

Funktionalitäten des Datenbankmanagementsystems (DBMS)

1. Wahl des DBMS

Für die Erstellung der Datenbank wurde das relationale Datenbankmanagementsystem MySQL genutzt.

1.2 Begründung der Wahl

Vorteile von MySQL:

MySQL ist Open-Source, was bedeutet, dass es kostenlos genutzt und angepasst werden kann. Dies ist besonders für Start-ups, kleine Unternehmen und Entwicklerprojekte vorteilhaft, die eine leistungsfähige Datenbanklösung ohne teure Lizenzkosten benötigen. Allerdings gibt es auch professionelle Anwendungen für kommerzielle Versionen, die zusätzliche Support- und Sicherheitsfunktionen bieten.

Die einfache Installation und Konfiguration sorgt für einen unbeschwerten Start. Es erfordert weniger Systemressourcen als viele andere Datenbankmanagementsysteme, was es besonders für kleine bis mittlere Projekte oder Anwendungen geeignet macht. Ein weiterer Vorteil ist, dass es mit Tools wie MySQL Workbench und phpMyAdmin auch über eine grafische Benutzeroberfläche verwaltet werden kann.

Es bietet auch Mechanismen für die horizontale und vertikale Skalierung, sodass es sich an wachsende Anforderungen anpassen kann. MySQL bietet durch die InnoDB Storage Engine eine stabile und zuverlässige ACID-konforme Transaktionsverarbeitung. Dadurch bleiben Daten bei Systemfehlern konsistent und sicher.

MySQL wird von vielen Programmiersprachen (wie PHP, Python, Java, C++) und Plattformen (wie LAMP-Stack, XAMPP) unterstützt, was es einfach macht, es in bestehende oder neue Projekte zu integrieren. Neben dem, ist es auch mit vielen Webanwendungen wie WordPress, Drupal und Joomla kompatibel.

MySQL bietet robuste Sicherheitsmechanismen wie Benutzerverwaltung, Zugriffsrechte auf Tabellenebene und SSL-Unterstützung zur sicheren Datenübertragung.

Da MySQL eines der am weitesten verbreiteten Datenbanksysteme ist, gibt es bereits eine große Community, die Tutorials, Foren, Dokumentationen und Unterstützung bietet. Dies erleichtert es Entwicklern, Lösungen für Probleme zu finden oder sich in MySQL einzuarbeiten. Sicherheitslücken werden in der Community und von Oracle oft schnell behoben, was eine fortlaufende Wartung und Sicherheit gewährleistet.

MySQL läuft auf verschiedenen Betriebssystemen wie Linux, Windows, macOS und Unix, wodurch es in unterschiedlichen Umgebungen verwendet werden kann. Es ist auch für den Einsatz in der Cloud optimiert und wird in vielen Cloud-Diensten wie Amazon Web Services (AWS) und Google Cloud Platform unterstützt.

Nachteile von MySQL:

MySQL bietet weniger komplexe SQL-Funktionen im Vergleich zu anderen DBMS wie PostgreSQL oder Oracle.

Obwohl MySQL Transaktionen unterstützt (mit der InnoDB-Engine), ist es in diesem Bereich im Vergleich zu anderen Datenbanklösungen, wie Oracle oder PostgreSQL, eingeschränkt. Komplexe Abfragen und Transaktionen können in MySQL weniger effizient sein, besonders wenn es um ein hohes Volumen an Lese- und Schreibvorgängen geht. InnoDB ist für viele Anwendungsfälle gut, kann aber bei sehr spezifischen Workloads, z.B. mit vielen gleichzeitigen Schreiboperationen oder großen Datenmengen, im Vergleich zu spezialisierten Lösungen an ihre Grenzen stoßen

Während MySQL für viele allgemeine Anwendungen gut geeignet ist, bietet es weniger Erweiterungsmöglichkeiten als andere Systeme wie PostgreSQL, das eine breitere Palette von Erweiterungen und Funktionen bietet. So ist MySQL beispielsweise primär ein relationales DBMS und bietet weniger native Unterstützung für NoSQL-Funktionen (wie dokumentbasierte Speicherung oder Key-Value-Speicherung) im Vergleich zu spezialisierten NoSQL-Datenbanken oder hybriden Systemen wie PostgreSQL.

Die Open-Source-Version von MySQL ist zwar kostenlos, bietet allerdings keinen offiziellen Support. Wird dieser professionelle Support benötigt, muss die kommerzielle MySQL-Version genutzt werden, was dann wiederum Kosten verursacht.

Fazit:

Die Vorteile wie eine einfache Installation, Benutzerfreundlichkeit und die kostenlose Verfügbarkeit und eine Anwendung auf unterschiedlichsten Betriebssystemen machen MySQL besonders für die Anforderungen dieses Projektes sehr interessant. Dazu ist es äußerst nützlich, auf eine bereits bestehende Community zurückgreifen zu können.

Die Nachteile, die sich hauptsächlich auf den Umgang mit großen Datenmengen und komplexen Operationen beschränken, sind für dieses Projekt nicht weiter relevant.

Daher fällt die Wahl des DBMS für dieses Projekt auf MySQL.

2. Metadaten der Datenbank „buchtausch_app“

Metadaten	Information
Tabellenanzahl	14
Volumen	496k
Zeichensatz	Utf8mb4

2.1 Metadaten der Tabellen

ID	Tabellenname	Spaltenanzahl	Einträge	Anzahl genutzter Datentypen
1	Adresse	5	10	3
2	Ausleihvorgang	6	6	3
3	Autor	3	16	2
4	Benutzer	7	10	3
5	Buchbewertung	5	11	2
6	Buecher	12	18	5
7	Leihdauer	2	5	1
8	Nachricht	5	17	3
9	Nutzerbewertung	5	12	2
10	Reservierung	6	6	4
11	Standortfreigabe	2	3	2
12	Verlag	2	11	2
13	Versandoption	2	2	2
14	Zustand	2	3	2

3. Link zum GitHub Repository

https://github.com/Tegge-10/DataMart_buchtauschapp

4. Installationsanleitung

Vorraussetzung für das Installieren der Datenbank ist, dass MySQL auf dem System installiert und konfiguriert ist. Des Weiteren muss ein Zugriff auf einen MySQL Benutzer mit den notwendigen Berechtigungen zum Erstellen von Datenbanken und Tabellen bestehen.

Dann sollte das Terminal, oder aber die genutzte Workbench geöffnet werden und sich, mit den entsprechenden Informationen – angemeldet werden.

Nun kann nach und nach der Code eingefügt werden. Zuerst wird die Datenbank entsprechend erstellt. Dann folgt die Erstellung der einzelnen Tabellen und der Dummy-Daten.