



Unser Filmabend

[Projekt]

Inhaltsverzeichnis

Übersicht und Motivation	3
Detailanforderungen	3
1. Benutzermanagement	3
2. Lieblingsfilme	3
3. Filmabend planen.....	3
4. Filmabende einsehen.....	4
Filmdaten und Vorschläge.....	4
Dokumentation.....	5
Technisches.....	5

Übersicht und Motivation

In Zeiten der Streaming-Video-Dienste und der hohen Verfügbarkeit von Filmen wird es immer schwieriger, sich auf einen gemeinsamen Film zu einigen, den man in einer Gruppe zu einem Filmabend schauen kann. Die Grundidee der Software ist, dieses Problem zu lösen.

Nutzende können ihre Lieblingsfilme hinterlegen. Für einen Filmabend werden den Teilnehmenden zu einem definierten Genre passende Filme vorgeschlagen.

Die Software ist eine Spring Boot Anwendung mit angeschlossener Datenbank, die entweder über eine Web-UI oder über einen JavaFX-Klienten angesprochen werden kann.

Detailanforderungen

1. Benutzermanagement

Eine neue nutzende Person kann für sich selbst ein Konto anlegen, indem ein eindeutiger Nutzernamen und ein Passwort vergeben werden. Mit diesen Daten können sich Nutzende einloggen.

Nach dem Einloggen können Nutzende sich einen Avatar zuordnen. Wenn das nicht erfolgt, wird ein Platzhalter zugeordnet.

Nutzende haben die Möglichkeit, ihr Passwort zu ändern.

2. Lieblingsfilme

Nutzende können nach Filmen suchen und diese als Favoriten markieren. Entscheiden Sie, ob Sie eine Obergrenze definieren, wie viele Filme sich eine nutzende Person als Favoriten markieren darf.

3. Filmabend planen

Eine nutzende Person kann einen Filmabend planen und muss dazu mindestens folgende Parameter definieren:

- Datum und Uhrzeit
- Ort
- Liste von teilnehmenden Personen, durch Auswahl vorhandener Nutzer
- Genre des zu schauenden Films

Wenn alle Parameter vorhanden sind, werden passende Filme berechnet und diesem Filmabend zugeordnet. Die Berechnung der Filmliste soll die Favoriten aller teilnehmenden Personen und das definierte Genre berücksichtigen.

Legen Sie eine maximale Anzahl der vorgeschlagenen Filme fest und entscheiden Sie, ob Favoriten der teilnehmenden Personen auch in den Vorschlägen vorkommen dürfen oder nicht oder ob es konfigurierbar ist.

Sie können weitere Parameter zu einem Filmabend definieren, die Sie für sinnvolle Erweiterungen für die Nutzenden halten.

4. Filmabende einsehen

Nutzende haben eine Übersicht der anstehenden Filmabende, denen sie zugeordnet sind.

Zu einem Filmabend darf eine teilnehmende Person hinterlegen, ob sie teilnimmt und wenn ja, für Vorschläge stimmen.

Sie entscheiden, wie der Abstimmungsmodus aussieht, z.B. ob eine teilnehmende Person eine oder mehrere Stimmen hat oder ob Punkte oder Sterne vergeben werden, etc.

Sie entscheiden weiterhin, ob das System die Abstimmung **auswertet und den „Ergebnisfilm“ bestimmt** oder ob das System nur eine Zusammenfassung der Abstimmung zeigt und die teilnehmenden Personen „offline“ entscheiden.

Filmdaten und Vorschläge

Die Filmdaten sollen von einer externen API kommen und Sie dürfen abgefragte Filmdaten in der Datenbank zwischenspeichern, wenn Sie das für sinnvoll halten. Auch für die Vorschläge sollen Sie eine externe API verwenden. Es steht Ihnen frei, das Ergebnis der API zu modifizieren.

Recherchieren Sie geeignete APIs, die die Anforderungen erfüllen, und entscheiden Sie, welche Sie verwenden wollen. Hier eine kleine Auswahl von Beispielen:

- [IMDb API](#)
- [Unofficial IMDb](#)
- [trakt API](#)
- [The movie DB](#)
- [OMDb API](#)
- [Advanced Movie Search](#)
- [Myritebook](#)
- [OTT details API](#)
- [Tasteditive](#)

Wichtig: Schauen Sie sich die Nutzungsbedingungen der APIs an, die Sie verwenden und halten Sie sich an diese.

Dokumentation

Die Dokumentation sollte 8-10 Seiten (ohne Deckblatt und Verzeichnisse) umfassen und die Ergebnisse der Software beschreiben. Der Arbeitsauftrag ist genauer beschrieben, inklusive formulierter Anforderungen. Technische sowie fachliche Entscheidungen und Aspekte sollen nachvollziehbar beleuchtet werden. Auch die Architektur der Software soll mit geeigneten Mitteln, bspw. UML, beschrieben werden.

Auf der obersten Ebene des Projekts liegt eine *readme.md*-Datei, welche kurz beschreibt, wie das Projekt aufgebaut und zu starten ist.

Technisches

Das Spring-Boot-Backend und die zugehörige Datenbank sollen im Produktivbetrieb in der Microsoft Azure Cloud laufen.

Entscheiden Sie, ob Sie eine Web-UI oder einen JavaFX-Klienten entwickeln, der mit dem Spring-Boot-Backend per REST kommuniziert. Es muss möglich sein, dass mehrere Klienten mit einem Spring-Boot-Backend kommunizieren können.

Sicherheit ist wichtig, hier in diesem Projekt brauchen Sie aber nicht hohen Sicherheitsstandards genügen, d.h. Sie brauchen nicht auf Transportsicherheit zu achten (HTTPS) und dürfen zum Beispiel HTTP-Basic verwenden. Dokumentieren Sie die möglichen Schwachstellen. Allerdings speichern Sie keine unverschlüsselten Passwörter in der Datenbank ab, sondern verwenden Sie z.B. einen Hash-Algorithmus.