**Projektbericht:**

**Eigene Film- und Serienbewertung mit persönlichem Ranking**

**Ziel:**  
Mit diesem Projekt möchte ich eine Webanwendung entwickeln, in der man Filme und Serien individuell bewerten und ein eigenes Ranking erstellen kann, um seine All-Time-Lieblingsfilm oder -serie zu finden. Oft sieht man Rankings im Internet, die nicht den eigenen Geschmack widerspiegeln. Diese Anwendung soll es Nutzerinnen und Nutzern ermöglichen, ihre Lieblingsfilme und -serien nach persönlichen Kriterien zu bewerten, zu sortieren und später gesammelt anzusehen. Zusätzlich sollen aktuelle Neuerscheinungen angezeigt werden, damit man nichts verpasst.

**Funktionsweise:**  
Auf der Startseite befindet sich ein zentrales Eingabefeld, in das man einen Film oder eine Serie eintippen kann. Die Anwendung greift dabei auf eine öffentliche API (wahrscheinlich TMDB) zu und schlägt passende Titel vor. Wenn man einen Titel auswählt, wird eine kurze Beschreibung angezeigt, und man kann direkt zur Bewertung übergehen.

Die Bewertung erfolgt in fünf Kategorien:

* **Story**
* **Charaktere**
* **Schauspielerische Leistung**
* **Bild & Ton**
* **Unterhaltung**

Für jede Kategorie vergibt man 1 bis 5 Sterne. Aus den Einzelwertungen wird anschließend eine Gesamtnote berechnet. Diese wird gemeinsam mit dem Titel in der Datenbank gespeichert.

Im Bereich „Ranking“ sieht man alle bereits bewerteten Titel – wahlweise nur Filme, nur Serien oder beides gemischt. Die Liste ist nach der eigenen Bewertung sortiert, sodass man sofort erkennt, welche Titel für einen selbst ganz oben stehen.

Ein zusätzlicher Bereich zeigt aktuelle Neuerscheinungen an, um den Nutzer regelmäßig auf dem Laufenden zu halten und neue Denkanstöße zu geben, was sein nächster Film oder seine nächste Serie sein könnte.

**Verwendete Technologien:**

* **Frontend:** Vue.js
* **Backend:** Java mit Spring Boot
* **Datenbank:** MySQL mit Zugriff über JPA
* **API:** TMDB oder ähnliche Film-/Serien-API
* **Kommunikation:** HTTP mit JSON zwischen Frontend und Backend

**Herausforderungen:**

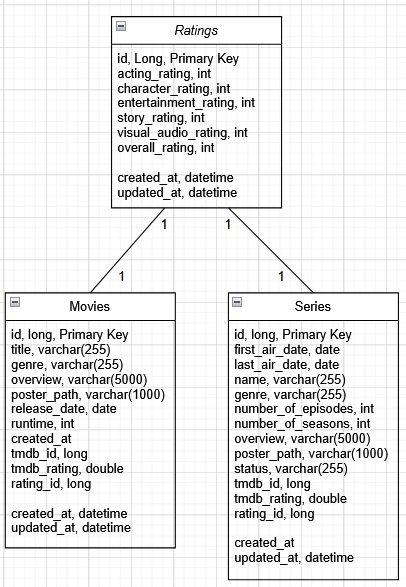
* Saubere Trennung von Frontend und Backend bei der API-Nutzung
* Dynamische Darstellung und Umschaltung zwischen Film/Serie im Ranking
* Berechnung und Speicherung der Bewertungen in der Datenbank
* Übersichtliche und ansprechende Benutzeroberfläche im Frontend

**Projektplanung (Gantt-Diagramm)**

**Ein Bild, das Text, Zahl, Schrift, Reihe enthält.

KI-generierte Inhalte können fehlerhaft sein.**

**Datenbankstruktur**



**Reflexion und Herausforderungen**

**1. Beschreibung zu lang für Datenbank**  
Beim Speichern traten Fehler auf, da die Beschreibung mancher Filme/Serien zu lang war. Lösung: Änderung von VARCHAR(255) auf VARCHAR(5000)

**2. Unterschiedliche Felder in der TMDB-API**  
Die API verwendet bei Filmen und Serien unterschiedliche Feldnamen (z. B. title vs. name). Lösung: zusätzliche Logik zur Unterscheidung im Backend