

1. Projektname

- DasBuch
-

2. Allgemeine Beschreibung

Eine Webplattform zur Erfassung und Verwaltung von Buchrezensionen. Ziel ist es, ein professionelles System mit einer RESTful API zu erstellen, das die PostgreSQL-Datenbank mit dem Front-End verbindet. Erweiterte Suchfunktionen und Datenmanipulation über CRUD-Operationen werden implementiert.

3. Hauptfunktionen

1. Vollständiges CRUD für Rezensionen (Erstellen, Lesen, Aktualisieren, Löschen).
 2. Suchfilter:
 - Suche nach Name.
 - Suche nach Genre.
 - Suche nach Autor.
 - Suche nach Lesezeitraum.
 - Suche nach Bewertung (Operatoren: >=, <=, =).
 - Sortieroptionen: Buchname, Autor, Bewertung und Lese-Datum.
 3. Datenanzeige:
 - Paginierte Liste der Rezensionen.
 - Detaillierte Ansicht jeder Rezension.
 4. Internationalisierung (i18n):
 - Mehrsprachige Unterstützung von Anfang an implementiert.
 5. Weboberfläche:
 - EJS für Node.js.
 - Razor Pages für .NET.
 - Front-End erstellt mit BootstrapStudio.
-

4. Datenstruktur (Review-Objekt)

Jede Rezension hat folgende Attribute:

- Buchname (String)
- Autorennamen (String)

- Genre (String)
 - Kommentar (String)
 - Bewertung (Ganzzahl: 1-5)
 - Lese-Datum (Datum)
-

5. Systemansichten (Views)

1. Startseite (Rezensionsliste)

- Zeigt die vollständige Liste der Rezensionen an.
- Filterung nach:
 - Genre
 - Autor
 - Lesezeitraum
 - Bewertung (\geq , \leq , $=$)
 - Sortieroptionen
- Aktionen:
 - Rezension hinzufügen (Link zur Erstellungsseite).
 - Details anzeigen (Link zur Detailseite).
 - Bearbeiten und Löschen direkt aus der Liste.

2. Rezension-Erstellungsseite

- Formular zur Erstellung einer neuen Rezension:
 - Buchname.
 - Autorenname.
 - Genre.
 - Kommentar.
 - Bewertung (1-5).
 - Lese-Datum.

3. Detailansicht der Rezension

- Vollständige Anzeige aller Informationen zu einer Rezension.
- Buttons zum Bearbeiten oder Löschen der Rezension.

4. Rezension-Bearbeitungsseite

- Formular wie die Erstellungsseite, aber mit vorausgefüllten Daten.

5. Erweiterte Suchseite

- Seite für detaillierte Suchoptionen.
- Ergebnisse werden als Liste angezeigt.

6. Fehler/404-Seite

- Generische Fehlerseite (z.B. „Rezension nicht gefunden“, ungültige Routen, etc.).
-

6. Verwendete Technologien

1. Projektmanagement:

- Trello

2. Back-end:

- NodeJS/Express
- .NET (C#)
- PostgreSQL-Datenbank

3. Front-end:

- EJS für NodeJS
- Razor Pages für .NET
- BootstrapStudio

4. Internationalisierung:

- i18n Standard für Mehrsprachigkeit.

5. Versionskontrolle:

- Git/GitHub

6. Agile Methodik:

- Sprint-Planung mit iterativen Lieferungen.