

Abstract

Im Rahmen des Projektes wurde eine relationale Datenbank für eine **Buchtausch-App** entwickelt. Ziel war es, eine Plattform abzubilden, auf der Benutzer:innen eigene Bücher zum Ausleihen anbieten und gleichzeitig verfügbare Werke anderer Personen ausleihen können. Die Lösung soll den gesamten Lebenszyklus einer Ausleihe unterstützen – von der Erfassung bibliographischer Daten bis zur Rückgabe – und durch Funktionen wie Reservierungen, Bewertungen und Kommunikation eine praxisnahe Nutzung ermöglichen. Ein besonderes Merkmal der Aufgabenstellung war zudem die **räumliche Suche nach verfügbaren Exemplaren**, sodass Nutzer:innen über eine Kartenansicht Angebote in ihrer Nähe identifizieren können. Die Bearbeitung des Projektes erfolgte in drei Phasen.

In der **Konzeptionsphase** wurde zunächst eine detaillierte Anforderungsspezifikation erstellt. Dabei wurden die Rollen Benutzer, Verleiher, Ausleiher und Administrator definiert sowie deren zentrale Aktionen wie Registrierung, Buchverwaltung, Ausleihe, Rückgabe und Support beschrieben. Darauf aufbauend entstand ein **Entity-Relationship-Modell** mit 21 Entitäten, mehreren m:n-Beziehungen und zwei Dreifachbeziehungen (u. a. für Ausleihe und Reservierung). Ein Datenwörterbuch ergänzte das Modell um die fachliche Beschreibung und Datentypen der Attribute.

Die **Erarbeitungs- und Reflexionsphase** diente der technischen Umsetzung. Das entwickelte ER-Modell wurde in MariaDB in ein vollständiges relationales Schema überführt. Alle Tabellen wurden in **dritter Normalform** umgesetzt, teilweise bis zur BCNF, wodurch Redundanzen vermieden und die referenzielle Integrität sichergestellt wurde. Jede Entität wurde mit mindestens zehn Dummy-Datensätzen befüllt, sodass realistische Testfälle – von Ausleihen über Bewertungen bis hin zu Chatnachrichten – abgebildet werden konnten. Zur Überprüfung der Funktionalität wurden für alle Tabellen SQL-Statements entwickelt, die unterschiedliche Features von SQL demonstrierten, darunter komplexe JOINS, Aggregationen, Views sowie Constraints. Dadurch entstand ein lauffähiges System, das die Anforderungen der Aufgabenstellung vollständig erfüllte.

In der **Finalisierungsphase** wurde das Schema verfeinert, indem CHECK-Constraints, Default-Werte und zusätzliche UNIQUE-Bedingungen ergänzt wurden. Außerdem wurde die Spalte *verleiher_id* aus der Tabelle *Ausleihe* entfernt, da diese Information bereits über die Beziehung zu *BuchExemplar* ableitbar war.

Parallel dazu wurde die Geo-Funktionalität deutlich ausgebaut: Adressen erhielten **Koordinaten (lat/lon)**, Indizes und Plausibilitäts-Checks wurden ergänzt, und es wurden **Views und Stored Procedures** für Umkreissuchen, Bounding-Box-Filter und kombinierte Text- und Radiusabfragen implementiert. Damit geht die Lösung über die ursprüngliche Aufgabenstellung hinaus und bietet einen erweiterbaren Ansatz für eine spätere Integration in eine App mit Kartenansicht.

Technische Herangehensweise (Breakdown): Die Umsetzung erfolgte schrittweise. Zunächst wurde das relationale Schema in SQL (CREATE-Skripte) definiert und in MariaDB getestet. Anschließend wurden alle Tabellen konsistent mit Dummy-Daten befüllt (INSERT-Skripte) und die Funktionalität durch gezielte Testabfragen (SELECT, JOIN, GROUP BY etc.) überprüft. Auf Basis des Feedbacks und eigener Analysen wurden Optimierungen am Datenmodell vorgenommen (Constraints, Redundanzabbau, Normalisierung). Schließlich wurde die Geo-Erweiterung als separates SQL-Modul implementiert, das sowohl Indizes als auch eine Haversine-Funktion für Distanzberechnungen enthält. Die Projektergebnisse zeigen eine konsistente, normalisierte und erweiterbare Datenbank, die die Kernprozesse einer Buchtausch-App realistisch und performant abbildet. Durch die Integration geografischer Funktionen wird die Lösung praxisorientiert erweitert und kann als solide Grundlage für eine reale Anwendung dienen.