausführen lassen und darf bei Standardnutzung keine Abstürze oder Datenverluste verursachen. Die Antwortzeiten beim Laden oder Speichern sollen kurz bleiben, sodass der Nutzer keine Verzögerung bemerkt.

Das Programm soll ohne Internetverbindung funktionieren und lokal auf Windows-Systemen ausgeführt werden können. Die Daten werden in einer mitgelieferten Datenbank gespeichert, sodass keine Installation oder Serverkonfiguration nötig ist.

Die Versionsverwaltung soll die Nachvollziehbarkeit und Wartbarkeit verbessern.

### 4. Glossar

Begriff	Definition
CRUD	Create, Read, Update, Delete (Grundoperationen einer Datenbank)
GUI	Graphical User Interface (grafische Benutzeroberfläche)
SQLite	Lokale relationale Datenbank
JavaFX	Framework für grafische Oberflächen in Java
Maven	Build-Tool zur Verwaltung von Abhängigkeiten
GitHub	Plattform zur Versionsverwaltung und Quellcode-Veröffentlichung