

Handreichung

DigiTeLL Partnership NoSQLconcepts



Stiftung
Innovation in der
Hochschullehre

Inhaltsverzeichnis

1	Einleitung	1
2	Zielgruppe	1
3	Voraussetzungen	2
3.1	Lernvoraussetzungen	2
3.2	Lehrvoraussetzungen	2
3.3	Technische Voraussetzungen	2
4	Lernziele	3
4.1	Art und Komplexität der Intended Learning Outcomes (ILOs)	3
4.2	Zielführende Anordnung, Reihenfolge und Kombination der ILOs	3
4.3	Abstimmung der digitalen Projektmaßnahmen auf die ILOs . .	4
5	Methode	5
5.1	Kompetenzorientierte Lehr-/Lernformate	5
5.2	Zusammenhang zwischen geplanten Studierendenaktivitäten und ILOs	5
5.3	Innovation für kompetenzorientierte, evidenzbasierte digitale Lehrformate	6
6	Anleitung zur Nutzung des NoSQLconcepts Tools	7
6.1	Installation	7
6.2	Anpassung von Komponenten	8
6.2.1	Dashboard-Karten	8
6.2.2	Interaktive Aufgabenblätter	9
6.2.3	Backend	9
6.3	Datenbank Design	9
6.4	Ausblick	10

1 Einleitung

Das Projekt NoSQLconcepts hat das Ziel, ein digitales Lerntool zu entwickeln und zu analysieren, das den Studierenden ein tiefgreifendes Verständnis verschiedener Datenmodelle und Datenbanken vermittelt, insbesondere solcher, die unter dem Begriff *NoSQL* bekannt sind. Im Gegensatz zum traditionellen tabellarischen SQL-Datenmodell bieten NoSQL-Datenbanken alternative Ansätze zur Organisation und Verwaltung von Daten.

Das Hauptziel des Projekts NoSQLconcepts besteht darin, ein fortgeschrittenes digitales Lerntool zu entwickeln, das über herkömmliche Bildungsdesigns hinausgeht. Unser digitales Lerntool zielt darauf ab, den Studierenden nicht nur NoSQL-Datenmodelle und -Datenbanken näherzubringen, sondern auch das Lernerlebnis durch verschiedene Funktionen zu verbessern. Dazu gehören die Vereinheitlichung des Datenbankzugriffs, adaptive Soforthilfen, individuelles und automatisiertes Feedback, die Bewertung von Benutzerfreundlichkeit und Erlernbarkeit, sowie die kontinuierliche Bewertung des Lernerfolgs.

Das Projekt strebt durch die Integration dieser Elemente die Schaffung eines effektiven und umfassenden Lerntools an, mit dem die Studierenden die Konzepte von NoSQL beherrschen können.

Die vorliegende Handreichung wurde entwickelt, um anderen Lehrenden die Nutzung des Tools zu erleichtern. Sie bietet eine umfassende Anleitung zur effektiven Implementierung und Anwendung des digitalen Lerntools NoSQLconcepts in Bildungsumgebungen. Darüber hinaus enthält die Handreichung eine Beschreibung der Lernziele und eines Learning Designs. Dies soll verdeutlichen, wie das Tool effektiv eingesetzt und in ein Lernkonzept integriert werden kann, um den Lernprozess der Studierenden zu fördern und ihre Kenntnisse im Bereich der NoSQL-Konzepte zu vertiefen.

2 Zielgruppe

Die Zielgruppe umfasst Studierende der Informatik und Wirtschaftsinformatik, sowie Lehrende im Bereich Datenbanken oder verwandter Fachrichtungen. Darüber hinaus richtet sich das Angebot an alle, die Interesse am Lernen und/oder Lehren von NoSQL-Datenbanken haben. Zukünftig werden weitere Anwendungsfälle hinzukommen, sodass die Zielgruppe entsprechend erweitert werden kann.

3 Voraussetzungen

3.1 Lernvoraussetzungen

Studierende sollten über grundlegende Kenntnisse in Datenbankkonzepten verfügen, einschließlich relationaler Datenbanken und SQL, um die Unterschiede und Vorteile von NoSQL-Konzepten besser zu verstehen.

Ein grundlegendes Verständnis der Funktionsweise von Computern und der Nutzung von Softwareanwendungen ist ebenfalls nötig, um das Lerntool nutzen zu können.

Zudem sollten Studierende ein Interesse an Datenbanktechnologien und NoSQL-Konzepten haben und bereit sein, eigenständig zu lernen und das Lerntool aktiv zu nutzen, um ihr Verständnis zu vertiefen.

3.2 Lehrvoraussetzungen

Lehrende sollten über fundierte Kenntnisse im Bereich Datenbanken verfügen, einschließlich relationaler Datenbanken und NoSQL-Datenbanken, um den Studierenden bei Fragen und Problemen kompetente Unterstützung bieten zu können. Neben dem Interesse an Datenbanken und NoSQL-Konzepten sollten Lehrende bereit sein, das Lerntool in ihren Lehrplan zu integrieren und den Studierenden auch bei der Nutzung des Tools Unterstützung anzubieten. Da das Tool auf Technologien wie JavaScript, React und Node.js basiert, wäre es für Lehrende hilfreich, über entsprechende Kenntnisse zu verfügen. Dies ermöglicht es ihnen, das Tool besser zu verstehen und bei Bedarf Anpassungen vorzunehmen.

3.3 Technische Voraussetzungen

Ein Zugang zu Computern oder anderen Geräten mit Internetverbindung ist erforderlich, um das digitale Lerntool verwenden zu können.

4 Lernziele

4.1 Art und Komplexität der Intended Learning Outcomes (ILOs)

Unser Learning Design zielt darauf ab, folgende Lernziele auf den Ebenen des Verständnisses, der Anwendung und der Evaluation zu erreichen. Studierende sollen in der Lage sein, verschiedene (NoSQL)Datenbanken zu unterscheiden und die zugrundeliegenden Datenmodelle zu kennen. Des Weiteren sollen sie befähigt werden, mit den verschiedenen (NoSQL)Datenbanken umzugehen und zielführende Datenbankabfragen zu formulieren. Schließlich sollen sie in der Lage sein, zu beurteilen, welche Datenbank für welches Anwendungsszenario geeignet ist. Anders formuliert lauten die Learning Outcomes:

- Studierende können verschiedene (NoSQL)Datenbanken unterscheiden und kennen die zugrundeliegenden Datenmodelle. (Verständnis-Ebene)
- Studierende können mit den verschiedenen (NoSQL)Datenbanken umgehen und zielführende Datenbankabfragen formulieren. (Anwendungs-Ebene)
- Studierende können beurteilen, welche Datenbank für welches Anwendungsszenario geeignet ist. (Evaluations-Ebene)

4.2 Zielführende Anordnung, Reihenfolge und Kombination der ILOs

Die Lerninhalte sind so strukturiert, dass die Studierenden schrittweise durch die verschiedenen Konzepte und Fähigkeiten geführt werden, beginnend mit dem Verständnis der Grundlagen, über die praktische Anwendung bis hin zur Bewertung und Auswahl der geeigneten NoSQL-Datenbanken für verschiedene Szenarien.

Studierende lernen zuerst innerhalb einer Einführungsveranstaltung die verschiedenen NoSQL-Datenbanken kennen. Dies bietet ein solides Fundament für ihr Verständnis und ihre Arbeit mit NoSQL-Datenbanken.

Nachdem die Studierenden ein grundlegendes Verständnis für die verschiedenen NoSQL-Datenbanken und ihre Datenmodelle entwickelt haben, sollten sie in der Lage sein, mit den Datenbanken umzugehen und zielführende Datenbankabfragen zu formulieren. Praktische Übungen und Projekte können

ihnen helfen, ihr Wissen anzuwenden und ihre Fähigkeiten in der Arbeit mit NoSQL-Datenbanken zu entwickeln.

Sobald die Studierenden die verschiedenen NoSQL-Datenbanken kennengelernt haben und praktische Erfahrungen gesammelt haben, sollten sie befähigt sein, zu beurteilen, welche Datenbank für welches Anwendungsszenario am besten geeignet ist. Dies erfordert ein tiefgreifendes Verständnis für die Stärken und Schwächen verschiedener Datenbanken sowie die Fähigkeit, fundierte Entscheidungen zu treffen.

Durch diese Reihenfolge und Kombination können die Studierenden schrittweise ihre Fähigkeiten und Kenntnisse im Umgang mit NoSQL-Datenbanken entwickeln und vertiefen, wodurch sie letztendlich in der Lage sind, komplexe Datenbankprobleme zu lösen und fundierte Entscheidungen zu treffen.

4.3 Abstimmung der digitalen Projektmaßnahmen auf die ILOs

Verständnis-Ebene:

- Bereitstellung von Vorlesungsmaterial (Präsentationsfolien), die Studierende durch die verschiedenen NoSQL-Datenbanken führen und zugrundeliegende Datenmodelle aufzeigen.
- Zusätzliche Tutorials¹, Quizze oder Selbsttests, um sicherzustellen, dass die Studierenden die Grundlagen der Datenbanken verstanden haben, bevor sie mit den praktischen Übungen fortfahren.

Anwendungs-Ebene:

- Bereitstellung von praktischen Übungen, bei denen die Studierenden verschiedene NoSQL-Datenbanken verwenden und Datenbankabfragen formulieren können.
- Einrichtung einer Umgebung (NoSQLconcepts Tool), in der die Studierenden frei experimentieren, Datenbankabfragen ausführen und ihre Fähigkeiten im Umgang mit den Datenbanken entwickeln können.

Evaluations-Ebene:

¹Bspw.: https://www.sqlzoo.net/wiki/SQL_Tutorial

- Bereitstellung von Anwendungsszenarien für ein Selfstudy Projekt, in denen die Studierenden bewerten müssen, welche NoSQL-Datenbank für bestimmte Anwendungsfälle am besten geeignet ist.
- Präsentation des Selfstudy Projekts und Durchführung von Gruppendiskussionen, in denen die Studierenden ihre Entscheidungen zur Auswahl einer Datenbank begründen und verteidigen müssen.

5 Methode

5.1 Kompetenzorientierte Lehr-/Lernformate

In unserem kompetenzorientierten Lehr-/Lernformat richten wir uns nach dem European Qualifications Framework (EQF) und definieren Lernergebnisse anhand von Kompetenzen, die in praxisrelevanten Situationen angewendet werden können. Innerhalb des Praktikums zum Thema Datenbankmanagementsysteme (DBMS) setzen wir auf konkrete Anwendungssituationen in den gestellten Aufgaben. Ein exemplarisches Szenario, das den Entwurf einer Datenbank für die Personalabteilung eines Unternehmens umfasst, bietet den Studierenden praxisnahe Einblicke. Um die Unterschiede zwischen verschiedenen NoSQL-Datenbanksystemen zu verdeutlichen, verwenden wir dasselbe Szenario und denselben Datensatz für jedes Datenbanksystem. Die Aufgabenblätter, die nun digital in unserem Lerntool bearbeitet werden können, sind so gestaltet, dass sie für jede Datenbank eine Reihe von Aufgaben mit einer einzigen Abfrage zur Lösung enthalten. Diese Aufgaben sollen die Stärken des jeweiligen Datenbanksystems hervorheben. Gleichzeitig werden auch einige anspruchsvollere Aufgaben für jede Datenbank bereitgestellt, um die Schwächen jedes Datenbanksystems zu verdeutlichen. Durch diesen Ansatz gewinnen die Studierenden Einblicke in die Vor- und Nachteile verschiedener NoSQL-Datenbanken (siehe auch [1]).

5.2 Zusammenhang zwischen geplanten Studierendenaktivitäten und ILOs

Als Learning outcomes wurden zuvor die folgenden definiert:

- Learning outcome (1) Studierende können verschiedene (NoSQL)Datenbanken unterscheiden und kennen die zugrundeliegenden Datenmodelle

- Learning outcome (2) Studierende können mit den verschiedenen (NoSQL)Datenbanken umgehen und zielführende Datenbankabfragen formulieren
- Learning outcome (3) Studierende können beurteilen, welche Datenbank für welches Anwendungsszenario geeignet ist

Übersicht der geplanten Studierendenaktivitäten²:

- Einführung und Motivation: Diese Einheit dient der Einführung in die Themen des Kurses sowie der Vorstellung der genutzten Datenbanken. Sie unterstützt Learning Outcome (1) durch die Bereitstellung grundlegender Informationen zu den Datenbanken und ihren Datenmodellen.
- Assignments PostgreSQL/Cassandra/Neo4J/MongoDB: Diese Einheiten umfassen Besprechungen, Assignments und Diskussionsrunden, in denen die Studierenden praktische Erfahrungen mit den verschiedenen Datenbanken sammeln. Sie tragen zur Erfüllung von Learning Outcome (2) bei, indem die Studierenden lernen, mit den Datenbanken zu arbeiten und komplexe Aufgaben zu lösen.
- Self-management Projekt: Dieses Projekt ermöglicht es den Studierenden, selbstständig an einem Projekt zu arbeiten und ihre Fähigkeiten im Umgang mit den Datenbanken weiter zu vertiefen. Es unterstützt Learning Outcome (3), indem die Studierenden lernen, verschiedene Datenbanken kritisch zu bewerten und die am besten geeignete für bestimmte Anwendungsszenarien auszuwählen.
- Abschlussveranstaltung: Durch die Präsentation des Projekts können die Studierenden ihre Fähigkeiten demonstrieren und ihre Beurteilungsfähigkeiten gemäß Learning Outcome (3) unter Beweis stellen.

5.3 Innovation für kompetenzorientierte, evidenzbasierte digitale Lehrformate

Das digitale NoSQLconcepts Lerntool bietet Studierenden eine einheitliche Umgebung, um mit den verschiedenen NoSQL-Datenbanken zu arbeiten und

²Mehr Details sind in unserem Learning Design zu finden unter <https://learning-design.eu/en/preview/2cacf20920dc9dfb881ab929/details>

die komplexen Konzepte zu verstehen. Das im Tool integrierte Learning Analytics Dashboard ermöglicht es den Studierenden, ihren Lernfortschritt zu verfolgen und ihre Leistung zu überwachen. Dies fördert ein bewusstes Lernen und unterstützt die Studierenden dabei, ihre Lernziele zu erreichen.

Die automatische Messung der benötigten Zeit für Aufgaben und die Angabe des empfundenen Schwierigkeitsgrads für Aufgaben bieten den Lehrkräften wertvolle Informationen über die Bedürfnisse und Lernstile der Studierenden. Auf dieser Grundlage können die Lehrkräfte den Kurs oder die Aufgaben entsprechend anpassen, um den individuellen Lernbedürfnissen gerecht zu werden und ein optimales Lernerlebnis zu gewährleisten.

Diese Funktionen des digitalen NoSQLconcepts Lerntools stellen eine Innovation für kompetenzorientierte, evidenzbasierte digitale Lernformate dar, weil sie eine personalisierte Lernerfahrung ermöglichen und den Lehrkräften wertvolle Einblicke in den Lernprozess der Studierenden bieten.

6 Anleitung zur Nutzung des NoSQLconcepts Tools

Im Folgenden wird beschrieben, wie das NoSQLconcepts Lerntool genutzt werden kann.

Der Code der Anwendung ist unter dem Github Link <https://github.com/VaneMeyer/nosqlconcepts> zu finden. Das Projekt beinhaltet folgende Ordner, die relevant sind:

- backend: hier befinden sich alle Dateien, die den Server betreffen, inklusive der Datenbankverbindungs-Einheiten
- src: hier befindet sich der Code für das Frontend (also das, was im Browser zu sehen ist)

6.1 Installation

React Projekte werden in der Regel mit Node.js und npm (Node Package Manager) erstellt und verwaltet. Daher müssen diese zuerst installiert werden³. Falls noch nicht geschehen, sollte auch Git installiert werden, um Github nutzen zu können⁴. Als nächstes öffnen Sie eine Kommandozeile oder Terminal

³Download Node.js und npm: <https://nodejs.org/en>

⁴Download Git: <https://nodejs.org/en>

und erstellen ein neues React-Projekt mit dem Befehl:

```
npx create-react-app mein-react-projekt
```

Navigieren Sie in das neue Projektverzeichnis mit dem folgenden Befehl:

```
cd mein-react-projekt
```

Klone anschließend das Repository mit dem folgenden Befehl im Terminal:

```
git clone https://github.com/VaneMeyer/nosqlconcepts.git
```

Navigieren Sie in das geklonte Verzeichnis Ihres Projektes und installieren Sie alle benötigten npm-Pakete mit dem folgenden Befehl:

```
npm install
```

Starten Sie die React App innerhalb des geklonten Verzeichnisses mit dem folgenden Befehl:

```
npm start
```

In Ihrem lokalen Browser sollte sich die App automatisch öffnen und Sie können nun mit der Entwicklung/Anpassung beginnen.

6.2 Anpassung von Komponenten

In diesem Abschnitt zeigen wir eine kurze Übersicht der Hauptkomponenten, die für die eigene Nutzung bei Bedarf angepasst werden können. Wir beginnen mit dem Frontend Code und gehen anschließend auf das Backend ein, welches unter anderem die Datenbankverbindung beinhaltet.

6.2.1 Dashboard-Karten

Die Dashboard-Komponente ist im Ordner `src/scenes/dashboard/index.jsx` zu finden. Die Dashboard-Karten werden darin durch die sogenannte *Stat-Box* Komponente definiert. Die Dashboard-Karten können nach Belieben angepasst, entfernt oder auch dupliziert werden. Zum Anpassen können neue Titel, Logos, Icons festgelegt werden. Ein neuer Link muss ebenfalls in der `src/App.js` Datei als Route hinzugefügt bzw. der bestehende überschrieben werden.

6.2.2 Interaktive Aufgabenblätter

Die Komponenten zu den Datenbank-Aufgaben sind im Ordner `src/scenes/assignments` zu finden und können nach eigenen Bedürfnissen (Titel, Untertitel) angepasst werden. Aufgabenstellungen/Datenmodelle werden innerhalb der Komponente `OptTaskForm` genutzt. Diese Komponente befindet sich im Ordner `src/components/optTaskForm.jsx`. Bilder zu den Datenmodellen befinden sich im Ordner `src/images` und können durch eigene Bilder ersetzt werden (hier müssten die Bildnamen entsprechend im `optTaskForm` Code angepasst werden). Aufgabenstellungen sind im Ordner `src/data/tasksData.js`. Auch hier können die Aufgabenstellungen durch eigene Aufgabenstellungen ersetzt werden und die Namen der Arrays angepasst werden (auch hier wieder `optTaskForm` entsprechend anpassen).

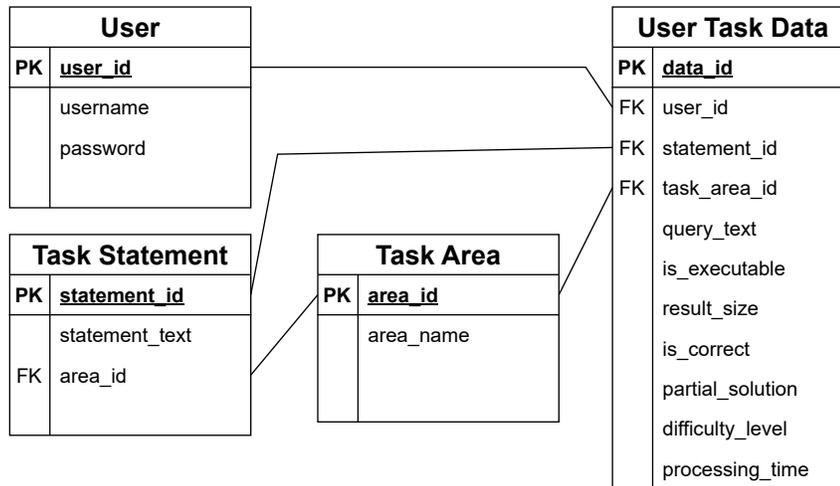
6.2.3 Backend

Die Server Datei `app.js` befindet sich im `backend` Ordner. Hier werden die Verbindungen zu den verschiedenen Datenbanken aufgebaut. Zudem sind hier die verschiedenen routes für die `get` und `post` requests definiert, die zur Kommunikation zwischen Frontend und Backend genutzt werden. Innerhalb der `.env` Datei im `backend` Ordner werden die eigenen Usernamen, Passwörter, Host, usw. für die genutzten Datenbanken eingetragen. Im Github Repository befindet sich eine Template-Datei (`.env.sample`). Diese kann mit den entsprechenden Verbindungsdaten befüllt werden und in `.env` umbenannt werden.

6.3 Datenbank Design

Für die Aufgaben wurde der öffentlich zugängliche Enron Email Datensatz⁵ verwendet. Für das Speichern von Userdaten haben wir folgende Datenbankstruktur in einer relationalen Datenbank erstellt:

⁵<http://www.cs.cmu.edu/~enron/>



Die User-Tabelle enthält Informationen über die BenutzerInnen wie Username und Passwort. Task Area enthält die IDs und Namen der verschiedenen Aufgabenbereiche (in unserem Fall die Namen der Datenbanken, also PostgreSQL, Cassandra, Neo4J und MongoDB). In der Task Statement Tabelle sind die Aufgabenstellungen gespeichert. User Task Data enthält die spezifischen Daten, die von den BenutzerInnen für jede Aufgabenstellung erfasst werden. Dazu gehören die eingegebene Query, die Ausführbarkeit der Query, die Ergebnisgröße, die Korrektheit des Ergebnisses, die Teillösung, den empfundenen Schwierigkeitsgrad und die gemessene Bearbeitungszeit.

6.4 Ausblick

Bisher setzt die Anpassung und Nutzung des Lerntools eine gewisse Vertrautheit mit der Entwicklung in React und Node, sowie Kenntnisse in der Datenbanknutzung voraus. Um die Nutzung zugänglicher zu gestalten, planen wir zukünftig die Integration eines Baukastensystems in unser Lerntool. Hierdurch wird es beispielsweise möglich sein, sich als Administrator*in online anzumelden und auf dem Dashboard individuelle Bereiche hinzuzufügen oder zu entfernen. Innerhalb der Bereiche können dann eigene Aufgaben erstellt werden. Dazu werden wir einige Komponenten modularer gestalten und allgemeinere Namen für Komponenten verwenden.

Literatur

- [1] Lena Wiese, Aboubakr Benabbas, Golnaz Elmamooz, and Daniela Nicklas. One db does not fit it all: Teaching the differences in advanced database systems. *Datenbank-Spektrum*, 21:83–89, 2021.