

## Datenbanksysteme 2. Praktisches Projekt

### Einführung

Im Laufe des praktischen Projekts sollen Sie ein Online-Tagebuch entwickeln. Die Entwicklung dieses Projekts soll dabei in vier Schritten ablaufen:

1. ER-Modellierung der Datenbank
2. Überführung der ER-Modellierung in ein relationales Modell
3. Implementierung der Datenbank in SQLite
4. Umsetzung des dazugehörigen Anwendungsprogramms in Java

Die einzelnen Arbeitsschritte bauen hierbei aufeinander auf. Die aktuelle Aufgabenstellung wird im letzten Abschnitt des Kapitels „Aufgabenstellung“ beschrieben.

### Wichtige allgemeine Hinweise

Für die ersten drei Schritte haben Sie jeweils einen Bearbeitungszeitraum von einer Woche, für die abschließende Implementierung haben Sie drei Wochen Zeit. Um die praktischen Übungen zu bestehen, müssen Sie *jeden* der vier Abschnitte bestehen. Dies bedeutet, dass Sie den zweiten Schritt nur bearbeiten können, wenn Sie den ersten *bestanden* haben. Klären Sie daher kritische Punkte und Fragen *vor* der Abgabe in den Sprechstunden, da es aufgrund des Zeitplans keine Nachbearbeitungszeit geben kann.

Sie müssen alle Arbeitsschritte *alleine* und selbstständig bearbeiten. Gruppenarbeiten, auch bei nicht ganz identischen Abgaben, führen zum Ausschluss aller Beteiligten.

Die Abgabe dieses vierten Teils ist bis Montag, den 10.12.2018, um 10:00 Uhr über das Abgabesystem möglich. Achten Sie auf die **aktuelle Aufgabenstellung** am Ende des Übungsblattes.

# GUI

Die Lösung des letzten Abschnittes des Abschlussprojektes sieht die Entwicklung eines Programms vor, welches eine Datenbankverwaltung ermöglichen soll. Dieses Programm soll eine graphische Oberfläche besitzen. Wir stellen eine mögliche Oberfläche zur Verfügung. Diese ist speziell für das Projekt von Alexander Thelen entwickelt worden. Sie liegt in einem Repository bereit, welches außerdem eine Anleitung enthält, die die ersten Arbeitsschritte vereinfachen soll. Das Package kann im Uni-internen GIT heruntergeladen werden <sup>1</sup>. Es enthält das Projekt, sowie eine Anleitung. Wir stellen es jedem Studenten frei, zwischen unserer GUI und der eigenen Lösung zu wählen. **Bitte berücksichtigen Sie, dass die Anleitung einen Teil der Aufgabenstellung enthält.** Jede angelegte Tabelle (außer Account) muss über das Suchfeld durchsuchbar sein. Auch müssen die vorgegebenen Buttons funktionieren. Weitere Informationen sind der Anleitung zu entnehmen.

newpage

## Anwendungsszenario

Das Ziel der gesamten praktischen Übung besteht in der Implementierung eines Online-Tagebuchs. Die explizite Aufgabe für dieses Blatt finden Sie weiter unten.

Es sollen folgende Eigenschaften dargestellt werden:

Ein **Benutzer** besitzt einen Vor-, sowie Nachnamen, eine eindeutige E-Mail-Adresse und ein Passwort.

Ein **Autor** ist ein Benutzer. Ein Benutzer kann beliebig viele Autoren abonnieren. Ein Autor kann von beliebig vielen Benutzern abonniert werden. Der Autor kann beliebig viele andere Autoren empfehlen, sowie von beliebig vielen anderen Autoren empfohlen werden. Der Autor besitzt ein Pseudonym und ein optionales Avatar. Er legt einen Preis für sein Tagebuch fest, sofern es mindestens eine private Seite enthält (sonst gilt Preis = 0). Der Benutzer kann zu jedem Autor genau eine Bewertung abgeben.

Eine Bewertung besteht aus einem Text und einer Benotung (1-5 Sterne).

Ein Autor kann jeden Tag (Datum) genau eine **Seite** in seinem Tagebuch anlegen. Jede erstellte Seite hat eine Nummer und einen Typ (privat/öffentlich). Jede Seite wird von genau einem Autor angelegt. Sie enthält mindestens einen Eintrag.

Ein Autor kann beliebig viele **Einträge** schreiben. Jeder Eintrag wird von genau einem Autor geschrieben und enthält einen Titel, eine Uhrzeit, einen Text und gehört zu genau einer Seite (nur einer eigenen Seite des Autors). Außerdem kann ein Eintrag beliebig viele **Bilder** enthalten.

Jedes **Bild** gehört zu genau einem Eintrag. Das Bild wird in der Datenbank als Blob abgelegt. Zu manchen Bildern werden die GPS-Koordinaten (Längen- und Breitengrad) gespeichert.

Zu jedem Bild können beliebig viele **Tags** gesetzt werden. Jedes Tag ist eindeutig und kann beliebig oft verwendet werden. Sowohl die Tags als auch die Bilder werden von Autoren erstellt/zugewiesen/hochgeladen.

Ein Benutzer führt **Transaktionen** unter Angabe von Zweck aus. Optional kann zu einer Transaktion ein Gutscheincode angegeben werden. Mit dem Gutscheincode können die Tagebücher

---

<sup>1</sup><https://git.hhu.de/cs/dbs/datenbanksysteme/programmierpraktikum/Project>

ebenfalls erworben werden, sofern der Wert ausreicht. Ist der Gutscheinwert höher als der Preis eines Tagebuches, freut sich der Autor ganz besonders über den Verkauf.

Jede **Transaktion** enthält einen Betrag, ein Datum und eine Zahlungsmittelwahl (z.B. Paypal). Jede Transaktion wird von genau einem Autor empfangen.

## 4. Aufgabenteil

Implementieren Sie das System in Java unter Verwendung der vorgegebenen oder einer eigenen GUI auf Basis Ihrer im dritten Teil abgegebenen Datenbank. Hierbei soll Ihr System folgende Anforderungen erfüllen:

- Ein neuer Benutzer soll die Möglichkeit haben sich zu registrieren.
- Ein registrierter Benutzer soll die Möglichkeit haben sich anzumelden.
- Ein registrierter Benutzer soll die Möglichkeit haben eine Änderung seiner Daten vorzunehmen.
- Ein registrierter Benutzer soll die Möglichkeit haben öffentliche Seiten von beliebigen Tagebüchern zu lesen.
- Ein registrierter Benutzer soll Zugriff zu allen Seiten eines Tagebuches erhalten, sofern er diesen erworben hat.
- Ein registrierter Benutzer soll Transaktionen durchführen können und seine durchgeführten Transaktionen einsehen können.
- Ein Autor soll einen Avatar hochladen können.
- Ein Autor soll Tags erstellen und zuweisen können.
- Ein Autor soll Bilder hochladen und zuweisen können.
- Ein Autor soll sein Tagebuch führen können. Es soll auch eine spätere Korrektur der Seiteninhalte soll möglich sein.
- Ein Autor darf bereits verkaufte private Seiten seines Tagebuches nicht mehr löschen. Eine Bearbeitung soll nur im Sinne der Ergänzung möglich sein.
- Ein Autor soll seinen aktuellen Gesamtverdienst einsehen können.
- Es soll möglich sein, nach Autoren zu suchen.
- Es soll möglich sein, Autoren zu bewerten und die Bewertungen anderer Benutzer zu sehen.
- Es soll möglich sein, sich Einträge anzeigen zu lassen, die an einem bestimmten Datum entstanden sind und einen bestimmten Titel haben.
- Es soll möglich sein, sich Bilder anzeigen zu lassen, die in einem bestimmten GPS-Bereich liegen (Koordinaten [von, bis]).
- Es soll möglich sein, nach einem oder mehreren Tags zu suchen und sich Bilder zu Einträgen auf öffentlichen Seiten anzeigen zu lassen. Bei privaten Inhalten soll nur das Vorhandensein dieser Information angezeigt werden und die Möglichkeit gegeben werden, diese zu erwerben.

Achten Sie bitte darauf, dass Sie die gegebenen Anforderungen von der Aufgabenstellung der letzten Übungsblätter nicht losgelöst betrachten.

Geben Sie abschließend im Abgabesystem eine ZIP-Datei mit folgendem Inhalt ab:

- Ihr System als ausführbare JAR-Datei. Diese Datei sollte vorher auf einem Rechner im Rechenpool auf Lauffähigkeit getestet werden. Nutzen Sie bitte hierfür die vorgesehenen Sprechzeiten. Ihre Zugangsdaten bekommen Sie in der Sprechstunde. Kann die abgegebene Datei nicht richtig gestartet werden, gilt Ihr Projekt als nicht bestanden. Achten Sie bitte weiterhin auf eine ausreichende Eingabedatenvalidierung. (Zum Beispiel: gültige E-Mail-Adresse, falls dies in der Datenbank nicht vorgenommen wurde)
- Ihre verwendete und weiterhin mit Beispieleinträgen gefüllte SQLite Datenbank.