### Web-Applikation Buchverleih

#### Aufgabe:  
Entwickeln Sie eine Web-Applikation, die eine einfache Verwaltung eines Buchverleihsystems abbildet. Die Applikation soll eine Datenbank enthalten, die Entitäten mit \*\*1:1\*\*, \*\*1:n\*\* und \*\*n:m\*\* Beziehungen modelliert. Sie sollen für die Erstellung des Backends und Frontends geeignete Technologien verwenden.

#### Anforderungen:

1. \*\*Entitäten & Beziehungen:\*\*  
   - \*\*Benutzerprofil  
    - Ein Benutzer hat genau ein Profil mit zusätzlichen Informationen (z.B. Adresse, Telefonnummer).  
 - Die Benutzer-IDs starten erst bei 1200  
 - Das Benutzerpasswort muss mindestens fünf Zeichen lang sein und ein Sonderzeichen enthalten.  
\*\*Bonus (optional): Speichere das Passwort gehasht in der Datenbank.  
   - \*\*Bücher  
    - Ein Benutzer kann mehrere Bücher ausleihen, aber ein Buch kann immer nur von einem Benutzer zu einer Zeit ausgeliehen werden.  
   - \*\*Autoren   
    - Ein Buch kann von mehreren Autoren geschrieben werden und ein Autor kann mehrere Bücher geschrieben haben.

2. \*\*Datenbankstruktur:\*\*  
 - Erstellen Sie eine Datenbank namens „Buchverleih“.  
   - Erstellen Sie die Tabellen für Benutzerprofile, Bücher und Autoren.   
   - Setzen Sie die entsprechenden Fremdschlüssel und Indizes für die Beziehungen   
 - Erstelle ein entsprechendes Entity-Relationship-Diagramm (ER-Diagramm) und markiere die Beziehungen zwischen den Tabellen (1:1, 1:n, n:m).  
 - Befüllen Sie jede Tabelle mit mindestens drei Einträgen  
 - Dokumentieren Sie die SQL-Befehle für die CREATEs und jeweils ein INSERT pro Tabelle.

Beispiel Benutzerprofil:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Benutzerprofil-ID | Vorname | Nachname | Adresse | Passwort | E-Mail |
| 1203 | Maximilian | Mustermann | Adresse 20 | aDfgts! | m.muster@mail.com |
| 1204 | Maria | Musterfrau | Adresse 57 | Abcde§ | Musterfrau.m@mail.at |

Beispiel Bücher:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ISBN-10 | Buchtitel | Erscheinungsjahr | Verfügbar | Ausgeliehen von |
| 0241984750 | The Catcher in the Rye | 1951 | False | 1203 |
| 3453877608 | Der Talisman | 1984 | True |  |
| 9781444720723 | The Shining | 1977 | False | 1204 |

Beispiel BuchAutor:

|  |  |
| --- | --- |
| Buch | Autor |
| 3453877608 | 401 |
| 3453877608 | 402 |
| 9781444720723 | 401 |
| 0241984750 | 400 |

Beispiel Autor:

|  |  |  |
| --- | --- | --- |
| Autor-ID | Vorname | Nachname |
| 400 | Jerome David | Salinger |
| 401 | Stephen | King |
| 402 | Peter | Straub |

- Dokumentieren Sie die SELECT-SQL-Befehle mit diesen Ausgaben:

1. Alle Bücher, die Steven King geschrieben hat.
2. Alle Bücher, die mit „The“ anfangen.
3. Alle Bücher, die nach 1960 und vor 1980 erschienen sind.
4. Die Anzahl an Bücher, die mehr als einen Autor haben.
5. Die Anzahl an Benutzern, die gerade mindestens ein Buch ausgeliehen haben.

- Dokumentieren Sie den SQL-Befehl, um das zu erreichen:  
Musterfrau Maria möchte, dass ihre E-Mail auf [maria.musterfrau@mail.com](mailto:maria.musterfrau@mail.com) geupdated wird.

----------------------------------------------------------------------------------------------------------------

3. \*\*Funktionen der Web-Applikation:\*\*  
   - \*\*Benutzerverwaltung\*\*: CRUD (Create, Read, Update, Delete) für Benutzer und Benutzerprofile.  
   - \*\*Bücherverwaltung\*\*: Benutzer können Bücher ausleihen und zurückgeben.  
   - \*\*Autorenverwaltung\*\*: CRUD für Autoren und Zuordnung von Autoren zu Büchern.  
   - \*\*Buchsuche\*\*: Implementieren Sie eine einfache Suchfunktion, um Bücher nach Titel oder Autor zu suchen.

4. \*\*Technische Anforderungen:\*\*  
   - Wählen Sie geeignete Technologien für das Backend und Frontend.  
   - Die Applikation soll über eine API kommunizieren.  
 - Die Benutzeroberfläche soll sich visuell an ein Handy- oder Desktopformat anpassen.   
   - Authentifizierung: Implementieren Sie eine einfache Authentifizierung für den Benutzerzugang (Login).   
  
 - Erstellen Sie eine Webseite, mit der sich der Benutzer per E-Mail und Passwort anmelden kann. Die Felder für das Passwort und die E-Mail dürfen nicht leer sein.  
 - Wenn der Benutzer erfolgreich angemeldet wurde, wird er zur nächsten Seite weitergeleitet.  
 - Die nächste Seite zeigt den Benutzervor- und Nachname in der linken oberen Ecke an.  
 - Es gibt ein Suchfeld für Buchnamen. Auch wenn nur „Cacher“ eingegeben wird, soll „The Catcher in the Rye“ gefunden werden.  
 - Zusätzlich ist eine Dropdown-Liste mit allen Bücher da.  
 - Wenn der Benutzer ein Buch auswählt oder im Suchfeld eingibt, wird in einer unteren Tabelle alle Daten zum Buch ausgeben. Wenn das Buch ausgeliehen ist, ist die Zeile Rot. Bei multiplen Autoren, werden die Autornamen mit Beistrich getrennt in derselben Zelle ausgegeben.  
 - Wenn in das Suchfeld geklickt wird, wird die Dropdown-Liste auf Default gesetzt und es gibt keine Auswahl durch das Dropdown.

5. \*\*Bonus (optional):\*\*  
   - Implementieren Sie eine Verleihhistorie für jedes Buch.  
   - Fügen Sie eine Filterfunktion für die Buchsuche hinzu (z.B. nach Autor, Ausleihstatus).

#### Bewertungskriterien:  
- Saubere Implementierung der 1:1, 1:n und n:m Beziehungen in der Datenbank.  
- Funktionalität und Benutzerfreundlichkeit der Weboberfläche.  
- Codequalität (Lesbarkeit, Struktur, Kommentare).  
- Nutzung von Best Practices (z.B. Sicherheit bei der Authentifizierung).