

Projektplanung – Film-Datenbank

Brainstorming

Hauptfunktionen:

- **Film hinzufügen:** Filme mit Titel, Genre und Erscheinungsjahr speichern.
- **Filme anzeigen:** Alle gespeicherten Filme anzeigen.
- **Filme löschen:** Filme aus der Datenbank entfernen.

Technische Anforderungen:

- **Datenmodell:** `Movie`-Klasse für Filmobjekte.
- **Datenbank:** MySQL (MariaDB) zur Speicherung.
- **Benutzeroberfläche:** JavaFX.
- **Entwicklung:** Maven, JDBC, IntelliJ.

Allgemeine Beschreibung des Projektziel

Das Ziel des Projekts ist es, ein System zur Verwaltung von Filmen zu entwickeln. Benutzer sollen Filme mit Titel, Genre und Erscheinungsjahr hinzufügen, anzeigen und löschen können. Das System verwendet eine grafische Benutzeroberfläche und speichert die Daten in einer MySQL-Datenbank.

Ist-Analyse

Derzeit existiert keine einfache Möglichkeit, Filme mit Basisinformationen wie Titel, Genre und Erscheinungsjahr zu speichern und zu verwalten. Es gibt keine zentrale Datenbank für die Speicherung.

Soll-Analyse

Das Ziel des Projekts ist die Entwicklung einer benutzerfreundlichen Anwendung zur Verwaltung von Filmen. Die Software ermöglicht es, Filme mit Titel, Genre und Erscheinungsjahr hinzuzufügen, anzuzeigen und zu löschen. Eine intuitive grafische Benutzeroberfläche wird mit JavaFX bereitgestellt. Die Daten werden in einer MySQL-Datenbank gespeichert, auf die über JDBC zugegriffen wird. Eine Tabellenansicht bietet die Möglichkeit der Anzeige aller Daten, um gespeicherte Filme effizient zu verwalten.

Die Anwendung validiert Benutzereingaben, um fehlerhafte Daten wie leere Felder oder ungültige Jahreszahlen zu vermeiden. Fehlerbehandlungen und aussagekräftige Fehlermeldungen gewährleisten eine stabile Nutzung. Das System wird modular aufgebaut, um zukünftige Erweiterungen zu ermöglichen. Eine strukturierte Dokumentation beschreibt die Implementierung und erleichtert die Nutzung der Software.

Aufgabenaufteilung - Milestones

Tag 1 - Projektplanung: Themenwahl, Erstellung der Projektplanung und Diagramme (UML-Klassen, ER-Diagramm, PAP).

Tag 2 - Setup und Datenbankdesign: Projektstruktur einrichten, Maven konfigurieren, `movies`-Tabelle in MariaDB erstellen.

Tag 3 - JDBC-Integration: Verbindung zur Datenbank herstellen, SQL-Methoden für CRUD vorbereiten.

Tag 4 - Model-Klasse: `Movie`-Klasse implementieren und mit SQL-Methoden verbinden.

Tag 5 - GUI-Start: JavaFX-Formular für Filmeingaben erstellen (Titel, Genre, Jahr).

Tag 6 - CRUD-Integration: GUI mit Backend verbinden, Filmeingaben speichern und anzeigen lassen.

Tag 7 - Anzeige und Löschung: Tabellenansicht in der GUI für gespeicherte Filme und Löschfunktion implementieren.

Tag 8 - Fehlerbehandlung und Validierung: Eingaben validieren, Benutzerwarnungen für ungültige Daten hinzufügen.

Tag 9 - Optimierung: GUI verbessern (Design, Benutzerfreundlichkeit), SQL-Abfragen optimieren.

Tag 10 - Projektabgabe: Letzte Tests durchführen, finale Optimierungen und Abgabe des Projekts.