

eledSQL – Entwicklung und Erprobung einer webbasierten Lernumgebung für Datenbanken und SQL

Anja Ufert¹, Andreas Grillenberger², Torsten Brinda³

¹Didaktik der Informatik, Universität Erlangen-Nürnberg
Martensstraße 3, 91058 Erlangen
anja.ufert@googlemail.com

²Technische Fakultät, Universität Erlangen-Nürnberg
Martensstraße 5a, 91058 Erlangen
andreas.grillenberger@fau.de

³Didaktik der Informatik, Universität Duisburg-Essen
Schützenbahn 70, 45127 Essen
torsten.brinda@uni-due.de

Abstract: Das Thema „Datenbanken“ ist zentraler Inhalt des Informatikunterrichts in den Sekundarstufen I und II (je nach Bundesland und Lehrplansituation) in Deutschland. Für den Einstieg in diesen Themenbereich werden oft professionelle Datenbankverwaltungsprogramme basierend auf der Datenbankabfragesprache *SQL* genutzt, da kaum schülerfreundliche Alternativen zur Verfügung stehen. Bei der Nutzung dieser Werkzeuge (z.B. *MS Office Access*, *HeidiSQL*) im Unterricht treten jedoch einige Nachteile auf, die sich aus ihrem professionellen Einsatzgebiet ergeben. So wird erheblich mehr Funktionalität angeboten, als für den Schulunterricht erforderlich ist und ohne *SQL*-Vorkenntnisse sind Datenbankabfragen nicht möglich. Das führt im Unterricht dann oft zu eher theorielastigen Einführungen oder es wird versucht, Datenbanksysteme mit Hilfe von Tabellenkalkulationsprogrammen zu simulieren. Vor diesem Hintergrund wurde die Lernumgebung „Erlangen Learning Environment for Databases and SQL“ (*eledSQL*) für den schulischen Einsatz entwickelt, die bereits ohne Kenntnisse von *SQL* im Informatikunterricht eingesetzt werden kann. Dieser Beitrag beschreibt zunächst die Motivation, Konzeption und Entwicklung der Lernumgebung und vergleicht diese mit verwandten Arbeiten. Anschließend werden das Lehr-Lern-Konzept und die wesentlichen Erkenntnisse einer unterrichtlichen Erprobung in zwei Informatikkursen (Bayerisches Gymnasium, Jgst. 9) vorgestellt.

1 Motivation

Für viele Themenbereiche des Informatikunterrichts der Sekundarstufe in Deutschland gibt es bereits geeignete softwarebasierte Lernumgebungen, die die Vermittlung abstrakter informatischer Sachverhalte unterstützen (z.B. „Karol, der Roboter“, „Kara“, „Greenfoot“, „Alice“, „BlueJ“). Bislang existieren jedoch nur wenige derartige Hilfsmittel für

den Themenbereich „Datenmodellierung und Datenbanksysteme“ (vgl. [Gri12], [GB12]). Häufig werden als „Lernumgebungen“ stattdessen Programme eingesetzt, die ursprünglich für die professionelle Verwaltung von Datenbanken konzipiert wurden, bspw. *HeidiSQL*¹, *phpMyAdmin*² oder *Microsoft Office Access*. Die Nutzung solcher Programme wird bspw. in der Handreichung für die 9. Jahrgangsstufe am naturwissenschaftlich-technologischen Gymnasium in Bayern [ISB04a] empfohlen. Obwohl diese Programme alle für den Unterricht benötigten Funktionalitäten zur Verfügung stellen, treten in der Schule dennoch Probleme auf, die sich aus dem originären Einsatzgebiet solcher Datenbankverwaltungswerkzeuge ergeben: Da nur ein geringer Teil der angebotenen Funktionen im Schulunterricht genutzt wird, erscheinen diese Programme oftmals als unübersichtlich und komplex. Zudem sind Kenntnisse der Datenbankabfragesprache *SQL* Voraussetzung für die Ausführung von Datenbankabfragen, wodurch zu Beginn solcher Unterrichtssequenzen meist nicht direkt mit dem Datenbanksystem gearbeitet werden kann. Infolgedessen werden Datenbanken oft zunächst ersatzweise mit Hilfe von Tabellenkalkulationsprogrammen simuliert oder es ergeben sich eher theorielastige Einführungen in die Thematik. Die Verknüpfung mit und Nutzung von Tabellenkalkulationssystemen birgt dabei zudem die Gefahr, dass manche Lernende diese Systemkategorien nicht klar unterscheiden können.

Angesichts dieser Problematik wurde ein System für den Schulunterricht entworfen und implementiert, das primär den Einstieg von Schülerinnen und Schüler in die Datenbankabfragesprache *SQL* erleichtern soll. Hierbei sollen sogenannte natürlichsprachliche Abfragen den Schülerinnen und Schülern die Möglichkeit bieten zunächst die Notwendigkeit von Datenbanksystemen kennenzulernen, bevor schließlich schrittweise die Syntax von *SQL* erarbeitet werden kann.

Mit dem Ziel der entwicklungsbegleitenden Qualitätssicherung wurde das System in zwei Phasen unterrichtlich erprobt. Vor der systematischen Entwicklung wurde die Konzeption zunächst anhand eines Prototypen in einer Schulklasse getestet. Die guten Erfahrungen in dieser Erprobung initiierten schließlich die systematische Entwicklung der Software, die nach dem Modell der agilen Softwareentwicklung erfolgte. Darauf folgte eine systematische Erprobung des unterrichtlichen Einsatzes in zwei Schulklassen der 9. Jahrgangsstufe eines bayerischen Gymnasiums anhand eines zu diesem Zweck entwickelten Unterrichtskonzepts.

2 Verwandte Arbeiten

Zunächst erfolgte eine Literaturanalyse zu vorhandenen Lernumgebungen im Bereich „Datenbanksysteme und *SQL*“. Beschrieben werden in der Literatur überwiegend E-Learning-Systeme, die für die (Weiter-) Bildung an Hochschulen konzipiert wurden und sich grob in zwei Kategorien einordnen lassen: Systeme, die *SQL*-Anweisungen multimedial veranschaulichen bzw. Systeme, die verschiedene *SQL*-Aufgaben anbieten und die einzutragenden Lösungen bewerten [BSL⁺08].

¹www.heidisql.com (letzter Zugriff: 28.06.13)

²www.phpmyadmin.net (letzter Zugriff: 28.06.13)

Zur ersten Kategorie gehören bspw. die Desktopanwendung *eSQL* [KSF97], die schrittweise das Auswählen der abgefragten Datensätze anzeigt, anstatt lediglich das Ergebnis gegebener *SQL*-Abfragen zu präsentieren, sowie *QueryViz* [DG11], das sich besonders verschachtelten Abfragen widmet.

Zur zweiten Kategorie zählen Programme wie *SQLTutor* [Mit98] und *SQLator* [SOSL04], die alltagsbezogene Problemstellungen anbieten, welche mit geeigneten *SQL*-Abfragen gelöst werden können. Die eingegebenen *SQL*-Anweisungen werden anschließend überprüft und dem Lerner wird Feedback zu seiner Lösung gegeben.

Programme wie *SQLzoo.net* und *Teradata SQL Assistant* (beide webbasiert) vereinen Elemente beider obiger Kategorien und bieten eine Vielzahl vorgefertigter und unveränderlicher Datenbanktabellen mit zugeordneten Abfragen an, die Lernende ausführen können und detailliertes Feedback dazu erhalten [CK10]. Das am weitesten ausgearbeitete System stammt von Brusilovsky et al. [BSL⁺08, BSY⁺10] und bietet interaktive Beispiele, einen *SQL*-Test und *SQL*-Aufgaben in einer selbstentwickelten Lernplattform an.

Bei all diesen Systemen handelt es sich um reine Lernsysteme mit Erklärungsfokus und teilweise unveränderlichen Beispielen. Ihre inhaltliche Ausrichtung ist aufgrund der Zielgruppe Hochschulstudierende auch teilweise eine andere, als für den Schulunterricht benötigt. Produktives Arbeiten und Konstruieren, das im Schulunterricht der Sekundarstufe ebenfalls seinen Stellenwert hat, wird dadurch nicht unterstützt, da ein schulgeeignetes System Lernen und Produzieren kombinieren sollte.

Es existieren einzelne Ansätze, um *SQL*-Abfragen für den Schulunterricht besser nutzbar zu machen, indem eine webbasierte Oberfläche zur Durchführung von Abfragen auf einer Datenbank zur Verfügung gestellt wird. Als Beispiele können hier das *SQL Tutorial*³ und die *terra*-Datenbank⁴ genannt werden. Zu beiden Web-Oberflächen existieren umfangreiche Erklärungen (bei *SQL Tutorial* direkt in die Oberfläche integriert, bei *terra* getrennt davon) sowie umfangreiche Datensätze (bei *SQL Tutorial* zu verschiedenen Themen, bei *terra* verschiedene Daten zur Erde). Beide Lernumgebungen helfen daher mit ihrem Konzept dabei, den Unterricht zu *SQL* einfacher zu gestalten und verringern organisatorischen Aufwand.

Aufgrund der Spezialisierung beider Lernumgebungen auf das Erlernen von *SQL* besteht jedoch nicht die Möglichkeit, die Web-Oberflächen für den Einstieg in das Unterrichtsthema Datenbanken direkt einzusetzen, so dass die in der Motivation beschriebenen Probleme weiterhin bestehen. Zudem werden durch die gemeinsame Arbeit auf nur einer Tabelle verschiedene für den Unterricht relevante Anfragen verhindert. Die Aufgabe, alle Daten einer ausgewählten Tabelle zu löschen kann bspw. nicht real ausgeführt werden, da dies nur durch jeweils eine Schülerin bzw. einen Schüler erfolgen könnte. Diese Problematik ist nur durch eine Arbeit auf mehreren Tabellen mit fein abgestuften Berechtigungen zu lösen, wobei zur Überprüfung und Vergabe dieser Berechtigungen unter anderem auch eine Benutzerverwaltung nötig ist.

³<http://www.schulserver.hessen.de/darmstadt/lichtenberg/SQLTutorial/> (letzter Zugriff: 28.06.13)

⁴<http://www.sn.schule.de/terra/> (letzter Zugriff: 28.06.13)

3 Konzeption und Entwicklung

Basierend auf der Motivation und den Erkenntnissen der Literaturanalyse werden in den nachfolgenden Abschnitten Anforderungen an sowie der Entwurf und die Implementierung von *eledSQL* skizziert. Ausführlich ist dieser Prozess dokumentiert in [Gri12].

3.1 Anforderungen an *eledSQL*

Leitlinie der Konzeption von *eledSQL* war es, ein flexibles, anpassbares und für schulische Rahmenbedingungen geeignetes System zu gestalten. Es sollte webbasiert sein und den vollen Zugriff via Webbrowser sowohl von Desktop- wie auch von Mobilgeräten ermöglichen, die im schulischen Kontext zunehmend eingesetzt werden. Damit werden die Anforderungen an einen Arbeitsplatz zur Nutzung des Systems soweit wie möglich reduziert und auch die Erledigung von Hausaufgaben wird ohne weitere Installation auf privaten Geräten ermöglicht. Installationsaufwand soll damit nur noch bei der Serverinstallation (auf Schulservern oder externem Webspace im Falle von Sicherheitsbedenken) anfallen. Die Installation soll so einfach wie möglich sein. Die Analyse der Literatur ergab zudem, dass vorhandene Systeme viele nützliche Probleme und Visualisierungen bereitstellen, aber deren Anpassung an spezifische Zielgruppen nicht oder nur selten unterstützt wird. Um die bestmögliche Anpassbarkeit für den schulischen Kontext zu ermöglichen, ist daher ein differenziertes Rollenkonzept mit den Rollen Administrator, Lehrer, Kurs, Schülergruppe und Schüler mit darauf abgestimmten Benutzerrechten und rollenbasiertem Login erforderlich. Damit wird auch das parallele Arbeiten verschiedener Schülergruppen mit dem System unterstützt.

Mit *eledSQL* sollen Lernende an die unterrichtsrelevanten Inhalte zu Datenbanken und *SQL* herangeführt werden. Aus der Zielsetzung der Heranführung ergibt sich die Notwendigkeit der Anpassbarkeit des Systems für bestimmte Schüler (-gruppen). Entsprochen werden soll dieser Anforderung durch drei verschiedene Benutzermodi: im natürlich-sprachlichen Modus stehen den Schülerinnen und Schülern einfache Abfragen über Formulare und natürlichsprachliche Abfragen der Form „Wähle alle _____ aus der Tabelle aus, für die die Bedingung _____ gilt.“ zur Verfügung. In einem zweiten Modus kann den Schülerinnen und Schülern angezeigt werden, in welche *SQL*-Abfrage die jeweiligen natürlichsprachlichen Abfragen umgewandelt werden, so dass die Analogien zu *SQL* erkennbar werden, um dann schließlich in einen reinen *SQL*-Modus zu wechseln. Diese Benutzermodi sollen rollenbasiert einstellbar sein.

Benutzer der Rolle Lehrer sollen Accounts von Schülern anlegen oder diese aus einer Datei importieren, die jeweiligen Benutzerrechte bzw. den jeweiligen Benutzermodus verwalten und ferner die einzelnen Schüler zu Kursen oder Gruppen zuordnen können. Zur Anpassbarkeit an den jeweiligen Kursverlauf sollen Lehrkräfte Tabellen anlegen, bearbeiten oder aus einer Datei importieren können, um diese den Schülern anschließend entweder als Kurs-, Gruppen oder Schülertabelle zur Verfügung zu stellen. Auf Kurstabellen können bspw. alle Schüler des jeweiligen Kurses zugreifen und darauf vom Lehrer freigegebene

Operationen ausführen. Damit können die Schüler auch Erfahrungen mit der Mehrbenutzerfähigkeit von Datenbanken sammeln. Tabellen sollen dabei vom Lehrer als Vorlagen angelegt werden, die den Schülern dann als Kopie zur Verfügung gestellt werden, so dass eine Wiederherstellung bei Bedarf leicht möglich ist. Um den Fokus auf die jeweils unterrichtsrelevanten Konzepte zu legen, sollen nicht benötigte Funktionen für Tabellen, wie z.B. select, update, insert, delete etc. individuell deaktivierbar sein. Alle Lehr- und Hilfetexte im System sollen durch die Lehrperson anpassbar sein. Administratoren sollen zusätzlich Lehreraccounts verwalten und generelle Systemeinstellungen sowie gestalterische Anpassungen der Benutzungsschnittstelle an das Corporate Design einer Schule vornehmen können. Alle Texte der Benutzungsschnittstelle sollen zudem lokalisierbar sein. Benutzer der Rolle Schüler sollen alle für sie freigegebenen Funktionen auf allen ihnen zur Verfügung stehenden Tabellen ausführen können.

3.2 Design und Implementierung

Den Anforderungen der Webbasiertheit und des geringen Installationsaufwands wurde durch die Realisierung als serverbasiertes System mit browserbasiertem Zugang entsprochen. Da das System sowohl auf schulischen Servern wie auch bei externen Webhostern betrieben werden können sollte, wurden die dort jeweils zur Verfügung stehenden Rahmenbedingungen genauer untersucht. Benötigt werden jeweils kostenfreie Standardkomponenten. Webhoster bieten oft Paketlösungen mit festgelegten Programmiersprachen und Datenbanken an. Als Programmiersprache wurde aufgrund ihrer Verbreitung und Eignung für Webanwendungen *PHP* gewählt. Zur möglichst flexiblen Nutzung wurde weiterhin die Festlegung eines konkreten Datenbanksystems vermieden und zusätzlich zu den von *PHP* zur Verfügung gestellten Möglichkeiten eine Datenbankabstraktionsschicht eingeplant und implementiert, mit der *eledSQL* prinzipiell mit jeder *SQL*-fähigen Datenbank einsetzbar ist. Alle internen *SQL*-Befehle wurden so ausgelegt, dass keine Funktionen, die nur ein spezielles Datenbanksystem zur Verfügung stellt, verwendet werden.

Insgesamt wurde die Architektur von *eledSQL* gemäß dem „Model-View-Controller“-Entwurfsmuster strukturiert (s. Abb. 1). Die Darstellung der Webseite (View) wird mittels verschiedener Templates für die jeweiligen Benutzungsoberflächen generiert (siehe Abb. 2) und kann somit leicht und ohne *PHP*-Wissen an das Corporate-Design einer Schule angepasst werden. Alle in der Benutzungsschnittstelle verwendeten Texte wurden in eine Sprachdatei ausgelagert, so dass die Lokalisierung leicht möglich ist. Die Funktionsklassen (Functions) stellen alle Funktionalitäten zur Verfügung und generieren Ausgaben basierend auf erfolgten Benutzereingaben und Datenbankszugriffen, die in die Templates eingefügt werden. Die nötige Koordination zwischen den beiden Ebenen wird durch die Controller erledigt, die zusätzlich Sicherheitsprüfungen durchführen. Um diese Sicherheitsprüfungen zu erleichtern, wurde zunächst über getrennte Datenbanken für Systemdaten, Kurstabellen, Gruppentabellen und Tabellen für einzelne Schüler nachgedacht. Hier könnten die bewährten Sicherheitsmechanismen des Datenbanksystems voll ausgenutzt werden. Da viele kommerzielle Webhoster jedoch nur eine begrenzte Anzahl an Datenbanken zur Verfügung stellen und diese damit für den Betrieb von *eledSQL* ausgeschlossen

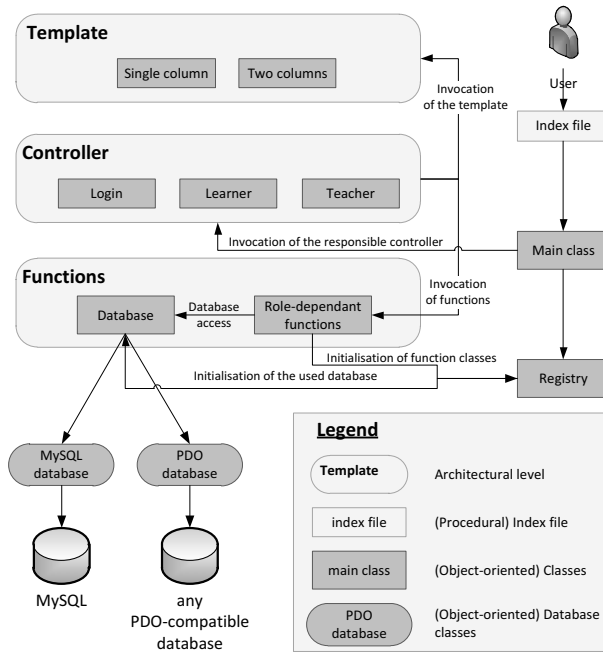


Abbildung 1: Architektur von *eledSQL*



Abbildung 2: *eledSQL* Benutzungsoberfläche (Schüleransicht)

würden, wurde eine alternative Lösung implementiert: je nach Zweck einer Tabelle enthält diese ein Präfix (bspw. für Systemtabellen „sys_“, für Kurstabellen den Namen des Kurses), so dass eine einfache Prüfung der Zugriffsberechtigungen ermöglicht wird. Auch *SQL*-Injections lassen sich damit vermeiden. Es wurde ebenfalls in Betracht gezogen, diese Präfixe vor den Lernenden zu verbergen. Jedoch wurde aus didaktischen Gründen darauf verzichtet. Das System wurde auf allen bedeutenden Webbrowsern der gängigen Betriebssysteme erfolgreich funktional getestet.

4 Unterrichtliche Erprobung

Um die Eignung und Akzeptanz von *eledSQL* im unterrichtlichen Einsatzszenario genauer zu erkunden, fand nach Abschluss der Entwicklung eine unterrichtliche Erprobung statt. Zur Erleichterung des unterrichtlichen Einsatzes von *eledSQL* für Lehrende wurde ein Unterrichtskonzept entwickelt und dokumentiert, das die Verwendung von *eledSQL* vorsieht. Nachfolgend werden zunächst die Rahmenbedingungen der erfolgten unterrichtlichen Erprobung skizziert, das darauf abgestimmte Unterrichtskonzept beschrieben und schließlich die Ergebnisse der begleitenden Untersuchungen vorgestellt. Ausführlich ist diese Erprobung beschrieben in [Ufe13].

4.1 Rahmenbedingungen

Die Erprobung fand in jeweils fünf Doppelstunden im Unterrichtsfach Informatik zweier Klassen der 9. Jahrgangsstufe eines Gymnasiums in der Metropolregion Erlangen-Nürnberg (Klasse 9a mit 17 Jungen und 15 Mädchen, Klasse 9b mit 15 Jungen und 17 Mädchen) statt. Insgesamt nahmen somit 32 Schülerinnen und 32 Schüler an der Erprobung teil. Der Unterricht fand in beiden Klassen am Vormittag in zwei aufeinanderfolgenden Stunden ab der zweiten Stunde (Klasse 9a) bzw. ab der dritten Stunde (Klasse 9b) statt und die Klassen wurden von derselben Informatiklehrkraft (weiblich) unterrichtet. Nach Angaben dieser Lehrkraft war die Klasse 9a etwas leistungsstärker als die Klasse 9b, beiden Klassen gehörten aber gute Schüler an, die den Unterricht voranbrächten. Insgesamt zeigten beide Klassen durchschnittliche bis gute Leistungen. Der bayerische Informatiklehrplan für das Gymnasium sieht ferner vor, dass in der ersten Hälfte der Jahrgangsstufe 9 funktionale Modellierung unter Verwendung von Tabellenkalkulationsprogrammen unterrichtet wird, Datenmodellierung und Datenbanken schließen sich dann in der zweiten Hälfte der Jahrgangsstufe 9 an (vgl. [ISB04b]). Die aufgrund des vorangegangenen Unterrichtsverlaufs zu erwartenden Vorkenntnisse zeigten sich auch im Unterricht bei den teilnehmenden Schülerinnen und Schülern. Ihnen wurde zu Beginn mitgeteilt, dass sie an einer Erprobung einer neuen Lernsoftware teilnehmen, worauf sie durchweg positiv und motiviert reagierten, ebenso wie die unterrichtende Lehrkraft.

4.2 Unterrichtskonzeption

Um Lehrkräften vertraute Rahmenbedingungen zu verschaffen und auch um zu einem späteren Zeitpunkt Vergleichsstudien bzgl. Kursen durchführen zu können, die unterschiedliche Systeme im Unterricht einsetzen, wurde das Unterrichtskonzept auf der Basis der Handreichung für den bayerischen Informatikunterricht der Jgst. 9 entwickelt (vgl. [ISB04a]). Zur Anknüpfung an den vorangegangenen Unterricht zu Tabellenkalkulationen werden Datenbankabfragen zunächst mit Hilfe eines Tabellenkalkulationsprogramms simuliert, dessen Grenzen bei größeren Datenmengen damit verdeutlicht (verwendet wurde bspw. eine Tabelle „Orte“ mit 25.000 Einträgen und der Frage nach der Summe der Anzahlen der Postleitzahlen dreier Orte) und damit die Notwendigkeit des Systemtypus Datenbank motiviert werden. Daran anschließend werden Datenbanksysteme und die Lernumgebung *eledSQL* als Datenbankverwaltungsoberfläche durch die Lehrkraft kurz vorgestellt. Die Schülerinnen und Schüler erhalten Arbeitsaufträge, die auch kurze Bedienungshinweise enthalten, um damit *eledSQL* und den Zugriff auf Datenbanken zunächst über die angebotenen natürlichsprachlichen Abfragen kennen zu lernen.

Im Anschluss an die Einführung in die Thematik werden in der ersten Doppelstunde zur Aufgabenbearbeitung die „Daten auflisten“-Funktion und das „Daten auswählen“-Formular zur gezielten Abfrage relevanter Datensätze aus den Datenbanktabellen eingesetzt. Unter anderem ermöglichen es insbesondere diese Funktionalitäten, *eledSQL* bereits ab Beginn der Unterrichtssequenz zu verwenden. Hauptgegenstand der zweiten Doppelstunde ist die Einführung wichtiger Datentypen, die durch das Hinzufügen von Datensätzen unpassenden Datentyps (z.B. Einfügen eines Textes in ein Feld mit Datentyp INT) über das „Daten einfügen“-Formular von *eledSQL* – eine weitere natürlichsprachliche Funktion – erörtert werden. In der dritten Doppelstunde folgt anschließend der Übergang zur Datenbankabfragesprache *SQL*. In *eledSQL* wird hierfür der Modus „Natürlichsprachliche Abfragen mit *SQL*-Anzeige“ für die Lernenden aktiviert, um die Syntax von *SQL* mithilfe der bereits bekannten natürlichsprachlichen Abfragen zu erarbeiten. Wird bspw. bei der ausgewählten Tabelle „9a_band“ die „Daten auflisten“-Funktion ausgeführt, wird die zugehörige *SQL*-Abfrage „SELECT * FROM 9a.band“ angezeigt. Durch geeignete Zuordnungsaufgaben verknüpfen die Schülerinnen und Schüler natürlichsprachliche Anweisungen mit der Syntax von *SQL*-Anweisungen. Diese Einbindung der natürlichsprachlichen Abfragen unterscheidet den Unterricht mit *eledSQL* maßgeblich vom derzeitigen Unterricht, der bspw. mit *MS Access* stattfindet. Weitere *SQL*-Befehle, wie z.B. verschiedene Aggregatfunktionen, werden anschließend in der vierten Doppelstunde eingeführt. Dadurch sind die Arbeitsaufträge nun immer weniger mit den natürlichsprachlichen Abfragen von *eledSQL* zu bearbeiten. Stattdessen wird vermehrt die Funktion „*SQL*-Abfrage ausführen“ eingesetzt, woraufhin in der fünften Doppelstunde der Modus „Nur *SQL*“ aktiviert wird. Die Schülerinnen und Schüler müssen Datenbankabfragen dann unter Verwendung aller bislang bekannten *SQL*-Schlüsselwörter – wie bspw. DISTINCT, GROUP BY oder ORDER BY – zur Lösung der Aufgaben ausschließlich mit *SQL* formulieren.

Auch wenn die Erprobung und somit das erstellte Unterrichtskonzept nur auf fünf Doppelstunden ausgelegt war, ist es leicht möglich anhand des vorliegenden Handreichungs-

konzepts den weiterführenden Unterricht mit *eledSQL* fortzusetzen.

4.3 Begleitende Untersuchungen

Um subjektive Eindrücke zum unterrichtlichen Einsatz von *eledSQL* aus Lehrerperspektive zu erhalten, wurde die Lehrkraft begleitend sowie nach Durchführung der Sequenz mündlich befragt. Die Fragen bezogen sich einerseits auf die Software an sich und andererseits auf den Unterrichtsverlauf anhand des Unterrichtskonzepts. Hierbei standen unter anderem etwaige Schwierigkeiten der Schülerinnen und Schüler und das Anforderungsniveau der Aufgaben im Vordergrund. Für die Schülerinnen und Schüler wurde in Kooperation mit dem ZiLL⁵ Nürnberg ein standardisierter Fragebogen entwickelt, der anschließend über die Open-Source-Anwendung *LimeSurvey* bereitgestellt wurde. Die Schülerinnen und Schüler wurden zu Beginn der Erprobung darum gebeten, darauf zu achten, welche Funktionen und Elemente der Benutzungsschnittstelle der Lernumgebung zu optimieren wären. Der Schülerfragebogen enthielt sechs Fragengruppen, die sich wie in Tabelle 1 gezeigt, zusammensetzten.

	Bezeichnung
Teil 1	Allgemeine Fragen zur Person
Teil 2	Fragen zu Computern, Informatik und dem Informatik-Unterricht
Teil 3	Fragen zu <i>eledSQL</i>
Teil 4	Fragen zur Aufgabenbearbeitung mit <i>eledSQL</i>
Teil 5	Fragen zu speziellen Funktionalitäten von <i>eledSQL</i>
Teil 6	<i>SQL</i> -Test

Tabelle 1: Fragengruppen des Schülerfragebogens

Der letzte Teil des Fragebogens, der „*SQL*-Test“, diente dabei als Lernzielkontrolle, ob die Schülerinnen und Schüler die für diese Unterrichtssequenz angesetzten Lernziele erreicht hatten.

Die begleitende Lehrkraft bewertete die Arbeit mit der Lernumgebung *eledSQL* im Vergleich zu den bisher verwendeten Programmen (*Microsoft Office Access* und *HeidiSQL*) insgesamt als angenehm. So sei die Unterrichtsvorbereitung mit *eledSQL* wesentlich komfortabler, da unter anderem bereits vorhandene Tabellen als Datenbanktabellen in das System importiert werden und mehrere Schülerinnen und Schüler gleichzeitig auf diesen Tabellen arbeiten konnten. Insbesondere gab sie an, dass die natürlichsprachlichen Abfragen vielen Schülerinnen und Schülern den Einstieg in *SQL* erleichtert hätten. Verbesserungsvorschläge richteten sich auf ausgewählte Funktionalitäten der Lehreroberfläche von *eledSQL*, die im Gegensatz zum restlichen Funktionsumfang weniger intuitiv erschienen. Das Unterrichtskonzept wurde insgesamt als passend bewertet. Konkrete Verbesserungsvorschläge betrafen bspw. eine Reduktion des Umfangs der bereitgestellten Arbeitsblätter sowie Formulierungs- und Formatierungswünsche bzgl. der Aufgaben. Die Aufgaben wur-

⁵Zentralinstitut für Lehr-Lernforschung, <http://www.zill.uni-erlangen.de> (letzter Zugriff: 28.06.13)

den inhaltlich als passend für die beteiligten Schülerinnen und Schüler gewertet, es sollten aber weitere schwierigere Aufgaben für besonders leistungsstarke Schülerinnen und Schüler ergänzt werden.

Die beteiligten Schülerinnen und Schüler waren überwiegend 15 Jahre alt (60%), 26% der Schüler waren 14 Jahre, der Rest war 16 Jahre alt (11%) oder machte keine Angabe. Mehr als 77% der Schülerinnen und Schüler gaben an, dass sie den Computer privat oft bis sehr oft nutzen, wohingegen nur 4% angaben, dass eine Nutzung außerhalb der Schule eher selten erfolge. Die Einschätzung der Teilnehmer bzgl. ihrer Schulleistung war normalverteilt mit 42% der Teilnehmer in der Mitte, die ihre Schulleistung als mittelmäßig einstufen. Bei ihrer Meinung zum Informatikunterricht gaben die Jungen zu 55% an, dass ihnen der Unterricht weder gut noch schlecht gefalle, die Mädchen wählten diese Option zu 66%. Gut oder sehr gut bewerteten die Jungen den Unterricht zu 26%, die Mädchen zu 19%. Bzgl. der Themen des Informatikunterrichts im Allgemeinen bekundeten 22% der Jungen Interesse, 40% Desinteresse. Nur 17% der Schülerinnen bewerteten die Inhalte des Unterrichts als interessant. Die Angaben zeigen, dass sich die Schülerinnen und Schüler (wie zuvor auch von der beteiligten Lehrkraft angegeben) ebenfalls als durchschnittlich interessiert und leistungsstark einschätzten.

Zur Frage, was ihnen an *eledSQL* am besten gefalle, wählten 32% aus vorgegebenen Optionen die einfache Handhabung, 26% das Design, weitere 26% die Funktionalität und der Rest (15%) wählte die Option Sonstiges. Bei der Bewertung des Designs waren deutliche Unterschiede zwischen Jungen und Mädchen erkennbar. Lediglich 22% der männlichen Probanden empfanden das Design als ansprechend, während 62% der Probandinnen das Design als ansprechend bewerteten, jedoch ebenfalls konkrete Verbesserungsvorschläge äußerten. Die Kritik der Schülerinnen und Schüler wurde nach Abschluss der Erprobung dazu verwendet, das Design weiterzuentwickeln.

47% der Schülerinnen und Schüler gaben an, dass die gestellten Arbeitsaufträge von ihnen einfach und ohne weitere Hilfestellungen mit *eledSQL* bearbeitet werden konnten, nur 11% aller Schülerinnen und Schüler beurteilten den Schwierigkeitsgrad der Aufgaben als „schwer“ und nur 6% gaben an, dass die Aufgabenbearbeitung mit *eledSQL* schwer sei. Dies wurde einerseits als Beleg für den erfolgreich umgesetzten intuitiven Zugang sowie andererseits auch – wie von der beteiligten Lehrkraft angesprochen – als Bedarf zur Integration weiterer, komplexerer Aufgaben in das Unterrichtskonzept interpretiert, was inzwischen auch umgesetzt wurde.

Die natürlichsprachlichen Abfragen wurden insgesamt von 34% der Schülerinnen und Schüler als hilfreich bis sehr hilfreich wahrgenommen, was die Einschätzung der Lehrkraft bestätigt. In Bezug auf die weiteren Funktionen von *eledSQL* machten die Schülerinnen und Schüler eine Reihe von konkreten Verbesserungsvorschlägen im Detail, die inzwischen umgesetzt wurden. Grundsätzliche Kritik am Konzept wurde nicht geäußert. Im abschließenden *SQL*-Test mussten die Lernenden in Auswahlfragen in Antwortalternativen vorgegebene *SQL*-Anfragen einer gegebenen natürlichsprachlichen Abfrage oder einer textuell beschriebenen Wirkung zuordnen. Dies gelang bei allen Fragen durchschnittlich 80% der Schülerinnen und Schüler, wobei die Mädchen bei allen Aufgaben mindestens gleich gut oder leicht besser abschnitten. Damit zeigt sich, dass wichtige Ziele des Unterrichts unter Verwendung von *eledSQL* als Medium erreicht wurden.

5 Schlussfolgerungen und Ausblick

Zusammenfassend lässt sich damit feststellen, dass sich *eledSQL* in diesem unterrichtlichen Einsatz sowohl aus Lehrer- wie aus Schülersicht grundsätzlich bewährt hat, auch wenn damit keine verallgemeinernde Aussage verbunden werden kann. Die Erprobung zeigte grundsätzliche Akzeptanz bei Lehrperson und Schülern. Mit *eledSQL* steht nun eine alternative Lern- und Arbeitsumgebung für den Anfangsunterricht zum Thema Datenbanken und *SQL* in der Sekundarstufe zur Verfügung, ein Bereich, zu dem es bislang nur sehr wenige unterrichtsgerechte Lernsysteme gibt. Der Zugang über natürlichsprachliche Abfragen wurde von der Lehrperson für manche Lerner als positiv bewertet, was sich auch in der Bewertung durch die Lernenden zeigt. Im Rahmen der Erprobung identifizierte Verbesserungsmöglichkeiten wurden begleitend oder nach ihrem Abschluss im System implementiert bzw. an den zugehörigen Begleitmaterialien vorgenommen.

Offen ist noch eine breitere Erprobung des unterrichtlichen Einsatzes mit mehr Probanden und mehr beteiligten Lehrkräften ebenso wie eine Vergleichsgruppenstudie mit anderen im Unterricht gebräuchlichen Datenbank-Systemen. Hierzu wäre ein weiterer Ausbau des Lehr-Lern-Konzepts um detaillierte Vorschläge für weiterführende Unterrichtsstunden ebenso wie eine Anpassung des Konzepts an die Lehrplanvorgaben anderer Schulformen sinnvoll. Funktional sollte *eledSQL* zukünftig noch um Aspekte der Datenmodellierung erweitert werden. Weiterhin wäre es anzustreben, *eledSQL* mandantenfähig zu machen, so dass nicht nur eine sondern auch mehrere Schulen mit einer Installation bedient werden können.

Literatur

- [BSL⁺08] Peter Brusilovsky, Sergey Sosnovsky, Danielle H. Lee, Michael Yudelson, Vladimir Zadorozhny und Xin Zhou. An open integrated exploratorium for database courses. In *Proceedings of the 13th annual conference on Innovation and technology in computer science education, ITiCSE '08*, Seiten 22–26, New York, NY, 2008. ACM.
- [BSY⁺10] Peter Brusilovsky, Sergey Sosnovsky, Michael V. Yudelson, Danielle H. Lee, Vladimir Zadorozhny und Xin Zhou. Learning SQL Programming with Interactive Tools: From Integration to Personalization. *Transactions on Computing Education*, 9(4):19:1–19:15, Januar 2010.
- [CK10] John Cigas und Barbara Kushan. Experiences with online SQL environments. *Journal of Computing Sciences in Colleges*, 25(5):251–257, Mai 2010.
- [DG11] Jonathan Danaparamita und Wolfgang Gatterbauer. QueryViz: helping users understand SQL queries and their patterns. In *Proceedings of the 14th International Conference on Extending Database Technology, EDBT/ICDT '11*, Seiten 558–561, New York, NY, 2011. ACM.
- [GB12] Andreas Grillenberger und Torsten Brinda. *eledSQL*: a new web-based learning environment for teaching databases and SQL at secondary school level. In *Proceedings of the 7th Workshop in Primary and Secondary Computing Education, WiPSCE '12*, Seiten 101–104, New York, NY, USA, 2012. ACM.

- [Gri12] Andreas Grillenberger. Entwurf und Implementierung einer Lernumgebung für den Informatikunterricht zur Unterrichtssequenz Datenbanksysteme. Zulassungsarbeit zum ersten Staatsexamen, Friedrich-Alexander-Universität Erlangen-Nürnberg, Februar 2012.
- [ISB04a] Staatsinstitut für Schulqualität und Bildungsforschung ISB. Informatik am Naturwissenschaftlich-technologischen Gymnasium, Jahrgangsstufe 9 (Handreichungsentwurf). <http://www.isb.bayern.de/gymnasium/materialien/informatik-naturwissenschaftlich-jgst-9/>, 2004. [letzter Aufruf: 21.06.2013].
- [ISB04b] Staatsinstitut für Schulqualität und Bildungsforschung ISB. Lehrplan Informatik, Jahrgangsstufe 9. <http://www.isb-gym8-lehrplan.de/contentserv/3.1.neu/g8.de/index.php?StoryID=26434>, 2004. [letzter Aufruf: 21.06.2013].
- [KSF97] R. Kearns, Stephen Shead und Alan Fekete. A teaching system for SQL. In *Proceedings of the 2nd Australasian conference on Computer science education*, ACSE '97, Seiten 224–231, New York, NY, 1997. ACM.
- [Mit98] Antonija Mitrovic. Learning SQL with a computerized tutor. In *Proceedings of the twenty-ninth SIGCSE technical symposium on Computer science education*, SIGCSE '98, Seiten 307–311, New York, NY, 1998. ACM.
- [SOSL04] Shazia Sadiq, Maria Orlowska, Wasim Sadiq und Joe Lin. SQLator: an online SQL learning workbench. In *Proceedings of the 9th annual ACM SIGCSE conference on Innovation and technology in computer science education*, ITiCSE '04, Seiten 223–227, New York, NY, 2004. ACM.
- [Ufe13] Anja Ufert. Erprobung und Weiterentwicklung der Lernumgebung eledSQL im Hinblick auf die Verwendung im Unterricht. Zulassungsarbeit zum ersten Staatsexamen, Friedrich-Alexander-Universität Erlangen-Nürnberg, Januar 2013.