

## **Abstract – Breakdown des Projektes: Data-Mart-Erstellung in SQL**

### **Aufgabenstellung**

Dieses Projekt umfasst die Konzeption und Implementierung einer relationalen Datenbank für eine Buchtausch-App, die den Austausch von Büchern innerhalb einer lokalen Gemeinschaft ermöglicht. Die Nutzer dieser Plattform können Bücher, die sie besitzen, zur Ausleihe anbieten und gleichzeitig Bücher von anderen Nutzern ausleihen. Ziel der Anwendung ist es, den Nutzern eine einfache Verwaltung der Buchausleihe und -rückgabe zu ermöglichen. Das Projekt wurde unter Verwendung von MySQL als Datenbank-Management-System (DBMS) und DBeaver als Entwicklungsumgebung umgesetzt.

### **Datenbankdesign**

Die Datenbank wurde gemäß den Anforderungen entworfen und umfasst Tabellen, die Informationen über Bücher, Nutzer, Adressen, Ausleihvorgänge, Bewertungen und Nachrichten zwischen den Nutzern speichern. Die Datenbankstruktur folgt einem relationalen Modell, das durch Entity-Relationship-Modellierung (ERM) mit Lucid Chart erstellt wurde. Es wurden vierzehn Entitäten erstellt. Um sicherzustellen, dass die Datenbank effizient und skalierbar ist, wurden die Tabellen in der 3. Normalform normalisiert, um Redundanzen und Inkonsistenzen zu minimieren.

### **Technische Implementierung**

Die Implementierung der Datenbank erfolgte mit MySQL. Für die Entwicklung wurde DBeaver genutzt, ein vielseitiges Tool zur Verwaltung von Datenbanken. Die Datenbank wurde mit SQL-Skripten erstellt, die die Tabellen und Beziehungen definierten, einschließlich Fremdschlüssel, um referenzielle Integrität zu gewährleisten. Dummy-Daten wurden in die Tabellen eingefügt, um die Funktionalität zu testen und realistische Abfragen zu ermöglichen. Ereignisse wie die tägliche Aktualisierung des Buchstatus (z.B. "reserviert" oder "ausgeliehen") wurden durch SQL-Events automatisiert.

Zusätzlich wurde ein Repository auf GitHub erstellt, welches den SQL-Code der implementierten Datenbank erhält. Der Link dazu findet sich in dem 3-seitigen PDF Dokument, welches im dritten Teil der Abgabe enthalten ist.

### **Tests und Ergebnisse**

Die Datenbank wurde auf Basis der Dummy-Daten mit verschiedenen Testfällen überprüft. Abfragen, wie die Anzeige verfügbarer Bücher oder die Bewertungshistorie eines Nutzers, wurden erfolgreich durchgeführt. Die Struktur der SQL-Befehle wurde dokumentiert und optimiert, um die

Datenintegrität zu gewährleisten. Zudem wurden alle notwendigen Schritte für die Reproduktion der Datenbankumgebung dokumentiert, einschließlich einer Installationsanleitung.

Abschließend wurden die Ergebnisse in gewünschter Form in einem ZIP-Ordner zusammengefasst und über PebblePad hochgeladen.