

Mobile Learning

Einsatz mobiler Endgeräte im Lernen, Wissenserwerb
sowie der Lehr-/Lernorganisation



Mostafa Akbari, Doreen Böhnstedt, Claudia Bremer,
Mohamed Amine Chatti, Christoph Rensing

© 2012 Mostafa Akbari, Doreen Böhnstedt, Claudia Bremer, Mohamed Amine Chatti, Christoph Rensing

Editor:	Dean of the Department of Mathematics and Computer Science
Type and Print:	FernUniversität in Hagen
Distribution:	http://deposit.fernuni-hagen.de/view/departments/miresearchreports.html

Mostafa Akbari, Doreen Böhnstedt, Claudia Bremer,
Mohamed Amine Chatti, Christoph Rensing

Mobile Learning

Einsatz mobiler Endgeräte im Lernen, Wissenserwerb sowie
der Lehr-/Lernorganisation

2. Workshop im Rahmen der DeLFI 2012

24. September 2012 in Hagen

Vorwort

Mobile Learning wird schon seit mehreren Jahren eine hohe und weiterhin zunehmende Bedeutung zugesagt. Das spiegelt sich in verschiedenen Studien wie beispielsweise dem Horizon Report wieder. Die zunehmende Leistungsfähigkeit, Ausstattung und die Bedienbarkeit mobiler Endgeräte sowie die immer häufigere und kostengünstige Verbindung der Geräte mit dem Internet verstärkt diese Annahme. In der Praxis werden mobile Endgeräte inzwischen sehr häufig auch zu Lernzwecken verwendet. In vielen Anwendungsfällen steht dabei orts- und zeitunabhängiges Lernen im Vordergrund. Ein zentrales Ergebnis der Diskussion des letztjährigen ersten Workshops zum Thema Mobile Learning im Rahmen der DeLFI war, dass vielfältige Potenziale des Einsatzes mobiler Technologien, die insbesondere im kontextabhängigen oder kontextübergreifenden Lernen gesehen werden, noch nicht ausgenutzt werden. Es besteht also weiterhin Forschungs- und Entwicklungsbedarf und ein starkes Interesse der Wissenschaft an der Beantwortung von Forschungsfragen rund um Mobile Learning.

Dieses wachsende Interesse zeigt sich auch an den Einreichungen für diesen Workshop. Nach sorgfältiger Begutachtung durch das Programmkomitee wurden sechs Beiträge für die Präsentation im Rahmen des Workshops und Aufnahme in die hier vorliegenden Proceedings ausgewählt. Drei weitere Beiträge präsentieren sich in Form von Videos auf dem YouTube Kanal der GI Fachgruppe E-Learning (<http://www.youtube.com/user/fgelearning>) und ergänzend mittels einer zweiseitigen Zusammenfassung in diesen Proceedings.

Beispiele für den Einsatz von mobilen Endgeräten stellen die Beiträge von Schmitz sowie Sonne und Harriehausen-Mühlbauer (Video) vor. Ersterer beschreibt die Nutzung verschiedener mobiler Anwendungen während eines Museumsbesuchs, während beim zweiten Beitrag die Gestaltung einer Mobilen Lernanwendung im Fokus steht. Einen Ausblick in die Zukunft mobilen Lernens an der Hochschule wagen Kapp et. al. in ihrem Video.

Brauner et. al. beschreiben eine Anwendung zum Erlernen von Programmierkonzepten, deren Vorzüge insbesondere darin bestehen, dass die Möglichkeiten der von einem Tablet bereitgestellten Interfaces (Touchpad und Sensorik) verwendet werden.

Aber nicht nur zur Wissens- und Kompetenzvermittlung sondern auch zum Testen und Prüfen des erworbenen Wissens können mobile Anwendungen genutzt werden. Dies steht im Fokus zweier weiterer Beiträge. Dalir und Rölke beschreiben eine Systemarchitektur zur mobilen Durchführung von Assessments und deren Nutzung in einer Fallstudie. Im Beitrag von Helmerich et. al. wird ein mobiles System zum Training mittels Fragesammlungen vorgestellt und mit einem rechnergestützten Training verglichen.

Fragestellungen der Integration mobiler Endgeräte und Anwendungen in bestehende E-Learning-Infrastrukturen werden in den weiteren Beiträgen des Workshops adressiert. Bernoth et. al. beschreiben mit einer Anwendung zum Einholen von Lernerfeedback während einer Präsenzveranstaltung ein zunehmend an den Hochschulen realisiertes Szenario. Der Schwerpunkt des Beitrags liegt auf der offenen Architektur und die Möglichkeit der Integration in verschiedene Anwendungen. Die Notwendigkeit bestehender heterogener Anwendungen, die nicht über Schnittstellen für mobile Anwendungen verfügen, in ein mobiles Nutzungsszenario zu integrieren besteht an vielen Hochschulen. Von den eigenen Erfahrungen und über eine Architektur zur Einbindung solcher Anwendungen berichten Bernoth et. al. Eine ähnliche Zielsetzung verfolgt der Beitrag von Kneifel und Lucke. Sie stellen in ihrem Video ein Nachrichtensystem vor, das in großen heterogenen Umgebungen Daten an mobile Endgeräte verteilen kann und damit eine Grundlage für neuartige Lernanwendungen darstellt.

Wir danken allen Autoren für ihre Beiträge zum Workshop, dem Programmkomitee und den Gutachtern für die sorgfältige Begutachtung, die konstruktiven Anmerkungen und die Unterstützung bei der Auswahl von interessanten Beiträgen und den Chairs der Gesamtkonferenz DeLFI für das in uns gesetzte Vertrauen.

Aachen, Frankfurt, Darmstadt im September 2012

Organisationskomitee und Workshopleitung

Christoph Rensing (Technische Universität Darmstadt)
Mostafa Akbari (RWTH Aachen University)
Doreen Böhnstedt (Technische Universität Darmstadt)
Claudia Bremer (Goethe Universität Frankfurt)
Mohamed Amine Chatti (RWTH Aachen University)

Programmkomitee

Mostafa Akbari (RWTH Aachen University)
Mohamed Amine Chatti (RWTH Aachen University)
Doreen Böhnstedt (Technische Universität Darmstadt)
Claudia Bremer (Goethe Universität Frankfurt)
Anna Hoberg (FHG IAO Stuttgart)
Christoph Igel (DFKI CeLTech – Saarbrücken)
Marco Kalz (CELSTEC)
Ulrike Lucke (Universität Potsdam)
Nadine Ojstersek (Universität Duisburg-Essen)
Christoph Rensing (Technische Universität Darmstadt)
Marcus Specht(CELSTEC)
Ralf Steinmetz (httc & TU Darmstadt)
Hendrik Thüs (RWTH Aachen)
Claudia de Witt (Fernuniversität Hagen)

Ergänzende Gutachter/innen

Alexander Kiy (Universität Potsdam)
Roland Klemke (CELSTEC)
Birgit Schmitz (CELSTEC)
Raphael Zender (Universität Potsdam)

Inhaltsverzeichnis

Mobile Lernunterstützung beim außeruniversitären Lernen im Museum. Ein Erfahrungsbericht aus Lernerperspektive	7
Daniela Schmitz	
ScratchTab – Eine Tablet-basierte Anwendung zum Erlernen von Programmierkonzepten	15
Philipp Brauner, Hendrik Thüs, Martina Ziefle, Ulrik Schroeder	
Hardware und Software für mobiles Assessment im Schulbereich.....	23
Mahtab Dalir, Heiko Rölke	
Nutzung elektronischer Prüfungsfragesammlungen: Evaluation und Vergleich eines mobilen mit einem rechnergestützten Trainingssystem	31
Jürgen Helmerich, Alexander Hörnlein, Marianus Ifland, Frank Puppe	
Click2Vote - Ein servicebasiertes Clicker-Framework zur Teilnehmeraktivierung in Lehrveranstaltungen.....	39
Tom Tschernak, Raphael Zender, Ulrike Lucke	
Studi.UP: Mobile Mehrwert-Dienste für konventionelle Legacy-Applikationen des Campus Management.....	47
Jan Bernoth, Alexander Kortsch, Volker Kaatz, Sander Lass, Raphael Zender, Ulrike Lucke	
Lufthansa Privacy Quiz.....	55
Udo Sonne, Bettina Harriehausen-Mühlbauer	
Die Zukunft des E-Learning	57
Martin Kapp, Lieven Leue, Tina Beigel, Ulrike Lucke.	
Angry Students: Mobile Multiplayer Action Game	59
Marc O. Kneifel, Ulrike Lucke.	

Mobile Lernunterstützung beim außeruniversitären Lernen im Museum. Ein Erfahrungsbericht aus Lernerperspektive

Daniela Schmitz

Technische Universität Dortmund
Lehrstuhl Technik und ihre Didaktik
Otto-Hahn-Straße 6
44227 Dortmund
daniela.schmitz@tu-dortmund.de

Abstract: Wie eine mobile Lernunterstützung eines Museumsbesuchs aussehen kann, zeigt der folgende Beitrag auf. Dazu wurde ein didaktisches Konzept mit mobiler Lernunterstützung und dem Lernen in Kleingruppen entwickelt und anhand des vorgestellten Museumsbesuches umgesetzt. Im Anschluss an die Darstellung des Konzepts und der Erfahrung folgt eine Beschreibung der Ergebnisse der teilnehmenden Beobachtung.

1 Einleitung

Mobile Learning, so vermuten Experten, wird für das Lernen der Zukunft und damit auch als didaktische Form bedeutender [Im11]. Unter mobilem Lernen wird hier das Lernen mit Hilfe mobiler Endgeräte im physischen Kontext verstanden. Das heißt, der Kontext, in dem das Lernen stattfindet, ist für das Lernen relevant [FS09]. Neben technischen Aspekten und der Weiterentwicklung von Apps, liegen gerade im didaktischen Bereich Potenziale, die es auszuschöpfen gilt. So ist es auch Aufgabe der Hochschullehrenden, sich mit dieser Technologie auseinanderzusetzen und didaktische Potenziale auszuloten. Im Rahmen des Lehramtsstudiums für technische Fächer müssen die Studierenden an Exkursionen teilnehmen. Exkursionen als außeruniversitäre Lehrveranstaltungen bieten ideale Ansatzpunkte für eine mobile Lernunterstützung. Im folgenden Beitrag wird dies an einer Exkursion zum LWL Industriemuseum Henrichshütte Hattingen vorgestellt. Hier wird ein Ansatz aufgezeigt, wie mit Hilfe mobiler Endgeräte eine neue Möglichkeit des gemeinsamen Lernens an Exponaten und des Wissenserwerbs im Museum gestaltet sein kann. Diese Exkursion beinhaltete den Besuch der Sonderausstellung „Leonardo da Vinci – Bewegende Erfindungen“ und der Dauerausstellung auf dem Außengelände. Zu beiden Ausstellungen gab es vom Lehrenden erstellte begleitende Aufgabenstellungen und Unterlagen auf dem iPad.

Die leitende Fragestellung für die Konzipierung des Lernarrangements ist, wie ein realistisches, für die Schulpraxis realisierbares, mobiles Lernszenario gestaltet werden

kann. Dies unterteilt sich in die Teilfragen: Wie kann ein schlanker Ansatz realisiert werden, das Lernen im Museum anzureichern? Wie können Studierende die Lernform der mobilen Lernunterstützung erleben und dazu den Fokus als zukünftig Lehrende einnehmen? Wie bewerten die Studierenden dieses Lernarrangement? Ein schlanker Ansatz beinhaltet, dass keine großen Kosten anfallen sollen, kein Einkauf externer Ressourcen stattfindet und auch mit durchschnittlicher Medienkompetenz von Lehrenden ein Lernsetting möglich ist. Der mediale Habitus ist bei LehrerInnen meist nicht so stark mediengeprägt, deswegen „bedürfte es bei den zukünftigen LehrerInnen weitreichender Kompetenzen, um möglicherweise über den (theoretisch fundierten) didaktischen Einsatz jugendkulturell relevanter Medien (zur Zeit z.B. das Handy) [...] neue Chancen zu eröffnen“ [Ko10]. Medienkompetenz und Medienbildung sind unabdingbar, „um Medien im Unterricht adäquat einsetzen zu können – und auch um sie zum Gegenstand des Unterrichts zu machen“ [ebd.]. Dazu gehört auch, passende Konzepte zum Einsatz neuer Medien zu kennen [Bi09].

Daher rückt in diesem Beitrag die technische Perspektive in den Hintergrund und es wird ein schlanker Ansatz für didaktische Innovationen verfolgt. Da viele LehrerInnen ausbaufähige Medienkompetenzen haben [Ko10], bieten sich eben nicht unbedingt komplexe mobile Autorensysteme an. Das Related Work beschreibt sehr innovative und vor allem interessante Ansätze für das mobil unterstützte Lernen im Museum [Va09 und OS99] oder Location based Learning [RTA11], die natürlich für die Schulpraxis bedeutend sind. Wenn in einem Projekt Content des Museums digitalisiert wird [Va09] oder direkt vom Museum zur Verfügung gestellt würde, ist dies ideal. Dies deckt sich allerdings nicht flächendeckend mit dem Museumsalltag. Die Autoren beschreiben ein umfangreiches Lernsystem, aber in der Evaluation selbst werden die Kosten der Geräte und der Digitalisierung durch die MitarbeiterInnen angesprochen sowie der Support und die Pflege eines solchen Systems, die geleistet werden müssen [Va09]. Nicht immer sind durch Projekte die Kostenübernahme und Begleitung gegeben. Daher muss es auch schlanke Lösungen geben, die mit einfachen technischen Möglichkeiten eine Lernumgebung gestalten. Es verfügt nicht jede angehende Lehrkraft über ein Smartphone, nicht jeder kann eine App programmieren oder arbeitet in einem Forschungsprojekt in dem eine App für die eigenen Bedürfnisse generiert wird. Dieser Beitrag zeigt auf, wie aus didaktischer Perspektive angehende LehrerInnen diese Lernform aus Nutzerperspektive und Lehrendenperspektive erleben und Ideen für die Umsetzung in der schulische Praxis erhalten. Der nächste Abschnitt stellt das didaktische Konzept für die Vorbereitung, Durchführung und Nachbereitung des Lernszenarios vor. Danach folgt eine Beschreibung der Ergebnisse der Evaluation sowie von Ansätzen für den Transfer

2 Das didaktische Konzept des Lernszenarios

2.1 Entwicklung des Konzepts und Vorbereitung der Exkursion

Die Entwicklung und Durchführung neuer Lernszenarien mit mobiler Technologie hat nicht nur für die Studierenden sondern auch aus Dozentensicht motivierenden Charakter [We11], besonders wenn durch zunehmende Lehrerfahrungen die organisatorische und

technische Vorbereitung abnimmt und so der Hauptbeschäftigungsfokus der Didaktik gelten kann. Aus exkursionsdidaktischer Sicht gilt es, die Vorbereitung, Durchführung und Nachbereitung von Exkursionen in didaktischer, organisatorischer und rechtlicher Perspektive zu gestalten [Sc11] und für das didaktische Konzept für das Lernen im Museum zu berücksichtigen. Ausgangspunkt eines didaktischen Konzepts ist, die Lehr-/Lernziele festzulegen. Das Lehrziel war das Kennen lernen und Einsetzen dieser Form des mobilen Lernens. Die Lernziele der Exkursion waren, die Erfindungen da Vincis nachzuvollziehen und mit Hilfe des mobilen Endgerätes in der Gruppe Inhalte zu erschließen, Aufgaben zu lösen und eigene Aufgaben zu generieren. Dadurch, dass nicht alle elf teilnehmenden Studierenden ein eigenes Gerät hatten, waren gleichzeitig neue Interaktionsformen als Duo oder Trio gefragt. Von den Lernzielen ausgehend werden die Lerninhalte, Lernmethoden und Medien festgelegt. Für diese durchgeführte Exkursion standen die Erfindungen da Vincis sowie der Verhüttungsprozess mit Hilfe vordefinierter Aufgaben und mobiler Lernunterstützung im Vordergrund. Informationen zum Ablauf der Exkursion sowie den mobilen Zugriff auf die benötigten Lerninhalte vor Ort bereitzustellen, gehörte zur organisatorischen Vorbereitung. Außerdem wurde das konkrete Lernsetting generiert und der benötigte Content für das iPad bereitgestellt. Die Auswahl der Apps erfolgte entsprechend der leitenden Fragestellung und des generierten didaktischen Konzepts. In einer Testphase wurden kostenfreie Apps auf ihre Funktionen und Usability getestet. Die Auswahl fiel auf die folgenden Apps: iBooks, Documents Free, Video, Multiple Choice Trainer und Notes. Die nachfolgende Abbildung zeigt die Bausteine des didaktischen Konzepts auf:

Phase	Dauer	Ziel	Methode	Medien
Vorbereitung	90 min.	Umgang mit dem iPad, Erkundung Umgebung Geschichte der Henrichshütte Nachbau einer Erfindung	Formulieren von Lernzielen, Generieren eines Lernquiz	Podcast und Apps, Holzleisten
Durchführung	120 min.	Erfindungen da Vincis in ihrer Bedeutung und Funktion nachvollziehen	Erkunden, technische Skizze, Lernquiz, Zwischenpräsentation	Apps: iBooks, Documents Free, Video, Multiple Choice Trainer, Notes
	120 min.	Geschichte von Eisen und Stahl nachvollziehen	Erkunden, Lernquiz	
Nachbereitung	60 min.	Lernquiz, Reflexion der Exkursion, Evaluation der mobilen Lernunterstützung	Reflexion des Lernweges, Blitzlichtrunde Befragung	Apps: Documents Free, Multiple Choice Trainer

Abbildung 1: Konzept der Exkursion

Ein „normaler“ Museumsbesuch in der Henrichshütte erfolgt ohne mobile Unterstützung. Hier kann an den jeweiligen Exponaten, die Infotafel als Informationsquelle

genutzt oder die MuseumsmitarbeiterInnen gefragt werden. Innerhalb der Exkursionsdidaktik ist das Lernen mit mobilen Endgeräten zur Lernunterstützung ein noch vernachlässigtes Thema. Hier bestehen meist papierbasierte Ansätze zu exkursions-typischen Handlungen wie beschreiben, messen, sammeln, befragen etc. [K107] Daneben steht als reine online Variante die virtuelle Exkursion, die komplett als Internettour stattfindet. Der Vorteil des Einsatzes mobiler Technologie ist, dass mit Hilfe eines Gerätes ein mobiler Lernbegleiter für den Tag bereit steht, der in der Rolle der Lernenden Informationen und Videos bei Bedarf bereitstellt sowie in der Rolle der Lehrenden die Dokumentation und Generierung von Lernfragen ermöglicht.

2.2 Durchführung der Exkursion

Die *Vorbesprechung* zur Exkursion fand vor Ort statt und begann mit der Frage, welche Vorerfahrungen die Studierenden mit mobilen Endgeräten und dem Lernen mit diesen haben. Die Studierenden haben alle schon mal mit einem iPod gespielt oder Musik gehört. Einige haben schon damit im Internet gesurft und Mails abgerufen. Dass mit diesen Geräten das Lernen unterstützt werden kann, wurde nicht genannt. Der erste Auftrag der Vorbesprechung war die Umgebung des Museums zu erkunden. Ein ehemaliger Bunker der Henrichshütte, der heute als Serverfarm dient, liegt etwas abseits vom Museum und wird daher nicht sofort dem ehemaligen Unternehmen als zugehörig empfunden. Auf dem Weg dorthin wurde spekuliert, wo der Zusammenhang zum Museum besteht und der Hinweis gegeben, dass mit Hilfe der mobilen Endgeräte die Lösung gefunden werden kann. Daher bot sich die Erkundung im Rahmen der Vorbesprechung und dem gleichzeitigen Kennenlernen sowie Arbeiten mit dem iPad gut an. War die Lösung gefunden, sollte als nächstes mit Hilfe eines Podcasts erfahren werden, was unter mobilem Lernen zu verstehen ist. Wegen des Wetters folgte die Erschließung der Geschichte der Henrichshütte in einer nahe gelegenen Cafeteria. Mit Hilfe vorbereiteter Unterlagen in Form kurzer Lerneinheiten als Pdfs in iBooks konnte die Geschichte in den Lernteams erarbeitet werden. Aus den bisherigen Inhalten der Vorbesprechung sollte abschließend Lernfragen entwickelt und in der App Multiple Choice Trainer eingegeben werden. Ein Ziel des Tages war, zu jeder Station Fragen aus Lehrendenperspektive zu generieren. Weiterhin sollten die Studierenden in der App Documents Free ihre persönlichen Lernziele für den Exkursionstag notieren, um an den einzelnen Stationen des Tages zu prüfen, wie weit sie diese erreicht haben. Danach folgte eine praktische Aufgabe (s. Abb.2).



Abbildung 2: Leonardo-Brücke aus Holzleisten

Als Vorbereitung auf die Sonderausstellung wurde mit Hilfe von Holzleisten die so genannte Leonardo-Brücke nachgebaut (Abb. 2), welche durch einfaches übereinanderlegen der Leisten eine hohe Stabilität erreicht und das simple Prinzip dahinter veranschaulicht.

Die *Durchführung* der Exkursion begann mit der Sonderausstellung „Leonardo da Vinci – bewegende Erfindungen“. Der Schwerpunkt der interaktive Ausstellung (vgl. <http://www.leonardo-bewegende-erfindungen.de/>) liegt auf dem Ausprobieren der Erfindungen Leonardo da Vincis. Hierzu gab es einen Videoclip auf der Homepage des Museums, welchen die Studierenden während des Wartens am Eingang anschauten, um die Wartezeit zu verkürzen. Der Besuch der Sonderausstellung verlief selbstorganisiert in zweier oder dreier Teams. Die Studierenden erhielten den Arbeitsauftrag, die einzelnen Exponate nachzuvollziehen. Die Exponate waren so angelegt, dass jedes ausprobiert werden konnte. Dazu hatte jedes eine kleine Infotafel seitens des Museums. Zusätzlich hatten die Studierenden auf den iPads das Begleitdokument über diese Ausstellung, welches es vor Ort nicht gab. So konnten sie bei Bedarf zu jedem einzelnen Exponat, vor dem sie sich befanden, zusätzliche Informationen heranziehen, so dass keine inhaltlichen Fragen mehr offen blieben. Aus den Exponaten sollten sie schließlich eines auswählen, zu dem folgende Leitfragen zu bearbeiten waren: Welche Funktion erfüllt das Modell? In welchen Ingenieurbereichen wird es verwendet? Welche Maschinenelemente enthält das Modell? Welche technischen Anwendungen in der heutigen Zeit enthalten Elemente dieses Modells? Zuletzt sollte eine grobe Skizze des technischen Funktionsprinzips in der App Notes skizziert werden. Nachdem diese Aufgaben bearbeitet waren, erfolgte eine Zwischenpräsentation der Ergebnisse zu den Exponaten. Hier versammelte sich die Gruppe, präsentierte ihre Skizzen und Ergebnisse, formulierte weitere Lernfragen und überprüfte den Stand ihres Lernweges. Nach einer kleinen Pause, fand die Gruppe wieder zusammen und schaute in Vorbereitung auf die Dauerausstellung den Videoclip „Vom Erz zum Stahl“ als Input an, welcher den Hochofenprozess veranschaulicht. Danach gingen die Studierenden selbstorganisiert und interessengeleitet durch die einzelnen Bereiche auf dem Außengelände, wie z.B. die Anlieferung des Erzes, Erztaschen oder Besteigen des Hochofens. Zudem wurden an selbst ausgewählten Exponaten weitere Lernfragen generiert. Da der Außenbereich sehr groß war und viele Informationen bereithielt, war die Motivation sehr hoch. Dies zeigte sich darin, „besonders schwere“ Fragen für das abschließende Lernquiz zu finden.

In der *Nachbesprechung* wurden die formulierten Lernziele aus der Vorbesprechung thematisiert. Die Lernziele sowie Stichpunkte zum Inhalt, zur Arbeit in der Gruppe und zum methodischen Vorgehen, hatten die Studierenden in der App Documents Free dokumentiert. Die Studierenden stellten ihre Lernziele vor und reflektierten anschließend ihre Lernwege. Als Ausblick für zukünftiges Lernen thematisierten sie, was sie beim nächsten Mal anders machen würden. Als inhaltliche Nachbereitung dienten die von den Studierenden erstellten Lernfragen in der App Multiple Choice Trainer. Jede Gruppe hatte insgesamt mindestens 8 bis maximal 15 Fragen zu den einzelnen Schwerpunkten des Tages generiert. Nun wurden die iPads reihum im Uhrzeigersinn von Gruppe zu Gruppe weitergereicht und die Fragen beantwortet. Das Bearbeiten der Lernquizze war ein guter inhaltlicher Abschluss, der den Studierenden viel Spaß bereitete. Zum einen erhielten sie so Einblick, was die anderen Gruppen inhaltlich für wichtig oder interessant

befanden, zum anderen versteckten sich einzelne Spaßfragen darin, die zum Schmunzeln brachten. Danach folgte die Auswertung des Mobile Learning Einsatzes sowie das Ausfüllen des standardisierten Fragebogens.

3 Evaluation des mobile Learning Einsatzes durch die Studierenden

Das oben vorgestellte didaktische Konzept und dessen Umsetzung am Beispiel der Henrichshütte wurden anhand standardisierter Fragen zum Lernszenario des Exkursionstages bewertet. Zusätzlich konnten die erhobenen Daten durch die teilnehmende Beobachtung der Gruppen und der Arbeit mit dem iPad ergänzt werden. Die einzelnen Items wurden mit einer 5er Skala von „1 = Stimme zu ... 5 Stimme nicht zu“ abgefragt. Von den elf teilnehmenden Studierenden wurden alle elf Evaluationsbögen mit Hilfe von EvaSys erfasst. Die Studierenden bewerteten das Mobile Learning auf der Exkursion global mit einem Mittelwert von $m = 1,42$ bei einer Standardabweichung von $s = 0,48$. Als Grund für die Teilnahme an der Exkursion kreuzten 90,9% „aus Interesse“ und 63,6% „als Pflichtveranstaltung“ an. Alle abgefragten Items wurden anhand der obigen 5er Skala beantwortet. Das Item „Die Exkursion trägt zu meinem Interesse an Mobile Learning“ erzielte einen Mittelwert von 1,7. Das Lernarrangement mit mobilem Endgeräte wurde mit einem Mittelwert von 1,18 als angemessen bewertet. Insgesamt sind 36,4% der befragten Studierenden mit dieser Exkursion voll zufrieden und 63,6% eher zufrieden. Dass die Apps hilfreich waren, bestätigten 90,9%. Zudem gab es einen gesonderten Block zu Mobile Learning, welcher nachfolgend tabellarisch aufgezeigt wird. Auch hier gilt die 5er Skala. Dabei werden für jedes Item der Mittelwert (m), die Standardabweichung (s) und der Median (md) angegeben.

Item	m	s	md
Ich kam mit dieser Lernform gut zurecht.	1,27	0,47	1
Der Einsatzes des mobilen Endgerätes unterstützte meinen Lernprozess gut	1,27	0,47	1
Der Einsatz des mobilen Endgerätes unterstützte unseren Lernprozess in der Gruppe gut	1,27	0,47	1
Ich bin mit den Inhalten (Videos u.ä.) auf dem iPad zufrieden.	1,45	0,52	1
Mit der App Multiple Choice Trainer bin ich zufrieden.	1,36	0,5	1
Mit der App iBooks bin ich zufrieden.	1,33	0,5	1
Mit der App Documents Free bin ich zufrieden.	1,78	0,83	2
Insgesamt bin ich mit dieser Lernform zufrieden.	1,45	0,52	1

Abbildung 3: Bewertung des Mobile Learning

Bei den offenen Fragen zum mobilen Lernen sollten die Studierenden insbesondere darauf eingehen, was besonders gut gelaufen ist und was weniger. Im Sinne des Lehrziels, mobiles Lernen kennenzulernen und einzusetzen, war es auch im Sinne der Studierenden eine kurze Schulung zum iPad zu erhalten. Besonders die Videoclips wurden als sehr anschaulich und informativ bewertet, da diese das jeweils benötigte Thema auch in kurzer Form demonstrierten. Hinsichtlich des mobilen Lernens im Rahmen der Vorbesprechung gaben die Studierenden an, dass es gut zur Exkursion passe, dass die Informationen direkt vor Ort abgerufen und dass es überall genutzt werden konnte. Auch die Vielfältigkeit der Medien auf dem iPad wurde betont sowie die Erkenntnis von Einzelnen, dass mit Hilfe eines solchen Geräts, das Lernen und Üben an Attraktivität gewinnen kann. Ergänzend dazu lassen sich Aussagen zum Medieneinsatz

finden, dass es viele mediale Möglichkeiten gibt, die didaktisch sinnvoll eingesetzt werden können. Zur Durchführung der Exkursionen wurden ebenfalls aus Lernendensicht die multimedialen Inhalte hervorgehoben. Ebenso positiv empfanden sie die selbstorganisierte Herangehensweise, die durch Strukturen und Rahmenbedingungen mit dem iPad gelenkt wurde. Die freie Zeiteinteilung, das selbstständige Arbeiten und damit auch die Arbeit mit dem iPad wurden positiv hervorgehoben. Einstimmig als lehrreich, interessant und als gelungener Abschluss wurde das abschließende Lernquiz in der App Multiple Choice Trainer bewertet. Die Studierenden zogen abschließend folgende Erkenntnisse aus der Exkursion und dem Einsatz von mobilem Lernen: Zunächst wurde hier die Einsatzvielfalt, die Sinnhaftigkeit und Multimedialität des mobilen Einsatzes genannt. Das Lernarrangement wird besonders gut bewertet aufgrund der freien Zeiteinteilung, der Arbeit mit iPads und der Selbstständigkeit. Dies führte zudem auch zu mehr Interesse an dem Thema und brachte als neue Form des Lernens Abwechslung. Dies lässt sich im Sinne des Lernziels interpretieren, dass sie diese Lernform kennen lernten, einsetzten und aus Perspektive von Lehrenden betrachteten, indem sie Transferansätze für den Unterricht reflektierten. Insgesamt wurde mobile Learning als gute Abwechslung und auch als spielerisches Lernen gesehen.

Aus der teilnehmenden Beobachtung lassen sich die Aussagen der Studierenden bestätigen. Die Studierenden haben sich sehr intensiv mit den Inhalten auseinandergesetzt und darüber hinaus das iPad an sich ausprobiert. In der Vorbesprechung hatten die iPads und die Podcasts einen besonders motivierenden Charakter. Bei der Exkursionsdurchführung wurden Informationen am Standort der Exponate abgerufen und über Fragen für das Lernquiz diskutiert. Die Arbeit in einer Zweiergruppe mit einem iPad erwies sich als ideal. In vergangenen Exkursionen mit größeren Teilnehmerzahlen konnten nie alle gleiche aktiv mit dem iPad arbeiten, in Zweiergruppen gelang dies sehr gut. Aus der Beobachtung heraus war das Erstellen von Lernfragen eine sehr motivierende Angelegenheit, weil sich die Studierenden so über den Tag hinweg anstregten, gute Fragen für ihr Lernquiz zu finden und sich dadurch mit den Inhalten des Museums stärker auseinandersetzen als bei einem herkömmlichen Museumsbesuch.

4 Ausblick

Das hier beschriebene Lernszenario lässt sich methodisch auch auf andere Lernorte und weitere Lernsettings transferieren. Als Lehrender einer Lerngruppe können die oben beschriebenen Aufgaben auf andere Museumsbesuche adaptiert werden. Die Lernenden werden in ihren Lernprozessen unterstützt und können sich so intensiver mit den Inhalten auseinandersetzen sowie dazu passende Lernfragen generieren. Durch einen persönlichen mobilen Lernbegleiter, der das Lernen unterstützt, lässt sich ein Mehrwert für die Studierenden erzielen. Eingeschränkt lässt es sich bei Betriebsbesichtigungen umsetzen, da viele Betriebe während der Besichtigung ihres Produktionsbereiches keine mobilen Endgeräte erlauben. Auf bisherigen Exkursionen wurden weitere Formen der Interaktion der Studierenden mit den iPads umgesetzt. Beispielsweise lassen sich gut vorstrukturierte Aufgabenstellungen realisieren, die ein Weiterreichen der iPads

zwischen den einzelnen Gruppen vorsehen, damit diese schon früher an den Lernergebnissen der anderen Gruppen teilhaben. Das didaktische Konzept und dessen Evaluation zeigten auf, wie ein schlanker Ansatz ohne Kosten praktikabel von Lehrenden umgesetzt werden kann, um das Lernen mit Hilfe eines mobilen Lernbegleiters anzureichern. Aufbauend auf den bisherigen Erkenntnissen steht als nächstes das Programmieren einer eigenen App zur Umsetzung didaktischer Methoden an, so dass es in den verschiedenen Lehrveranstaltungen des Lehrstuhls Einsatz finden kann. Mit dieser wird es möglich sein, das Lernen besser zu unterstützen. Bisher gibt es für jede Idee nur Einzellösungen, so dass zwischen diesen gewechselt werden muss. Für das Lernen aus einem Guss, fehlt es aber an einem Komplettpaket. Zudem steht auch eine Anbindung an die ePortfolio-Software Mahara (<https://mahara.org>) an. Diese wird in verschiedenen Lehrveranstaltungen als Persönliche Lernumgebung genutzt, die individuelle und kollektive Arbeitsaufträge beinhaltet. Hier fehlt ein didaktisches Konzept, was weit über Dateiupload und –zugriff hinausgeht. So können weitere Synergieeffekte für das Lernen und seine Organisation erreicht werden.

Literaturverzeichnis

- [Bi09] Biermann, R.: Der mediale Habitus von Lehramtsstudierenden. Wiesbaden: VS 2009.
- [FS10] Froberg, D.; Schenk, B.: Einen Sack voll Flöhe hüten. Lernsteuerung beim mobilen Lernen. In: Breitner, M. H.; Lehner, F.; Staff, J. & Winand, U. (Hrsg.): E-Learning 2010. Berlin u.a.: Physica-Verl. 2010, S. 249-261.
- [Im11] Institut für Medien- und Kompetenzforschung: MMB Monitor 2011. URL: http://www.mmb-institut.de/monitore/trendmonitor/MMB-Trendmonitor_2011_II.pdf
- [KI07] Klein, M.: Exkursionsdidaktik. Eine Arbeitshilfe für Lehrer, Studenten und Dozenten. Baltmannsweiler: Schneider Verlag Hohegehren 2007.
- [Ko10] Kommer, S.: Kompetenter Medienumgan? Eine qualitative Untersuchung zum medialen Habitus und zur Medienkomptenz von SchülerInnen und LehrerInnen. Budrich UniPress: Opladen Farmington Hills 2010.
- [OS99] Oppermann, R. & Specht, M.: Adaptive Mobile Museum Guide for Information and Learning on Demand. In: Bullinger, H.-J. & Ziegler, J. (Hrsg.): Proceedings of 8th HCI International, Mahwah, New Jersey: Lawrence Erlbaum 1999, S. 642-646.
- [RTA11] Rensing, C.; Tittel, S. & Anjorin, M.: Location based Learning Content Authoring and Content Access in the docendo platform. In: Zambonelli, F.; & Midkiff, S. F.: PerCom Workshops 2011: 2011 IEEE International Conference on Pervasive Computing and Communications Workshops. IEEE eXpress Conference Publishing, 2011, S. 165-170.
- [Sc11] Schmitz, D.: Exkursionen im Technikunterricht. In: Zeitschrift für Technik im Unterricht, tu 141(2011), S. 10-17.
- [Va09] Vavoula, G.; Sharples, M.; Rudman, P.; Meek, J. & Lonsdale, P.: Myartspace: Design and evaluation of support for learning with multimedia phones between classrooms and museums. In: Computers and Education, 53 (2009), 2, 286–299.
- [We11] Wegener, R.; Bitzer, P.; Oeste, S. & Leimeister, J. M.: Motivation und Herausforderungen für Dozenten bei der Einführung von Mobile Learning. In: Jahrestagung der Gesellschaft für Informatik Nr. 41, Berlin 2011.

ScratchTab – Eine Tablet-basierte Anwendung zum Erlernen von Programmierkonzepten

Philip Brauner, Hendrik Thüs, Martina Ziefle, Ulrik Schroeder

RWTH Aachen

{brauner,ziefle}@humtec.rwth-aachen.de

{thues,schroeder}@cs.rwth-aachen.de

Abstract: ScratchTab ist eine mobile und vielseitig einsetzbare Programmierumgebung für Tablet-Geräte. Dieser Artikel stellt die Motivation für eine derartige Umgebung dar, beschreibt die Anforderungen und den Entwicklungsprozess, sowie die Ergebnisse einer Nutzerstudie mit Programmierneulingen. Die Ergebnisse der Nutzerstudie zeigen, dass eine derartige mobile Programmierumgebung einerseits das Interesse steigert, sich mit Programmierung zu beschäftigen, und andererseits die Sensoren und Aktuatoren eines Tablet-Geräts neue und vielfältige Einsatzszenarien in der Programmierausbildung bieten können.

1 Motivation

In der heutigen Zeit hat die Informatik in nahezu jeden Bereich des alltäglichen Lebens Einzug gehalten. Elektronische Gegenstände, teilweise programmierbar, werden tagtäglich von der breiten Menge der Bevölkerung genutzt. Die Informatik ist somit ein grundlegender Bestandteil der heutigen Gesellschaft, den es zu fördern gilt. Im Bereich der MINT-Fächer und insbesondere in der Informatik sind Programmierkenntnisse elementar. Um Programmieranfänger möglichst früh für solch eine Tätigkeit zu begeistern, ist es wesentlich, die Interessierten nicht zu überfordern. Programmierparadigmen können Anfänger leicht überlasten und sollten demnach in einer einfachen Art und Weise dargebracht werden. Das hier dargestellte Projekt ScratchTab bietet eine solche Umgebung. Mit den richtigen Schnittstellen können jedoch nicht nur Programmieranfänger angesprochen werden. Auch Personen verschiedensten Alters, die nicht die Absicht haben, sich in die Programmierung einführen zu lassen, können ihren Vorteil von diesem System haben. Es ist beispielsweise denkbar, bestimmte Abläufe zu automatisieren. Die Regeln für diese Prozeduren könnten durch ScratchTab definiert werden, ohne dass die bedienende Person Vorkenntnisse der Programmierung benötigt.

2 Konzept

Bei der Konzeption von ScratchTab stehen drei Aspekte im Vordergrund. Erstens soll die Programmierumgebung eine geringe Einstiegshürde aufweisen um auch von Personen mit geringer Erfahrung im Umgang mit digitalen Medien und insbesondere ohne Programmierkenntnisse nutzbar sein. Zweitens soll die Umgebung auf Tablets basieren und sich von den sonst in der Ausbildung von Programmieranfängern eingesetzten PCs

oder Laptops absetzen. Drittens sollen die auf Tablets inhärent verfügbaren Schnittstellen und Sensoren ausgenutzt und für die Programmierung bereitgestellt werden.

Zur Erfüllung des ersten Punkts, der geringen Einstiegshürde, basiert ScratchTab auf der vom MIT Media Lab entwickelten Programmiersprache „Scratch“ [HR08], die eine Puzzlemetapher nutzt. Programme werden nicht textuell formuliert, sondern durch Befehlskacheln zusammengestellt. Diese haben je nach Funktion unterschiedliche Formen sowie unterschiedliche Aussparungen zur Verzahnung mit anderen Kacheln. Hierdurch wird sichtbar gemacht, welche syntaktische Struktur ein Programm haben kann; Syntaxfehler sind durch diese Restriktionen ausgeschlossen. Die Lernenden der Programmiersprache müssen sich so weniger auf die Syntax konzentrieren und verfügen so nach der Cognitive Load Theory [CS91] über mehr lernbezogene kognitive Kapazität um sich mit dem primären Lerngegenstand, der Semantik der Sprache, auseinanderzusetzen.

Als zweites Ziel wurde festgelegt, dass ScratchTab auf Tablet-Geräten nutzbar sein soll. Diese mobilen Endgeräte erfreuen sich wachsender Beliebtheit. Im Jahr 2011 wurden in Deutschland mehr Tablets als Netbooks verkauft und zusammen mit Smartphones erobern diese zunehmend den Markt der Normalverbraucher [Bu11]. Ein Tablet zu besitzen ist nichts Besonderes und deutet nicht auf ein bestimmtes Interessengebiet des Besitzers hin, die Bedienbarkeit und Beherrschbarkeit ist für Personengruppen jeglichen Alters möglich, was bei herkömmlichen PCs oder Laptops nicht immer gegeben ist. Der mobile Aspekt der Tablets erleichtert es den Nutzern weiterhin, sich leichter mit anderen Personen auszutauschen und die erstellten Ergebnisse zu vergleichen. Des Weiteren ermöglichen solche Geräte den Einsatz in praxisnahen Szenarien. Beispielsweise kann ein System auf einem Tablet unabhängig vom Ort eingesetzt werden.

Ausgehend von Tablets als Medium wurde als drittes Ziel definiert, dass die integrierten Sensoren, Aktuatoren und Schnittstellen direkt in die Programmiersprache integriert werden sollen. Intention ist, dass das Programmieren-Lernen so besonders motivierend gestaltet werden kann. Die Programmierung von Gegenständen außerhalb des eigentlichen Programmiermediums steigert nicht nur die Lernmotivation, sondern gleichzeitig auch den Lernerfolg [Br10]. So werden LEGO Roboter erfolgreich für den Programmierunterricht für Schülerinnen und Schüler eingesetzt [HS05, LS09]. Diese empfinden es als reizvoll, eben nicht den PC zu programmieren, sondern den Robotern autonomes Verhalten beizubringen.

3 Realisierung

Im vorangegangenen Abschnitt wurden Anforderungen an ScratchTab definiert. In diesem Abschnitt wird nun die Umsetzung in Bezug auf die Gestaltung der Benutzeroberfläche, die verfügbaren Programmbausteine und die Programmausführung dargestellt.

3.1 Benutzeroberfläche

Die Benutzeroberfläche gliedert sich in drei Komponenten (vgl. Abbildung 1): Eine Steuerungsleiste mit den verfügbaren Programmbausteinen, sowie die Arbeitsfläche zum Zusammenstellen der Programme.

Die Steuerungsleiste dient dem Starten und Beenden von Programmen, der Konfiguration der Bluetooth-Verbindung zu anderen Geräten, sowie dem Verstellen der Zoomstufe. Zukünftig werden noch Funktionen zum Laden und Speichern von Programmen ergänzt. In der Seitenleiste befinden sich die nach Kategorien sortierten verfügbaren Befehlsblöcke. Durch diese Liste kann vertikal gescrollt werden; durch einen Tap auf den Namen einer Kategorie ist es ebenfalls möglich, diese direkt anzusteuern.

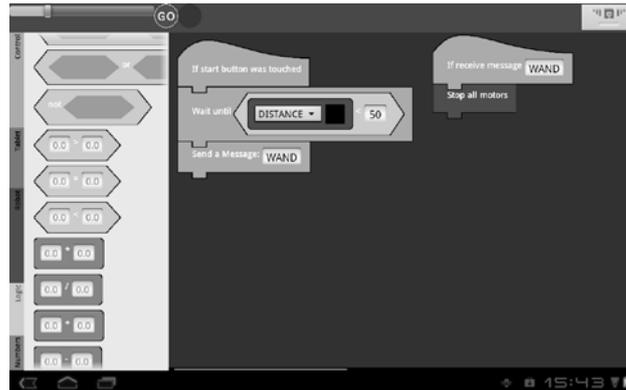


Abbildung 1: Oberfläche der Programmierumgebung ScratchTab (oben: Steuerungsleiste, links: Leiste mit verfügbaren Programmbausteinen, große Fläche rechts: Arbeitsfläche)

3.2 Programmbausteine

Analog zu Scratch bedient sich ScratchTab einer Puzzlemetapher und verfügt über viele Bausteine, aus denen die Lerner Programme zusammenstellen können. Form und etwaige Aussparungen der Blöcke stellen sicher, dass nur syntaktisch kombinierbare Blöcke tatsächlich kombiniert werden können. Die Kombination von Programmbausteinen erfolgt durch eine Drag-und-Drop-Geste, bei der ein Baustein entweder von der Arbeitsfläche oder aus einer Leiste mit allen verfügbaren Bausteinen herausgezogen und an andere Bausteine angedockt wird (vgl. Abbildung 2). Während des Drag-Vorgangs werden die möglichen Andockpunkte visuell hervorgehoben.

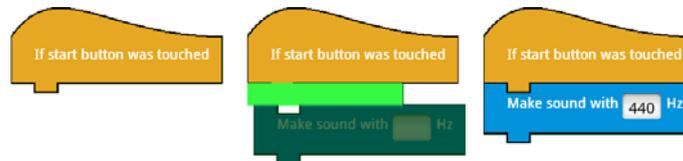


Abbildung 2: Startblock (links), Verknüpfungsvorgang von Startblock mit einem Kommandoblock (Mitte), abgeschlossene Verknüpfung (rechts)

Bausteine können untereinander gereiht werden, was für eine Programmsequenz steht. Einige Bausteine verfügen auch über „Slots“, in die andere Bausteine hierarchisch eingeordnet werden. So enthält ein Schleifenbaustein zwei Slots für die Bausteine im Schleifenrumpf und für die Bausteine im Schleifenkopf (z.B. Abbruchbedingung). In der derzeitigen Entwicklungsstufe sind Programmbausteine für grundlegende Kontrollstruk-

turen wie Bedingungen, Schleifen mit fixen Wiederholungen oder dynamischen Abbruchbedingungen, Logikoperatoren und arithmetischen Funktionen umgesetzt. Ferner können die Sensoren und Aktuatoren des Tablets genutzt werden. Alle Programmbausteine untergliedern sich in Kategorien, die im Folgenden dargestellt werden.

Zur Kategorie „Programmablauf“ gehören Bausteine für IF-THEN-ELSE-Blöcke oder numerischen oder ausdrucksorientierten Schleifen (FOR i bzw. WHILE CONDITION).

Für Berechnungen existieren Bausteine für Additionen, Multiplikationen und dergleichen. Als Operanden dienen entweder vom Programmierer vorgegebenen konstante Zahlen oder auch Sensorwerte des Tablets. Analog können mit den Bausteinen für logische Vergleiche Zahlen auf Gleichheit oder Größenunterschiede untersucht werden. Ferner existieren boolesche Operatoren für das logische UND, ODER und NICHT.

Die Sensoren und Aktuatoren des Tablets können über Programmbausteine der Kategorie „Tablet“ angesteuert werden. Derzeit stehen zur Ausgabe die Bausteine „Make sound“, sowie „Make a screenflash“ zur Verfügung. Zukünftig sollen auch komplexerer Soundsamples oder Vibrationen angeboten werden. Als Sensoren stehen die auf Tablet-Geräten häufig verfügbaren Beschleunigungssensoren, ein Kompass oder ein Lichtsensor zur Verfügung.

Wie im Kapitel Konzept dargelegt wurde, sollen ScratchTab-Programme sich nicht nur auf das Tablet auswirken, sondern auch auf Ereignisse außerhalb des Tablets reagieren können. Als Machbarkeitsbeweis wurde eine Verknüpfung zu einem Lego Mindstorms Roboter realisiert. Durch Bausteine der Kategorie „Roboter“ kann ein über eine Bluetooth-Schnittstelle verbundener Roboter gesteuert und dessen Sensorwerte abgefragt werden. Statt einer direkten Steuerung der einzelnen Motoren wurde auf das vereinfachende Konzept der Turtle-Grafik [Pa80] mit Befehlen wie „TURN LEFT“ oder „DRIVE FORWARD“ zurückgegriffen. Von den Sensoren können die für Entfernung, Geräusche oder für Berührungen verwendet werden. Mit wenigen Kacheln lässt sich so beispielsweise ein Programm schreiben, das in Abhängigkeit der Neigung des Tablets verschieden hohe Töne ausgibt. Auch ist es leicht möglich, über das Tablet einen Roboter so zu programmieren, dass dieser beim Berühren einer Wand umkehrt oder gar autonom aus einem Labyrinth navigiert. Das kreative und motivierende Potenzial von ScratchTab liegt jedoch vor allem in der Kombination der verschiedenen Befehlskacheln.

3.3 Starten von Programmen

Programme können über drei Wege gestartet werden: Erstens über den grünen Startknopf aus der Steuerungsleiste und eines dazugehörigen Startbausteins auf der Arbeitsfläche. Wird der Startknopf ausgewählt, beginnt die Programmausführung bei den Startblöcken der jeweiligen Programmsegmente. Zweitens können Programmsegmente durch das Empfangen von Nachrichten aktiviert werden, die entweder durch Sensorwerte ausgelöst werden oder aber aus dem Programm selbst gesendet werden (z.B. zum Aufrufen von Unterprozeduren). Drittens lassen sich Programmsegmente durch einen Doppel-Tap auf einem Baustein starten. Je nach Programmstruktur kann ein Programmsegment selbstständig terminieren oder durch den Stopp-Knopf in der Steuerungsleiste beendet werden. Zur Verfolgung des Programmablaufs wird zum einen die Ausführungsge-

schwindigkeit künstlich reduziert, so dass die Ausführung eines Programmbausteins etwa 250ms benötigt. Zum anderen leuchtet der jeweils aktive Baustein für diese Zeitspanne auf.

3.4 Vergleichbare Applikationen

Die beschriebene Puzzlemetapher wird nicht nur in der schon beschriebenen Applikation Scratch eingesetzt. Andere Beispiele von Applikationen sind etwa Googles App Inventor¹ oder das von Microsoft entwickelte TouchDevelop². Beide Projekte können zur Entwicklung von Applikationen für mobile Geräte genutzt werden. Das bedeutet, sie bieten eine andere Möglichkeit der Programmierung auf stationären PCs. Es können während dieser Tätigkeit keine Mehrwerte von mobilen Geräten genutzt werden, da diese erst für die Ausführung eingesetzt werden und nicht schon während der Entwicklung.

4 Evaluation

Die Gebrauchstauglichkeit der hier vorgestellten Programmierumgebung wurde regelmäßig untersucht. Während der Entwicklung wurden Nutzertests und Expertenreviews periodisch durchgeführt, um die vielfältigen Designentscheidungen zu fundieren. Abschließend wurde eine formale Nutzerstudie mit 10 Probanden durchgeführt, bei der drei Programmieraufgaben mit ScratchTab bearbeitet werden sollten.

4.1 Evaluation während des Entwicklungsprozesses

Parallel zur Entwicklung wurde die Gebrauchstauglichkeit durch Expertenreviews und Nutzertests in periodischen Abständen überprüft. Prüfungsgegenstand waren etwa Darstellungsgröße der Programmierbausteine oder die Frage, wie ineinander verschachtelte Bausteine (z.B. FOR-Schleifen) darzustellen sind. Die Ergebnisse flossen direkt in die weitere Entwicklung ein. So wurde etwa ausgehend von Nutzerwünschen eine stufenlose Zoomfunktion der Arbeitsfläche geschaffen. Zur Darstellung von verschachtelten Bausteinen präferierte in einer Onlineumfrage (n=94) mit 70% eine deutliche Mehrheit eine Darstellung, bei der einerseits alle Blöcke eine konstante Breite haben und andererseits die Verschachtelungstiefe direkt erkennbar ist.

4.2 Summative Evaluation des derzeitigen Prototypen

ScratchTab hat hinsichtlich Programmfunktionalität und -stabilität einen Reifegrad erreicht, der es ermöglicht, die Umgebung summativ und stärker formalisiert zu evaluieren. Die Evaluation ist explorativ angelegt und soll einen Eindruck von der Gebrauchstauglichkeit liefern, Verbesserungspotential identifizieren und weitere Anwendungsszenarien aufzeigen.

¹ <http://info.appinventor.mit.edu/>

² <http://research.microsoft.com/en-us/projects/touchdevelop/>

Für die Studie wurden den 10 Probanden sequenziell drei Aufgaben steigender Komplexität gestellt. In der ersten Aufgabe sollten sich die Probanden mit der Umgebung vertraut machen und ein Programm erstellen, das über den Lautsprecher drei aufsteigende Töne ausgibt. In Aufgabe 2 war über ein Schleifenkonstrukt und einen darin verschachtelten Noten-Block wiederholt ein Ton mit gleichbleibender Tonhöhe auszugeben. Aufgabe 3 erfordert die Integration des Neigungssensors in das Programm; es war solange ein Ton auszugeben, bis das Tablet um einen gewissen Winkel geneigt wurde (die Musterlösungen zu den Aufgaben finden sich in Abbildung 3). Zur Lösung der ersten Aufgabe sind mindestens 4 Blöcke nötig (Start-Block, sowie 3 Noten-Blöcke). Die zweite Aufgabe erfordert 3 Blöcke (Start-Block, Schleife und ein Noten-Block). Die dritte und letzte Aufgabe erfordert 5 Blöcke (Start-Block, Wiederhole-Bis-Block mit einem Kleiner-Block für die logische Prüfung und darin der Block für die Neigungssensorabfrage, sowie ein Noten-Block).

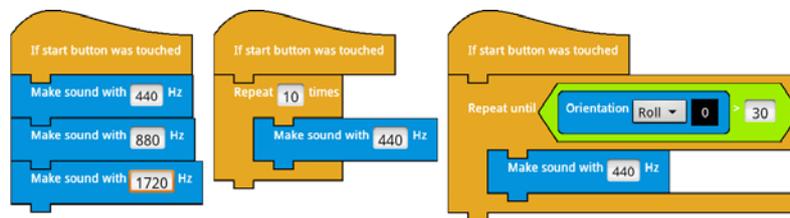


Abbildung 3: Musterlösungen für die Aufgaben 1, 2 und 3 (v.l.n.r.)

Eine Einführung in die Programmierumgebung fand nicht statt, stattdessen sollten die Probanden die Funktionsweise der Umgebung selbstständig erkunden. Erst auf Anfrage oder bei wiederkehrenden Fehlern gaben die Versuchsleiter kurze Hinweise.

An der Studie nahmen 10 Personen teil (4 Frauen und 6 Männer). Die Probanden waren Studierende und Mitarbeiter unseres Lehrstuhls. Das Alter reichte von 23 bis 31 Jahren. Von den Probanden gab die Hälfte an, über kein oder lediglich elementares Programmierwissen zu verfügen. Vier Personen bewerteten ihre Programmiererfahrung als „mittel“ und eine Person gab an, über viel Programmierwissen zu verfügen. Letztgenannte Person hat Informatik studiert und programmiert sowohl privat wie auch beruflich.

Wenig überraschend hatte der Informatiker keine Schwierigkeiten, die Aufgaben zu lösen. Selbst die in der dritten Aufgabe nötige mehrfache Verschachtelung von Bausteinen bereiteten ihm keine Schwierigkeiten. Auch die anderen Probanden mit keiner oder wenig Programmiererfahrung konnten die Aufgabe 1 problemlos lösen. Aufgabe 2 wollte etwa ein Drittel ohne den Einsatz eines Schleifenbausteins lösen. Nachdem die Funktion dieses Bausteins jedoch erneut erläutert wurde, wurde auch dieser erfolgreich eingesetzt. Die 3. Aufgabe stellte etwa die Hälfte der Probanden vor größere Schwierigkeiten, die nur mit Hilfestellungen gemeistert werden konnten. Zwar kamen alle Probanden auf die Idee, ein Schleifenkonstrukt einzusetzen, jedoch musste zum einen die Wahl des richtigen Konstrukts gelegentlich unterstützt werden, zum anderen wollten die Probanden häufig direkt den entsprechenden Sensorwert (Zahlenwert) in die Schleifenbedingung (Boolescher Wert) einsetzen. Die hier nötige Umwandlung des Sensorwerts in einen booleschen Ausdruck durch einen Vergleichsbaustein musste erst und teils wie-

derholt erläutert werden. Hier scheint der Einsatz einer visuellen Programmiermetapher der Forschung die Möglichkeit zu eröffnen, die kognitiven Prozesse beim Lernen der Programmierung besser beobachten zu können, als bei einer textuellen Programmiersprache. Festzustellen ist, dass nicht nur der Informatiker, sondern alle Probanden die drei Aufgaben erfolgreich lösen konnten.

Bei allen Versuchsteilnehmern traten verschiedene Interaktionsfehler auf. So wurde gelegentlich ein einzuschachtelnder Baustein nicht in den vorgesehenen Slot, sondern daneben abgelegt. Offenbar ist die Fläche eines Slots mit $5 \times 5 \text{ mm}^2$ zu klein, um Präzise Drag-und-Drop Operation ausführen zu können, da der Finger den Zielbereich verdeckt.

Gefragt nach möglichen Einsatzbereichen äußerte mit 6 von 10 Probanden die Mehrheit den Einsatz in der Ausbildung von Programmieranfänger/-innen. Hier wurde von einigen Probanden der Einsatz in der Grundschule und im Kinderkarten genannt, um dort bereits frühzeitig die Grundzüge der Programmierung zu vermitteln. Dies würde sich mit den Ansätzen und guten Erfahrungen der ElectronicBlocks [WP03] decken. Weniger offensichtlich waren die mehrfach genannten Vorschläge ScratchTab für Gebäudeautomation einzusetzen. Ebenfalls wurde empfohlen ScratchTab in Richtung eines Baukastensystems für kognitionspsychologische Experimente auszubauen. Es soll ein einfacher Weg gefunden werden, damit Psychologen ohne Programmiererfahrung Experimente frei gestalten können, bei denen verschiedene Stimuli dargeboten werden, Reaktionszeiten bestimmt und in Abhängigkeiten von der Nutzerantwort unterschiedliche Feedbacks gegeben werden können.

Abschließend sollten sich die Probanden auf einer 6-stufigen Likert-Skala dazu äußern, ob ihnen die Arbeit mit ScratchTab Spaß gemacht habe, ob sie die Umgebung gut lernbar fanden und ob sie eine visuelle Programmiersprache für geeigneter halten als textuelle. Alle Probanden äußerten, dass ihnen die Arbeit mit ScratchTab Spaß gemacht habe (keine negativen Äußerung, 2 stimmten eher zu, 2 stimmten zu und 6 stimmten voll zu) und dass die Umgebung schnell zu erlernen sei (keine negativen Äußerungen, 1 Person stimmte eher zu, 3 stimmten zu und 6 stimmten voll zu). Die visuelle Programmierung wird von einer Mehrheit, jedoch nicht von allen Probanden bevorzugt (3 Personen bevorzugen eher eine textuelle Sprache, 7 Personen bevorzugen eine visuelle Sprache).

5 Fazit und Ausblick

Obwohl die hier vorgestellte Programmierumgebung in einem frühen Prototypenstatus ist, zeigt der erste Nutzertest gute Ergebnisse und lieferte Hinweise für Verbesserungen an der Benutzerschnittstelle und Anregungen für weitere Einsatzmöglichkeiten, die weit über den Einsatz in der Programmierausbildung hinausgehen.

Es ist hervorzuheben, dass ScratchTab eine der wenigen Programmierumgebungen ist, die für den Gebrauch auf mobilen Endgeräten mit kleinen Bildschirmen entwickelt wurde. Durch die Möglichkeit, ortsunabhängig zu programmieren, eröffnet die Applikation eine Reihe von Anwendungsgebieten, die mehr Interaktivität zulassen oder Mobilität der Nutzer voraussetzen. Durch die Schnittstellen der mobilen Endgeräte – wie Bluetooth und WLAN – wird das Kommunizieren mit anderen Geräten, wie Robotern, vereinfacht.

Nutzer/-innen haben so die Möglichkeit, ihre mit ScratchTab erstellten Programme direkt auf dem ausführenden Gerät zu testen, Erfolge werden so direkt sichtbar.

Während der Nutzertests wurden weitere Schnittstellen vorgeschlagen, die in Zukunft eingearbeitet werden sollen. Denkbar ist etwa die Steuerung von Arduino-Mikrocontrollern, mit denen sich eine Fülle weiterer Möglichkeiten eröffnet.

Danksagungen

Die Autoren bedanken sich bei Alexander Friesen, der große Teile von ScratchTab realisiert hat. Auch danken wir den Teilnehmerinnen und Teilnehmern der Nutzertests.

Literaturverzeichnis

- [Br10] Brauner, P. et al.: The effect of tangible artifacts, gender and subjective technical competence on teaching programming to seventh graders. In: Hromkovic, J. et al. (eds.) Proceedings of the 4th International Conference on Informatics in Secondary Schools: Evolution and Perspective 2010 (ISSEP), Zurich. pp. 61-71 Springer, Heidelberg (2010).
- [Bu11] Bundesverband Informationswirtschaft, T. und neue M. e. V.: Tablet Computer erobern den Massenmarkt, http://www.bitkom.org/61374_70631.aspx, erstellt am 14.10.2011, zuletzt besucht am 14.08.2012.
- [CS91] Chandler, P., Sweller, J.: Cognitive Load Theory and the Format of Instruction. *Cognition and Instruction*. 8, 4, 293-332 (1991).
- [HS05] Hartmann, S., Schecker, H.: Bietet Robotik Mädchen einen Zugang zu Informatik, Technik und Naturwissenschaft? - Evaluationsergebnisse zu dem Projekt Roberta. *Zeitschrift für Didaktik der Naturwissenschaften*. 11, 7-19 (2005).
- [HR08] Hernández, A.M., Resnick, M.: FEATURE: Empowering kids to create and share programmable media. *Interactions*. 15, 2, 50-53 (2008).
- [LS09] Leonhardt, T., Schroeder, U.: go4IT!: Initiierung und nachhaltige Förderung von Interesse an MINT-Fächern bei Mädchen. *Informatische Bildung in Theorie und Praxis, Beiträge zur INFOS 2009*, 13. GI-Fachtagung - Informatik und Schule. , Berlin (2009).
- [Pa80] Papert, S.: *Mindstorms: Children, Computers, and powerful Ideas*. Basic Books, Inc. New York, NY, USA. (1980).
- [WP03] Wyeth, P., Purchase, H.C.: Using developmental theories to inform the design of technology for children. *IDC '03: Proceedings of the 2003 conference on Interaction design and children*. pp. 93-100 ACM, New York, NY, USA (2003).

Hardware und Software für mobiles Assessment im Schulbereich

Mahtab Dalir¹, Heiko Rölke²

^{1,2} Deutsche Institut für Internationale Pädagogische Forschung

Schloßstraße 29

60486, Frankfurt am Main

¹dalir@dipf.de

²roelke@dipf.de

Zusammenfassung: Mobile Endgeräte wie Tablets oder Smartphones sind mittlerweile in allen Bereichen des täglichen Lebens verbreitet. Ihre Nutzung zum Lernen und Testen in Schule und Freizeit ist nicht mehr nur eine Zukunftsvision. Zur Umsetzung wird ein vielseitiges und umfassendes System aus Hard- und Software benötigt, das die verschiedenen Arten von Lern- oder Testeinheiten präsentieren kann. Darüber hinaus müssen alle weiteren Aspekte einer Langzeittestung abgedeckt werden, wie zum Beispiel Testsicherheit, Datensicherung, Anpassbarkeit an neue Zeitabläufe usw. Erst dann kann ein echter Mehrwert gegenüber dem bereits praktizierten Einsatz von Computern zum Lernen und Testen erzielt werden. In diesem Beitrag präsentieren wir an einem praktischen Beispiel das Konzept und die Realisierung eines solchen Systems. Es wird im Rahmen von wissenschaftlichen Untersuchungen der täglichen Schwankungen kognitiver Leistungsfähigkeiten im Schulkontext eingesetzt.

1 Einleitung

Das tägliche Leben wird immer mehr durch die Nutzung von *Smartphones* und anderen mobilen Computern beeinflusst. Dieses „*not being tied to particular locations and times*“ [HNM05] ist nicht auf bestimmte Lebensbereiche beschränkt. Mobilität ist auch in den Bereichen Lernen und Assessment interessant, insbesondere auch im Schulkontext.

Für bestimmte Arten von Studien ist es wichtig, dass die Tests über einen längeren Zeitraum und mehrmals am Tag durchgeführt werden. Darüber hinaus sollten die Tests möglichst gut in den normalen Tagesablauf integriert sein (nicht-invasives Testen), um zum einen organisatorische Probleme zu vermeiden und zum anderen die Gültigkeit der Daten zu gewährleisten. Um diesen Herausforderungen gerecht zu werden, können mobile Technologien zum Einsatz kommen. Kostengünstige mobile Geräte, die sowohl ausreichend leistungsfähig für Testzwecke sind als auch geeignet für einen längeren Einsatz mit Kindern, waren allerdings bis vor kurzem nur schwer zu finden. Dies hat sich mit dem Aufkommen der Smartphones und insbesondere der Android-Plattform geändert. Mit Smartphones oder Tablets zu testen hat noch einen weiteren positiven Nebeneffekt: Kinder begeistern sich für die Verwendung von neuen und „coolen“ Technologien was wiederum – möglicherweise - ihre Motivation erhöht.

Unsere Gruppe wurde für die technische Unterstützung einer Reihe von psychometrischen Studien angefragt. In diesen Studien werden Schwankungen kognitiver Ressourcen von Schulkindern täglich über mehrere Wochen innerhalb von vollständigen Schulklassen erfasst. Etwa 120 Kinder zwischen 9 und 10 Jahren wurden dafür rekrutiert. Um die Studien technisch zu ermöglichen, haben wir eine vielseitige und umfassende Lösung zum mobilen Assessment entworfen, die auch für Lernzwecke eingesetzt werden kann. Unser System hat Schulkinder als Zielgruppe, ist aber nicht darauf beschränkt. Das System baut auf offenen Standards auf und ist im Hinblick auf Aufgaben- und Lerninhalten nicht eingeschränkt. Besonderer Wert wurde auf Faktoren wie Sicherheit, autonome Testauslieferung und automatisierte Datensynchronisation gelegt.

Der folgende Abschnitt dieser Arbeit gliedert die Anforderungen an unser mobiles Assessment-System, gefolgt von der System-Architektur in Abschnitt 3. Anschließend wird in Abschnitt 4 eine Fallstudie präsentiert, die auf unser System aufbaut. Zum Schluss werden verwandte Arbeiten präsentiert und ein Ausblick gegeben.

2 Systemanforderungen

Unser Ziel ist die technische Unterstützung von mobilem Assessment, besonders im Schulkontext. Die Anforderungsanalyse zur Umsetzung dieses Ziels ergibt drei wesentliche Anforderungsbereiche:

- Inhalte: Große Auswahl von Aufgabentypen, externe Erstellung oder Generierung von Aufgaben und Nutzung von Standards zur Aufgabenbeschreibung
- Mobile Hard- und Software: Kostengünstig und zuverlässig, einfacher Zugriff auf Systeminterna
- Testauslieferung: Unüberwachte Testdurchführung, insbesondere außerhalb der Schule (Nachmittags, Wochenende), Testsicherheit und Datenschutz

Der wichtigste Teil eines Assessment-Systems ist der Inhaltsbereich, also die Erzeugung und Auslieferung von Tests. Das System sollte zum einen zahlreiche Item-Typen unterstützen und dabei die bestehenden Standards zur Inhaltsbeschreibung berücksichtigen. Dadurch kann Interoperabilität und Wiederverwendbarkeit sichergestellt werden. Unsere Wahl fiel auf die *IMS Question & Test Interoperability Specification (QTI)* [IMS11] als Ausgangspunkt. Obwohl dieser Standard bestimmte Defizite hat, bietet er die Möglichkeit der Verwendung vorhandener externer (auch PC-basierter) Tools sowie von Assessment-Inhalten. Darüber hinaus wird eine spätere Integration unseres Systems in Learning Management Systeme wie Moodle [RE12, MO12] einfacher.

Ein weiterer wichtiger Entscheidungsfaktor für jede mobile Applikation ist die Wahl des zugrunde liegenden Hardware-Systems (und damit auch der zugehörigen Softwareumgebung). Es ist prinzipiell möglich, Hardware-unabhängige mobile Software zu entwickeln, z.B. durch die Verwendung von Browser-Technologie. In der Realität ergibt

sich daraus aber ein nur schwer aufzulösender Widerspruch zu zentralen Anforderungen der Testsicherheit und der autonomen Testdurchführung.

Wir haben deshalb beschlossen, eine native Anwendung statt einer Web-Applikation zu entwickeln. Dafür muss aber eine Plattform ausgewählt werden. Es gibt viele Möglichkeiten wie Apples iOS, Android, Samsungs Bada, Windows Mobile, MeeGo und BlackBerry, um nur einige zu nennen. Wir haben uns für die Android-Plattform entschieden. Der wichtigste Grund dafür war die reiche Auswahl an vorhandener Hardware und die gute Verfügbarkeit von Software und Entwicklungswerkzeugen.

Testsicherheit wurde bereits als eine wichtige Anforderung eingeführt. Es muss soweit als möglich verhindert werden beim Test zu schummeln, die Ergebnisse zu fälschen, Items zu kopieren oder den Zugang zu den Daten anderer Testteilnehmer zu erlangen. Darüber hinaus müssen wir unüberwachte Tests z.B. am Nachmittag oder Wochenende ermöglichen. Daher müssen alle Prozesse wie Testdauer, Datensynchronisation und Updates vollständig unter unserer Kontrolle sein. Dafür werden die Smartphones abgeschottet und Benutzerinteraktionen auf die Durchführung der Tests beschränkt.

3 System-Architektur

Abbildung 1 stellt die entworfene Systemarchitektur dar:

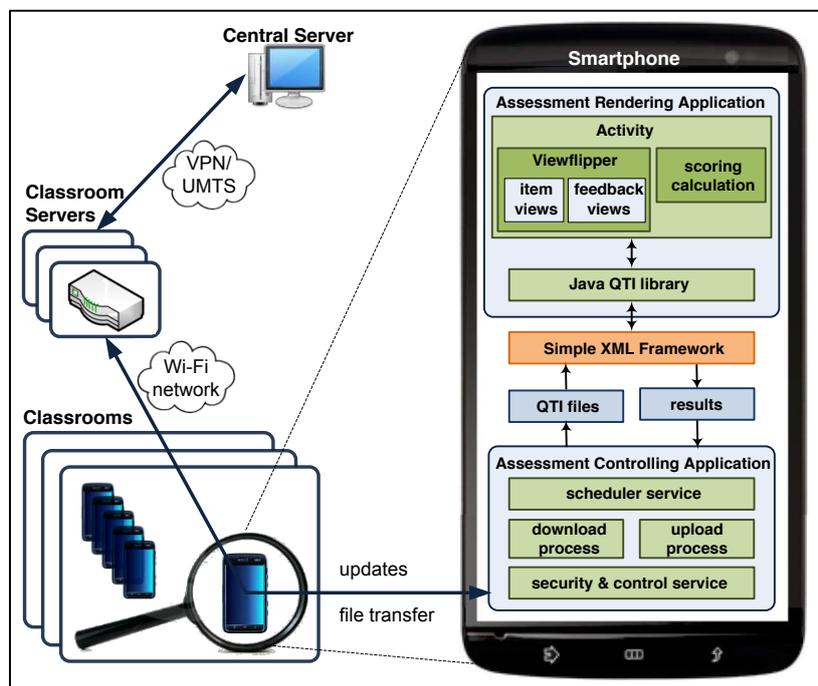


Abbildung 1: Aufbau des mobilen Assessment-Systems

Das System besteht aus drei wesentlichen Teilen: der „*Assessment Rendering Application*“ (ARApp), und der „*Assessment Controlling Application*“ (ACApp), die beide auf den Smartphones ausgeführt werden, sowie der Server-Infrastruktur. Wir beschreiben diese Systembestandteile in den nächsten Abschnitten.

3.1 Assessment Rendering Application (ARApp)

ARApp stellt die Tests und Aufgaben dar, führt sie aus und zeigt die Testergebnisse und Rückmeldungen an den Benutzer an. Die Applikation wurde nach dem Entwurfsmuster Model-View-Controller (MVC) [GHJ94] konzipiert. Von den Android-Komponenten übernehmen die *Activities* [REA12] die Aufgabe von Controller. Sie initialisieren das Modell, stellen die Zugriffsmethoden für das Abrufen von Daten aus dem Modell bereit. Sie behandeln die Interaktionen des Benutzers und benachrichtigen das Modell automatisch über Statusänderungen [RL10]. Die Komponente namens *Viewflipper* [REV12] übernimmt die Rolle des Views. Der *Viewflipper* ist ein Container für die Benutzeroberflächen für Testaufgaben und Rückmeldungen. Die Testergebnisse werden wiederum in *Activities* berechnet und stufenweise nach jedem definierten Zeitraum an den Benutzer zurückgemeldet.

Smartphones sind nur bis zu einem gewissen Grad mit üblichen PCs kompatibel. Einschränkungen bestehen vor allem im Bereich der Bildschirm- und Speichergröße auf der Hardware-Seite und bei Anpassungen an den Betriebssystemen und Programmiersprachen, die auf der Software-Seite unterstützt werden. Die Inhalte und die Applikationen, die für mobiles Assessment eingesetzt werden, können daher nicht direkt von vorhandenen computerbasierten Assessment-Systemen auf Smartphones übertragen werden. Dafür stellt die ARApp ein Framework bereit, um mobiles Assessment auf Android-Plattform zu unterstützen und die Kompatibilitätseinschränkungen zu minimieren. Unser System akzeptiert die Assessment-Inhalte im QTI-Format. QTI ist eine der meistverwendeten Spezifikationen im Assessment- und Lernbereich. QTI stellt verschiedene Vorlagen unter anderem für Aufgabenformate mit *Multiple-Choice*, *Matches*, *Hot-Spots* und *Feedback* bereit [IMS11]. Für das Bearbeiten des QTI XML benutzen wir ein Open-Source Framework namens *Simple* [SIM12]. Simple unterstützt den Prozess der Serialisierung von Java-Objekten zu XML und der Deserialisierung von XML zurück zu Java. Es steht allgemein für die Java-Plattform zur Verfügung und ist im Vergleich zu den entsprechenden Android-Bibliotheken flexibler und einfacher zu benutzen [CH11]. Die deserialisierten Dateien müssen interpretiert und ausgeführt werden. Dafür haben wir eine Java-Bibliothek entwickelt, die die QTI XML-Struktur abbildet. Diese Java-QTI-Bibliothek dient als Domain-Modell und stellt Aggregationen und Listen von QTI Tests und Aufgaben dar. Die Unterschiede zu anderen vorhandenen Bibliotheken sind im Abschnitt über verwandte Arbeiten erläutert.

3.2 Assessment Controlling Application (ASApp)

Dieser Teil unseres mobilen Assessment-Systems ist eine separate Android-Applikation, die hauptsächlich als Ablaufsteuerung für das Ausführen von Instanzen der ARApp

dient. Andere Aufgaben dieser Applikation sind die Koordination der Datenübertragung, die Verwaltung der Sicherheitsaspekte sowie das Sperren des Smartphones.

Eine Scheduler-XML-Datei beschreibt die Konfigurations- und Initialisierungseinstellungen für ein vollständiges Assessment¹ auf einem Smartphone. Die Datei spezifiziert die Daten und Zeitfenster über die auszuführenden Prozesse und steuert deren Verhalten. Ein Prozess kann zum Beispiel eine Instanz der ARApp, ein Download- oder Upload-Vorgang sein. Es ist möglich, für jeden Benutzer separate Ablaufpläne und Testmaterialien zu definieren.

Die ACApp stellt die Verbindung zwischen den Smartphones und dem Classroom Server her. Diese drahtlose Verbindung ist notwendig, um neue Assessment-Inhalte und Software-Updates vom Server auf die Smartphones herunterzuladen und die Ergebnisse der Tests auf den Server hochzuladen. Die Smartphones müssen dafür konfiguriert werden, mit einem Classroom Server kommunizieren zu können. Zu diesem Zweck gibt es eine Konfigurationsdatei mit den notwendigen Informationen für den Serverzugriff. Einer der Einträge dieser Konfigurationsdatei ist die Benutzer-ID. Diese ID wird verwendet, um zwischen den hochgeladenen Dateien aus verschiedenen Smartphones auf dem Server zu unterscheiden. Da in unüberwachten Tests Sicherheit eine wichtige Rolle spielt, bietet das Controlling-System die erforderlichen Sicherheitsmechanismen, um zu verhindern, dass fehlerhafte Daten oder gefälschte Ergebnisse erfasst werden. Der erste Schritt dazu ist das Umstellen des Start- und Home-Bildschirms des Smartphones auf eigenen Home-Bildschirm. Dieser Home-Bildschirm bietet keinerlei Interaktionsmöglichkeiten. Andere Funktionen der Smartphones, die nicht für die Tests relevant sind, werden deaktiviert. Somit sind die Benutzer nicht in der Lage, das System zu manipulieren.

3.3 Server-Infrastruktur

Unabhängig von der Client-Seite des Systems ist die Entwicklung der Server-Infrastruktur ebenfalls eine große Herausforderung. Die Server-Infrastruktur stellt ein Authentifizierungs- und ein Sicherheitssystem für die Datensynchronisation bereit und besteht aus einem Central Server und mehreren Classroom Servern. Die Classroom- und der Central Server sind durch ein *Point-to-Point Virtual Private Network (VPN)* verbunden. Die Classroom Server bieten eine WLAN-Verbindung zu den Smartphones an. Dabei ist kein Internetanschluss in den Klassenzimmern notwendig, da die Classroom Server Mobilfunk-Verbindungen nutzen, um sich mit dem Central Server zu verbinden. Smartphones sind so konfiguriert, dass sie sich in regelmäßigen Abständen mit einem Classroom Server verbinden, um die Testergebnisse zu exportieren.

Wir haben uns für zwei unabhängige und parallel zueinander betriebene Verbindungen zwischen den Classroom Servern und dem Central Server entschieden. Die erste (und bevorzugte) Möglichkeit besteht darin, das vorhandene Schul-Netzwerk zu verwenden und die Classroom Server in dieses Netzwerk zu integrieren. Jeder Classroom Server

¹ Ein solches Assessment kann sich über einen langen Zeitraum erstrecken und zahlreiche Einzeltestungen beinhalten.

stellt seine Verbindung zum Central Server über ein VPN bereit, um die Datensicherheit zu gewährleisten. Die Classroom Server sind mit mobilen Breitband-Modems (UMTS-Standard) ausgestattet. Die darüber aufgebaute autonome Internet-Verbindung bietet eine sekundäre VPN-Verbindung, die auch parallel zur ersten laufen kann. Die zweite Verbindungsmöglichkeit kommt ins Spiel, wenn die erste Verbindung fehlschlägt oder in einer Schule nicht zur Verfügung steht.

Der Central Server wird mit zwei getrennten Netzwerk-Schnittstellen ausgestattet, die jeweils für eine VPN-Verbindung vorgesehen sind. Dies ist notwendig, um beide Verbindungen (UMTS und Schulnetz) unterstützen zu können, die jeweils anders konfiguriert und geroutet sind. Sobald eine Verbindung aufgebaut ist, können die Datensynchronisationen oder Updates stattfinden. Die Verbindung wird auch für einen sicheren Remote-Zugriff auf die Classroom Server für administrative Einstellungen verwendet. Als Beispiel: Wenn der Netzwerkverkehr auf Grund einer Datensicherung zu hoch ist, könnte der Datentransfer angehalten oder verzögert werden. Der Central Server bietet eine einfache Web-Oberfläche u.a. zum Hochladen von Updates. Die Updates werden sofort auf die Classroom Server übertragen und an die Clients verteilt.

4 Fallstudie: FLUX

Das Projekt *Assessment of Cognitive Performance FLUctuations in the School ConteXt* (FLUX)² befasst sich mit täglichen Schwankungen der kognitiven Leistungsfähigkeit von Kindern im Schulkontext mittels Methoden des ambulanten Assessments. Kognitive Aufgaben zur Erfassung der Denkleistung (z.B. Arbeitsgedächtnis, Verarbeitungsgeschwindigkeit), sowie kurze Fragen zu Stimmung, Motivation und Schlafverhalten kommen drei- bis viermal täglich über mehrere Wochen zum Einsatz [DI11].

Die Aufgaben sind meist einfach aufgebaut, als Interaktionen kommen vor allem *choice*, *text entry*, *slider*, *hotspot* und *graphic order* vor. Abbildung 2 zeigt Aufgaben, die für FLUX entworfen wurden.



Abbildung 2. Beispielaufgaben³

² Die Durchführung der FLUX wurde durch die LOEWE Initiative der Hessischen Landesregierung gefördert.

³ Bitte beachten Sie, dass sämtliche Rechte an den Bildern dem FLUX-Projekt gehören.

Aufgabe 1 ist vom Typ *Single-Choice* und erfasst Verarbeitungsgeschwindigkeit. Der Teilnehmer muss entscheiden, ob die Bilder auf der linken und der rechten Seite identisch sind. Die Antwort wird durch Drücken einer der Tasten im unteren Teil des Bildschirms gegeben. Nach dem Drücken wird die nächste Aufgabe automatisch angezeigt. Aufgaben 2 und 3 sind *Multiple-Choice* Aufgaben zum Arbeitsgedächtnis. Die Aufgaben erfassen die Fähigkeit, Informationen aufrecht zu erhalten, zu verarbeiten und bei Bedarf zu aktualisieren. Aufgabe 2 zeigt Bilder in einem Gitterraster auf der ersten Seite der Aufgabe. Auf den nächsten Seiten der Aufgabe werden Positionsänderungen oder Bewegungen der Bilder relativ zum vorherigen Standort angezeigt. Der Proband soll sich an die geänderten Positionen erinnern und die Bilder zum Schluss entsprechend an die richtigen Positionen platzieren. In Aufgabe 3, soll der Proband die Zahlen, die in zufälliger Reihenfolge angezeigt werden, addieren oder subtrahieren und sich die Zwischenergebnisse merken, um zum Schluss anzugeben, wie viele Elemente eines Typs (z.B. verschiedene Süßigkeiten) vorliegen.

Die Ergebnisse dieser Studie bieten interessante Einblicke in die Möglichkeiten, mobile Techniken zur Unterstützung des ambulanten Assessment anzuwenden.

5 Fazit und Ausblick

Wir haben ein integriertes System aus Hardware und Software vorgestellt, um mobiles Assessment im Schulkontext zu ermöglichen. Besonderes Gewicht wurde auf die Benutzung offener Standards und auf Testsicherheit gelegt. Das System wurde in mehreren kleineren und einer größeren Erhebung eingesetzt. In letzterer wurden mehr als 120 Schüler über einen Zeitraum von mehreren Wochen mehrmals täglich getestet. Damit kann unser System für Zwecke eingesetzt werden, die weit über das mit konventionellen Testmethoden Mögliche hinausgeht. Psychometrisch wichtig ist die Erhöhung der ökologischen Validität der Testung durch die Integration in den Alltag der Schüler. Auf technischer Seite bedeutsam ist die Integration von Hardware und Software, die unser System auszeichnet. Die Kombination aus abgeschotteter Smartphone-Software und spezialisierter, abgestufter Hardware macht die langandauernden, unüberwachten Testzyklen erst möglich.

Für die Zukunft haben wir zahlreiche Ausbaupläne. Auf der Inhaltsseite wollen wir die Unterstützung für weitere Aufgabentypen ausbauen. Dabei werden wir an die Grenzen des QTI-Standards stoßen, so dass wir weitere Standards auf ihre Einsetzbarkeit hin analysieren wollen. Assessments sind ein wichtiger Bestandteil des Lernens. Unser System soll so ausgebaut werden, dass Lernaktivitäten besser unterstützt werden. Die Softwareentwicklung für Android ist prinzipiell geräteunabhängig. Bisher ist unser System in größerem Maßstab aber nur auf einem Smartphone-Typ eingesetzt worden, dem *Dell Streak 5*. Wir planen, unterschiedliche und neuere Smartphones einzusetzen – ein Unterfangen, das angesichts des Entwicklungstempos bei Mobilgeräten beständig fortgesetzt werden muss. Die Einbindung zusätzlicher Hardware wie eingebaute Kameras oder GPS-Systeme, die bei vielen Smartphones integriert ist, kann zu zusätzlichen Test- oder Kontrollmöglichkeiten führen.

Verwandte Arbeiten

Wichtige Teile unseres Systems basieren auf QTI XML. Wir haben eine Android QTI-Bibliothek in Java implementiert. Eine ähnliche Arbeit⁴ mit dem Namen Mobile QTI playr [ZWG10] wurde an der Universität Southampton durchgeführt. Es handelt sich um eine angepasste Version einer QTI engine⁵. Sie benutzt ein Web-Interface, um ein Assessment darzustellen. Dazu werden QTI XML-Dateien mittels XSL Transformation in zwei Schritten in XHTML-Dateien umgewandelt, die nachfolgend von einem spezialisierten *Renderer* in einem Browser dargestellt werden können. Auf jedem Mobilgerät wird daher ein spezieller Browser benötigt, i-jetty⁶.

Im Gegensatz dazu nutzt unsere QTI-Bibliothek XML-(De-)Serialisierung, um aus QTI-Dateien direkt Java-Objekte zu erzeugen und umgekehrt. Alle Aufgaben sind also direkt in Java implementiert und können als native „App“ auf jedem Android-Gerät ausgeführt werden. Der Grund für unsere Designentscheidung liegt hauptsächlich in der Testsicherheit. Native Applikationen erleichtern die Kontrolle über das Mobilgerät. Zusätzlich ist es wünschenswert, so wenige Prozesse wie möglich gleichzeitig auszuführen, um die Ressourcen für die eigentliche Testung reservieren zu können.

Literaturverzeichnis

- [BO05] Bowles, M.: The New Practices in Flexible Learning Framework Project (208TAS), <http://pre2005.flexiblelearning.net.au/projects/>, Zugriff 25.07.2012.
- [CH11] Chami, F.: Android XML Binding with Simple Framework Tutorial, <http://simple.sourceforge.net/articles.php>, Zugriff 03.02.2011.
- [DI11] DIPF: FLUX Project. <http://www.idea-frankfurt.eu/homepage/idea-projects/projekt-flux>.
- [GHJ94] Gamma, E.; Helm, R.; Johnson, R.; Vlissides, J.: Design Patterns, 1994
- [HNM05] Holzinger, A.; Nischelwitzer, A.; Meisenberger M.: Mobile phones as a challenge for m-learning: examples for mobile interactive learning objects (MILOs), In (IEEE), 2005.
- [IMS11] IMS Global Learning Consortium, Inc.: IMS Question & Test Interoperability Specification, <http://www.imsglobal.org/question/>, Zugriff 11.12.2011.
- [MO12] Moodle Trust: IMS QTI 2.0 format, http://docs.moodle.org/20/en/IMS_QTI_2.0_format, Zugriff 12.06.2012.
- [RL10] Rogers, R.; Lombardo, J.: Android Application Development, 26.05.2009; Section 10.1
- [RE12] Respondus Inc.: Question Import/Export Format: Respondus QTI Importer, <http://moodle.org/mod/data/view.php?id=13&rid=986>, Zugriff 11.06.2012.
- [REA12] Google's Android open source project: developer's API reference: <http://developer.android.com/reference/android/app/Activity.html>, Zugriff 21.06.2012.
- [REV12] Google's Android open source project: developer's API reference: <http://developer.android.com/reference/android/widget/ViewFlipper.html>, Zugriff 21.06.2012.
- [SIM12] Simple XML Serialization Inc.: <http://simple.sourceforge.net/>, Zugriff 20.06.2012.
- [ZWG10] Zhang, P.; Wills, G.; Gilbert, L.: IMS QTI Engine on Android to support Mobile Learning and Assessment. <http://eprints.soton.ac.uk/271485/>, 2010.

⁴ QTI mPlayer [BO05] ist ein anderes ähnliches Tool für die Geräte mit Windows mobile.

⁵ <http://wiki.qtitools.org/wiki/QTIEngine>

⁶ code.google.com/p/i-jetty/

Nutzung elektronischer Prüfungsfragesammlungen: Evaluation und Vergleich eines mobilen mit einem PC- basierten Trainingssystem

Jürgen Helmerich, Alexander Hörnlein, Marianus Ifland, Frank Puppe
Universität Würzburg, Am Hubland, 97074 Würzburg
helmerich@wiinf.uni-wuerzburg.de
hoernlein|ifland|puppe@informatik.uni-wuerzburg.de

Abstract: Zur Vorbereitung auf Prüfungen in der Medizin und anderen Fächern werden oft Sammlungen mit typischen Prüfungsfragen angeboten, die zur Prüfungsvorbereitung von Studierenden intensiv genutzt werden [HMI11]. Da Studierende viel Zeit in öffentlichen Verkehrsmitteln und anderen Wartesituationen verbringen müssen, ist ein mobiler Zugang zu Prüfungsfragesammlungen für Studierende attraktiv. Bei der Übertragung von elektronischen Fragesammlungen von Computern auf Smartphones sind neben dem reduzierten Bildschirm einige Randbedingungen zu beachten: Wenn die Fragesammlungen häufig aktualisiert werden, ist eine reine App-Anwendung mit einmaligem Download nicht möglich. Andererseits soll die Kommunikation zum Server minimiert werden, so dass nur zu Beginn die Fragen geladen und am Ende die Ergebnisse rückgemeldet werden. An der Universität Würzburg werden seit über 5 Jahren Fragesammlungen über das fallbasierte Trainingssystem CaseTrain [HHI09] angeboten. Wir haben entsprechend dem Wunsch vieler Studierender eine mobile webbasierte Version der Ablaufumgebung von CaseTrain für Smartphones mit reduzierter Funktionalität mit möglichst wenig Aufwand neu implementiert, die auf die meisten vorhandenen Fragesammlungen zugreifen kann. Die Erfahrungen mit über 50 Studierenden einer wirtschaftswissenschaftlichen Anfängervorlesung mit dem Einsatz dieser mobilen Lösung zeigen eine generell hohe Akzeptanz, aber auch Grenzen des Ansatzes wegen des umständlichen Aufrufs der Fragesammlungen über die Lernplattform Moodle und dem Laden der Fälle, so dass der konventionelle rechnergestützte Zugang bevorzugt wurde. Eine aufwändigere App-Lösung mit automatischer Web-Synchronisation der Inhalte und Rückmeldung der Ergebnisse ist ein vielversprechender Ansatz, diese Schwächen zu beheben und einen attraktiven mobilen Zugang zu Prüfungsfragesammlungen bereitzustellen.

1 Einleitung

Seit 2007 bietet das fallbasierte Trainingssystem CaseTrain [HHI09] Studierenden der Universität Würzburg in zahlreichen Fächern die Möglichkeit, Lerninhalte selbständig zu wiederholen und anhand herausfordernder Fragestellungen zu vertiefen. Dazu werden den Studierenden Problemstellungen am Rechner präsentiert, die sie schrittweise lösen können, wobei der Computer als virtueller Tutor nicht nur das Endergebnis, sondern auch alle Zwischenschritte überprüft und bewertet sowie im Bedarfsfall Hinweise und

Erläuterungen gibt. Seit seiner Einführung trifft diese Lösung auf hohe Akzeptanz bei den Studierenden mit inzwischen durchschnittlich mehr als 100.000 abgeschlossenen Fallbearbeitungen pro Semester. Vor allem Fragesammlungen, die der direkten Vorbereitung auf (Multiple Choice-)Prüfungen dienen, werden intensiv genutzt.

Um die Möglichkeiten der Nutzung von CaseTrain weiter auszuweiten, wurde auf Anregung von Studierenden zusätzlich zum etablierten PC-basierten Player eine Erweiterung für die einfache mobile Nutzung der Inhalte über Smartphones implementiert. Vor allem Prüfungsfragesammlungen, bei denen – anders als in klassischen Trainingsfällen – i.d.R. keine komplexen Szenarien präsentiert werden, die dann erst analysiert und Schritt für Schritt bearbeitet werden müssten, bieten sich für eine mobile Nutzung an, da diese gut auf die üblichen Längen von Wartezeiten und Pausen im studentischen Alltag (Warten auf den Bus, Zeiten zwischen Veranstaltungen etc.) ausgelegt werden können.

Mobile Tutorangebote finden zunehmend mehr Beachtung (z.B. [WPB11], [MTR11]), aber müssen sorgfältig auf ihren Nutzen gegenüber klassischen Ansätzen evaluiert werden. Im Folgenden soll die mobile Fallbearbeitung zunächst genauer skizziert werden. Anschließend werden wir Erkenntnisse, die aus der Evaluation und dem Vergleich des mobilen Trainingssystems mit dem konventionellen System gewonnen werden konnten, vorstellen und diskutieren.

2 Konzept

Für die Entwicklung des mobilen Players wurde explizit nicht das Ziel verfolgt, alle Funktionalitäten des bestehenden, auf Adobe Flash basierenden Players für die Bearbeitung am PC 1:1 auch auf eine Smartphone-Lösung zu übertragen; vielmehr sollten die besonderen Anforderungen und Beschränkungen von Smartphones oder Tablets mit kleinem Bildschirm bei der Konzeption der Lösung berücksichtigt werden. Dazu wurden speziell folgende Kriterien identifiziert:

- Die Benutzungsoberfläche muss auf die reduzierten Bildschirmgrößen von Smartphones ausgelegt sein und einen einfachen Dialog mit dem Anwender ermöglichen.
- Um die Kommunikation mit dem Server gering zu halten (vgl. hierzu auch [CH10, S. 3f.]), sollte die Anwendungslogik des Dialog weitgehend clientseitig ablaufen, d.h. das Smartphone lädt sich beim Start der Anwendung einen Trainingsfall bzw. eine Fragesammlung vollständig vom CaseTrain-Server und meldet erst beim Schließen des Players die Ergebnisse zurück an den Server.
- Um auch Fragesammlungen vollständig bearbeiten zu können, die für übliche mobile Nutzungsszenarien zu umfangreich sind, müssen sich Bearbeitungen pausieren und später an der gleichen Stelle wiederaufnehmen lassen.

Für die Umsetzung dieser Ziele kamen grundsätzlich zwei Ansätze in Frage: die Programmierung einer App für jede zu unterstützende Plattform (Android, iOS, Blackberry, Windows Phone etc.) oder die Entwicklung einer universell nutzbaren mobilen Web-Anwendung. Eine nach einmaligem Download und Installation rein offline ablaufende Anwendung war hierbei aufgrund der Rahmenbedingungen (laufende Aktualisierungen

von Fällen/Fragesammlungen, Rückmeldung der Ergebnisse an einen zentralen Server) keine Option. Aufgrund dessen und der einfacheren und schnelleren Realisierbarkeit des webbasierten Ansatzes gegenüber einer App-Lösung, die für jede Plattform gesondert angepasst werden müsste, fiel die Entscheidung zunächst auf die Umsetzung über eine mobile Web-Anwendung. Zwar muss hier bei jedem Start neben den eigentlichen Fragen auch der komplette Player mit geladen werden, was bei der realisierten Lösung mit einem Umfang von weniger als 200 kB jedoch vertretbar erschien.

Der mobile Player wurde daher als HTML5-Anwendung auf Basis von JavaScript/AJAX entwickelt. Hierbei werden beim Aufruf einer mobilen Fragesammlung zum einen der Player mit Anwendungslogik (definiert in JavaScript) und der komplette Content (Intro, Informationsabschnitte, Fragen mit Erklärungstexten, Abschlusskommentar) als XML-Dokument im gleichen Format wie beim konventionellen System geladen. Die Navigation durch die Fragen erfolgt dann rein offline ohne weiteren Serverzugriff: mittels AJAX werden die entsprechenden Elemente der XML-Struktur geladen und zur Anzeige gebracht. Sämtliche Benutzerinteraktionen (Navigationswege, ausgewählte Antworten etc.) werden von der Anwendung protokolliert und bei Abschluss bzw. Pausieren der Bearbeitung zurück an den Server übertragen. Ist der Internetzugang des Smartphones auch während der Bearbeitung aktiv, wird das Protokoll zusätzlich in regelmäßigen Intervallen übermittelt, um den Bearbeitungsstand auch bei unvorhergesehenen Abbrüchen zu sichern.

Im Vergleich mit dem Player für die PC-basierte Fallbearbeitung, der auf Basis von Adobe Flash entwickelt wurde, weist der mobile Fallplayer einige Beschränkungen im Funktionsumfang auf, die v.a. aufgrund der reduzierten Bildschirmgröße und zur Vermeidung überlanger Ladezeiten getroffen wurden. So werden von der mobilen Lösung keine Videos zur Inhaltspräsentation unterstützt. Auch komplexere Fragetypen (Mehrfacheingaben, Long-Menu-Fragen etc.), die eine größere Darstellungsfläche benötigen, wurden für die mobile Nutzung nicht umgesetzt.

Studierende greifen auf alle Fragesammlungen über die universitätsweite Moodle-basierte Lernplattform WueCampus zu, auf der diese neben Skripten, Präsentationsfolien oder Videoaufzeichnungen als Lernmaterialien für zahlreiche Lehrveranstaltungen zur Verfügung stehen. Aus einem WueCampus-Kursraum heraus lassen sich die Fragesammlungen sowohl für die rechnergestützte Bearbeitung als auch im mobilen Player öffnen.

Für die PC-basierte Nutzung wählen die Studierenden aus der Liste der Lernmaterialien den gewünschten Fall aus, worauf der Flash-Player in einem neuen Fenster startet. Beim Aufruf über ein Smartphone ist ein zusätzlicher Zwischenschritt erforderlich (vgl. Abb. 1): Im WueCampus-Kursraum (1) muss zunächst die „Fallliste für mobile Geräte“ (2) ausgewählt werden, die alle Trainingsfälle und Fragesammlungen eines Kurses enthält, die für eine mobile Nutzung geeignet sind. Aus dieser Liste heraus lässt sich anschließend der gewünschte Fall zur Bearbeitung starten (3 und 4). Zum schnelleren Erreichen der Fälle kann diese Liste auch in die Bookmarks des mobilen Browsers aufgenommen werden, was bei weiteren Zugriffen eine erneute Navigation über die Lernplattform erspart.



(1) Kursraum in WueCampus



(2) Mobile Fallliste



(3) Fallstart (Intro)



(4) Fallbearbeitung

Abbildung 1: Aufruf und Darstellung eines mobilen Falls im Smartphone (Bsp. iOS)

3 Ergebnisse

Zur Evaluation haben wir zum einen Log-Daten über die Fallbearbeitungen einschließlich einer Bewertung von Fallinhalt und Fallbedienung aufgezeichnet und weiterhin die Einschätzungen der Studierenden mit einem umfassenden Fragebogen zum Kurs erfasst.

Fall-Nummer	PC-basierte Fallbearbeitung							mobile Fallbearbeitung								
	# verschiedene Nutzer	# vollständige Bearbeitungen	% erfolgreiche Bearbeitungen	Bearbeitungszeit in Sek. (Median)	# Bewertungen	Bewertung Inhalt	Bewertung Bedienung	# verschiedene Nutzer	# vollständige Bearbeitungen	% erfolgreiche Bearbeitungen	Bearbeitungszeit in Sek. (Median)	# abgebrochene Bearbeitungen	Bearbeitungszeit in Sek. (Median)	# Bewertungen	Bewertung Inhalt	Bewertung Bedienung
1	318	463	69%	305	4	2,0	1,3	33	22	73%	416	23	161			
2	380	591	81%	347	10	1,5	1,1	29	17	82%	563	21	206			
3	336	507	67%	423	5	1,8	1,6	30	19	68%	395	23	375			
4	417	688	65%	469	12	1,7	1,4	30	15	60%	368	40	331			
5	338	559	53%	431	8	2,5	1,8	31	13	69%	426	27	408			
6	457	777	92%	311	25	1,6	1,5	46	25	100%	309	41	242	1	2,0	3,0
7	374	588	77%	288	8	2,0	1,6	33	17	76%	360	21	242			
8	533	1063	96%	341	65	1,5	1,3	73	48	94%	357	66	224	4	1,8	1,3
9	537	1258	71%	190	79	1,9	1,4	108	86	58%	235	107	113	8	1,6	1,3
10	478	1000	47%	372	20	3,4	1,7	45	21	52%	306	47	284	1	1,0	1,0
11	437	728	79%	275	13	2,2	1,3	28	15	67%	248	23	205	2	5,0	1,0
12	439	772	87%	419	18	1,8	1,9	36	20	90%	556	28	380	1	2,0	1,0
13	447	793	88%	380	15	1,8	1,7	36	19	68%	487	32	355			
14	411	733	45%	679	11	2,5	1,8	31	10	40%	927	42	479			
15	465	752	91%	298	23	1,6	1,3	24	14	79%	233	16	319			
16	381	606	73%	666	12	1,6	1,2	23	12	58%	846	18	517	1	2,0	3,0
17	456	893	63%	564	19	1,9	1,4	53	28	61%	701	46	450			
18	504	984	88%	455	42	1,6	1,4	58	30	87%	572	69	330	2	1,5	1,5
19	385	585	78%	398	9	1,3	1,3	28	18	78%	486	19	438			
20	359	497	80%	297	6	2,3	1,7	25	15	73%	223	22	199	1	1,0	1,0
21	330	466	95%	300	10	1,7	1,4	30	15	93%	541	27	315			
22	372	545	92%	309	10	2,0	1,7	27	15	80%	325	20	285	2	1,5	1,0
23	329	471	69%	467	11	1,2	1,1	32	20	90%	421	23	483			
24	340	520	83%	194	5	2,4	1,6	18	17	76%	201	8	198	1	2,0	2,0
25	299	395	94%	187	3	2,0	2,0	15	12	100%	276	8	16			
26	123	141	85%	534	18	1,6	1,1	17	5	100%	1269	15	263			
27	302	404	66%	463	3	2,0	1,7	15	12	58%	406	6	487			
28	358	492	95%	256	7	1,7	1,3	16	16	100%	211	4	227			
29	360	547	66%	214	5	3,2	1,3	17	15	73%	292	8	211	1	1,0	3,0
30	307	437	66%	344	2	3,0	2,0	17	13	69%	321	19	127			
31	349	521	60%	303	4	1,8	1,0	14	12	83%	275	6	313			
∅	385	638	76%	370	16	2,0	1,5	33	20	76%	437	28	296	2	1,9	1,7

Tab. 1: Vergleich von Kenndaten der konventionellen und mobilen Fallbearbeitung von 31 im Wintersemester 11/12 angebotenen Fällen zur Vorlesung „Einführung in die Wirtschaftsinformatik“

In Tab. 1 werden folgende Kenndaten pro Fall aufgelistet: Anzahl verschiedener Nutzer, die einen Fall bearbeitet haben (viele Nutzer bearbeiten einen Fall zwei Mal), Anzahl der vollständigen Fallbearbeitungen mit Erfolgsquote in Prozent, die durchschnittliche Bearbeitungszeit in Sekunden als Median (es wurde der Median statt dem Mittelwert gewählt, da er weniger empfindlich gegenüber Ausreißern ist, insbesondere Langläufern), die Anzahl der ausgefüllten Fallbewertungen und die Durchschnittsnote für Inhalt und Bedienung auf einer Schulnotenskala. Dabei werden die Kenndaten der PC-basierten Fallbearbeitung auf der linken Seite denen der mobilen Nutzung auf der rechten Seite gegenübergestellt. Bei der mobilen Nutzung gab es ein technisches Problem bei der Aufzeichnung: viele Studierende haben nach Bearbeitung aller Fragen nicht mehr den danach vorgesehenen Button gedrückt, so dass diese Fälle technisch als „abgebrochen“ gewertet sind. Daher sind im rechten Bereich von Tab. 1 auch abgebrochene Fälle aufgeführt und müssen teilweise zu den vollständig bearbeiteten Fällen hinzugezählt werden.

Die Daten in Tab. 1 zeigen, dass zwar 1/5 der Studierenden den Fall 9 auf ihrem Smartphone ausprobiert haben (108 im Vergleich zu 537), aber im Durchschnitt über alle 31 Fälle nur knapp 1/10 der Studierenden die mobile Fallbearbeitung genutzt haben (33 zu 385). Die Erfolgsquoten der Fallbearbeitung, die Bearbeitungszeit und die Bewertungen hinsichtlich Inhalt und Bedienung unterscheiden sich dagegen nicht auffällig zwischen PC-basierter und mobiler Fallbearbeitung.

Der umfassende Fragebogen mit 16 Fragen dient dazu, die Verbreitung von Smartphones unter den Studierenden und ihre Nutzungsgewohnheiten genauer zu erfassen, um ggf. auch Gründe und Verbesserungsmöglichkeiten für die mobile Bearbeitung von Fragestellungen herauszufinden. Er wurde von insgesamt 134 der über 500 Studierenden ausgefüllt, wovon 96 ein Smartphone hatten und ca. 50 damit auch Fälle bearbeitet haben. Die durchschnittlichen Ergebnisse werden soweit möglich mit den Ergebnissen eines analogen Fragebogens zur konventionellen rechnergestützten Nutzung derselben Trainingsfälle im gleichen Semester verglichen, der von 47 Personen ausgefüllt wurde (in runden Klammern). Die ersten fünf Fragen dienen zur Erfassung der Verbreitung und der Nutzungsgewohnheiten von Smartphones und die folgenden 10 Fragen beziehen sich auf die Fallnutzung (MW = Mittelwert; S = Standardabweichung; n = Anzahl der Studierenden, die die jeweilige Frage beantwortet haben). Hinzu kommen zwei Freitextfragen über qualitative Aussagen, deren Antworten aus Platzgründen nicht gezeigt werden:

Allgemeines:

1. Nutzen Sie ein Smartphone? [n=134]: Ja: 72%, Nein, Anschaffung innerhalb eines Jahres geplant: 9%, Nein, Anschaffung später geplant: 13%. Nein, keine Anschaffung geplant: 6%.
2. Welches Betriebssystem läuft auf Ihrem Smartphone [n=95]: Android: 38%, Apple iOS: 46%, Rest: 10%, weiß nicht: 5%.
3. Haben Sie einen Mobilfunkvertrag mit Daten-Flatrate / speziellem Datentarif [n=103]: Ja (bis 500 MB inkl.): 35%, Ja (500MB – 1 GB): 17%, Ja (> 1 GB): 18%, Nein: 29%.
4. Nutzen Sie Ihr Smartphone, um damit kurze Pausen (z.B. Warten auf den Bus, Pausen zwischen Veranstaltungen etc.) zu überbrücken? [n=98; 1 = regelmäßig, 5 = nie]: MW: 1,7; S: 1,1
5. Nutzen Sie unterwegs das Internet über Ihr Smartphone? [n=98; 1 = regelmäßig, 5 = nie]: MW: 2,0; S: 1,4

Nutzung mobiler Fälle (in Klammern Nutzung konventioneller Fälle):

6. Wie viele mobile CaseTrain-Fälle zur Veranstaltung haben Sie am Smartphone durchgearbeitet? [n=112]: *Alle: 2% (57%), mind. die Hälfte: 2% (32%); mehrere: 23% (11%), einen: 10% (0%), keine: 63% (0%)*

7. Wie greifen Sie auf die Fälle zu? *Nur über Mobilfunknetz: 14%, Nur über WLAN: 41%, über beide Netze: 45%*

Bedienung:

8. Der Aufruf der mobilen Fälle über WueCampus ist ... 1 = einfach, 5 = kompliziert [n=77]: Mittelwert: 2,1 (1,2); Standardabweichung: 1,2 (0,5)
9. Die Bedienung des mobilen Fallplayers ist ... (1 = einfach, 5 = kompliziert) [n=54]: MW: 1,9 (1,2); S: 1,0 (0,5)
10. Die Performance des mobilen Fallplayers (Lade/Reaktionszeiten) ist ... (1 = gut, 5 = schlecht) [n=53]: MW: 2,6 (1,9); S: 1,0 (1,0)
11. Die angebotenen Fälle sind für die mobile Nutzung ... (1 = geeignet, 5 = nicht geeignet): [n=52] MW: 2,0; S: 1,0
12. Die Präsentation der Fälle im mobilen Player ist ... (1 = ansprechend, 5 = nicht ansprechend) [n=52]: MW: 2,0 (2,1); S: 0,8 (0,9)

Persönlicher Nutzen:

13. Sollten mobile CaseTrain-Fälle zu weiteren Veranstaltungen angeboten werden? [n=104]: *ja: 95% (100%), nein: 5% (0%)*
14. Wären Sie an der Nutzung weiterer mobiler Lernangebote für Smartphones interessiert ... (1 = starkes Interesse, 5 = kein Interesse) [n=99]: MW: 2,2 S: 1,2

Gesamtbewertung:

15. Welche Schulnote würden Sie dem Angebot mobiler Trainingsfälle für Smartphones insgesamt geben (1 = sehr gut, 6 = ungenügend) [n=71]: MW: 2,1 (1,9) S: 0,6 (0,8)

Persönliche Angaben:

16. Geschlecht [n=124]: *männlich: 51% (69%), weiblich: 49% (31%)*

Es fällt auf, dass der Aufruf der Fälle über WueCampus von Smartphones aus als komplizierter empfunden wird als der bei konventionellen Rechnern (Frage 8; Differenz: fast 1 Punkt auf einer Skala von 1-5). Ähnliches gilt auch für die Bedienung (Frage 9). Die Performance des mobilen Fallplayers wird mit 2,6 auf einer Skala von 1-5 sogar als leicht unterdurchschnittlich bewertet (Frage 10). Dagegen wird die Präsentation der Fälle auf Smartphones als ähnlich gut wie auf Rechnern bewertet (2,0 zu 2,1). Die Studierenden haben ein großes Interesse, dass mobile Fälle auch zu weiteren Veranstaltungen angeboten werden (Frage 13: 95%) und geben dem Angebot mobiler Fälle insgesamt die Schulnote 2,1, was nur 0,2 Noten schlechter ist als bei dem Angebot der Fälle über konventionelle Rechner.

4 Diskussion und Ausblick

Die Ergebnisse zeigen, dass das Angebot mobiler Fallbearbeitungen von den Studierenden angenommen wurde, sie aber weiterhin die Fälle hauptsächlich mit konventionellen Computern bearbeiten. Die zwölfte Frage im Fragebogen zeigt, dass die Präsentation der Fälle im mobilen Player gut und ähnlich gut bewertet wird wie auf großen Bildschirmen, so dass die Darstellung wahrscheinlich nicht für die geringere Nutzung verantwortlich ist. Dagegen zeigen die Antworten auf die Fragen 8-10, dass wie oben ausgeführt Aufruf, Bedienung und Performance eine Ursache für die geringere Nutzung sein könnten. Dies zeigen auch die Freitextkommentare, die oft eine App wünschen.

Da eines der Ziele der Smartphone-Anwendung war, dass Studierende kurze Pausen im Tagesablauf durch Lernaktivitäten überbrücken können sollen, ist es wichtig, dass die

Studierenden ohne Verzögerungen das Tutorsystem starten können, was wegen des Aufrufs über WueCampus und des Ladens der Anwendung vom Server verlangsamt ist. Da die sonstigen Evaluationsergebnisse keine offensichtlichen Defizite des mobilen Tutor-systems erkennen lassen, scheint dies der Hauptgrund für die vergleichsweise geringe Nutzung der Trainingsfälle zu sein. Diese Schwäche zu beheben ist daher das vordringliche Ziel der weiteren Entwicklung. Dabei gibt es mehrere Möglichkeiten mit zunehmendem Aufwand und Nutzen: Am einfachsten ist die Umstellung auf die neue Moodle-Version 2.x, die eine optimierte Benutzungsoberfläche für Smartphones bietet. Ein größerer Nutzen ergäbe sich, wenn das Smartphone sich nur einmal gegenüber Moodle authentifiziert und bei weiteren Aufrufen sofort durchgeschaltet wird. Diese Option wird von Moodle angeboten, ist aber bei PC-basierten Anwendungen aus Sicherheitsgründen in WueCampus abgeschaltet, könnte jedoch für Smartphones bereitgestellt werden. Damit ließe sich die Anmeldung beschleunigen, nicht aber das Laden eines Falles. Um auch das zu erreichen, müsste man den Fallplayer als App mit minimaler Serverkommunikation neu programmieren (siehe auch [CH10, S. 12] und [TR06]): Die Studierenden laden damit einmalig (z.B. zu Beginn der Vorlesung) die Fallinhalte auf ihr Smartphone und können diese dann lokal als App starten. Dabei wird jedoch durch eine Serverabfrage kontrolliert, ob sich die Fallinhalte geändert haben. Bei kleineren Änderungen werden die Studierenden darauf aufmerksam gemacht und bei größeren Änderungen der Fallstart blockiert und ein erneutes Herunterladen erzwungen. Zusätzlich werden bei Serverkontakt im Hintergrund Protokoll Daten vom Smartphone auf den Server über die Fallbearbeitung zurückgespielt. Wir beabsichtigen, alle Möglichkeiten umzusetzen und zu evaluieren, die App-Option aus Aufwandsgründen aber zunächst nur für ein Smartphone-Betriebssystem. Bei entsprechendem Feedback sollen dann alle geeigneten CaseTrain-Fälle auch aus anderen Fakultäten für eine mobile Plattform freigeschaltet werden.

Literaturverzeichnis

- [CH10] Christmann, S.; Hagenhoff, S.: Softwarebasierte Lösungsansätze für mobilitätsbedingte Herausforderungen im mobilen Internet, Arbeitsberichte des Instituts für Wirtschaftsinformatik, Professur für Anwendungssysteme und E-Business, Universität Göttingen, Nr. 4, Göttingen, 2010.
- [HMI11] Hörnlein A.; Mandel, A.; Ifland, M.; Lüneberg, E.; Deckert, J.; Puppe, F.: Acceptance of medical training cases as supplement to lectures, GMS Z Med Ausbild 2011; 28(3).
- [HHI09] Helmerich, J.; Hörnlein, A.; Ifland, M.: CaseTrain – Konzeption und Einsatz eines universitätsweiten fallbasierten Trainingssystems. In: Apostolopoulos, N.; Hoffmann, H.; Mansmann, V.; Schwill, A. (Hrsg.): E-Learning 2009 - Lernen im digitalen Zeitalter, Medien in der Wissenschaft 51, Waxmann-Verlag, 2009, S. 173-184.
- [MTR11] Möller, A.; Thielsch, A.; Roalter, L.; Kranz, M.: MobiDics - Eine mobile Didaktik-Toolbox für die universitäre Lehre. In: Rohland, H.; Kienle, A.; Friedrich, S. (Hrsg.): DeLFI 2011 – Die 9. e-Learning Fachtagung Informatik der GI e.V., Dresden 2011, S. 139-150.
- [TR06] Trifonova, A.; Ronchetti, M.: Hoarding content for mobile learning, Int. J. Mobile Communications, Vol. 4, No. 4, 2006, S. 459–476.
- [WPB11] Wegener, R.; Prinz, A.; Bitzer, P.; Leimeister, J.: Steigerung von Interaktivität, Individualität und Lernzufriedenheiten in einer universitären Massenveranstaltung durch mobile Endgeräte. In: Rohland, H.; Kienle, A.; Friedrich, S. (Hrsg.): DeLFI 2011 – Die 9. e-Learning Fachtagung Informatik der GI., Dresden 2011, S. 173-184.

Click2Vote - Ein servicebasiertes Clicker-Framework zur Teilnehmeraktivierung in Lehrveranstaltungen

Tom Tschernack, Raphael Zender, Ulrike Lucke

Institut für Informatik
Universität Potsdam
August-Bebel-Str. 89
14482 Potsdam

vorname.nachname@uni-potsdam.de

Abstract: *Personal Response Systeme* (PRS) bzw. *Clicker* sind heute ein anerkanntes Mittel, um während einer Lehrveranstaltung Live-Feedbacks von Studierenden zu erhalten. Das Spektrum der Clicker-Lösungen reicht von Hardware-Systemen, über Software-Tools bis hin zu webbasierten Anwendungen für unterschiedliche Client-Klassen. Allerdings handelt es sich bei diesen Systemen um in sich abgeschlossene Lösungen mit proprietären Schnittstellen. Dadurch ist eine Integration in größere, beispielsweise universitäre IT-Infrastrukturen nur eng begrenzt mit vertretbarem Aufwand möglich. Dieser Artikel beschreibt ein offenes und flexibles Clicker-Framework, das über die Nutzung standardisierter Schnittstellen ein hohes Integrationspotential aufweist. Dadurch wird nicht nur der Einsatz mobiler Technologien für die klassische Präsenzlehre gefördert sondern auch eine technische Mobilität durch die Wahl beliebiger Hardware-Systeme gewährleistet. Die unterschiedlichen Nutzungs- und Integrationsansätze werden beispielhaft vorgestellt und anhand konkreter Anwendungsszenarien diskutiert.

1 Motivation

Moderne Lehrveranstaltungen zeichnen sich unter anderem durch ein hohes Maß an Interaktion zwischen Lehrenden und Lernenden aus. Dazu gehört auch die Erfassung und Berücksichtigung von Meinungen, Ansichten und Wissenständen der Lernenden. Sogenannte *Personal Response Systeme* (PRS) bzw. *Clicker* für die einzelnen Teilnehmer ermöglichen Abstimmungen zu bestimmten Fragestellungen und bieten verschiedene Optionen zur Verarbeitung des Resultates.

Die Funktionsweise von Clickern ist vergleichbar mit Publikumsabstimmungen in TV-Quizzes wie z. B. "Wer wird Millionär?". Der Lehrende stellt während einer Lehrveranstaltung Multiple-Choice-Fragen mit einer vorgegebenen Menge möglicher Antworten. Die Lernenden stimmen dann über Clicker-Clients für jeweils eine Antwort und der Lehrende erhält das über alle Abstimmenden aggregierte Ergebnis für den persönlichen Gebrauch oder die Auswertung im Plenum.

Clicker-Systeme helfen Dozierenden bei der aktiven Einbindung von Studierenden während einer Unterrichtseinheit und der Abschätzung des inhaltlichen Verständnisses. Zudem eröffnen sie Möglichkeiten zur sofortigen Reaktion auf offene Fragen [Mar07]. Da einzelne Teilnehmer in der Menge der Stimmen untergehen, ist außerdem eine gewisse Anonymität gewährleistet, die eine freie Meinungsäußerung fördert und authentische Ergebnisse erwarten lässt. Im Vergleich zu anderen "Backchannel"-Systemen bestechen Clicker durch ein klare Einsatzszenarien und eine intuitive Nutzung auch durch gering technikaffine Anwender. Positive Studien über den Lernerfolg durch die Verwendung von Clicker-Systemen liegen bereits mehrfach vor [OK11]. Zudem ist allein durch den Wechsel des pädagogischen Fokusses vom passiven zum aktiven Lernen von einem Lerngewinn auszugehen, unabhängig von einer konkreten Methode oder Technologie [Wes05].

Die auf dem Markt verfügbaren Clicker-Systeme lassen sich in drei Klassen unterteilen, die im folgenden Abschnitt vorgestellt werden. Es handelt sich jedoch ausnahmslos um geschlossene Systeme, die nur sehr umständlich erweiterbar sind. In Abschnitt 3 wird daher ein System vorgeschlagen, das die Clicker-Funktionalität in einem systematisch erweiterbaren und auf offenen, standardsisierten Schnittstellen basierenden Framework umsetzt. Der Mehrwert dieses Ansatzes wird durch eine Referenzimplementierung und in Abschnitt 4 anhand konkreter Anwendungsszenarien gezeigt. Abschließend werden dieser Beitrag zusammengefasst und zukünftige Erweiterungen betrachtet.

2 Clicker-Lösungen: Bestandaufnahme

Clicker-Systeme sind bereits seit den 60er Jahren im Einsatz und haben neben dem in diesem Artikel fokussierten Klassenraumszenario auch Realisierungsmöglichkeiten des interaktiven Fernsehens eröffnet. Während letztere Systeme durch das Internet fast völlig verdrängt wurden, haben Clicker zur Lerneraktivierung eine Evolution durchlaufen. Im Folgenden werden die drei wichtigsten Clicker-Arten sowie eine Auswahl typischer Vertreter aus dem Spektrum nahezu unzähliger Systeme betrachtet.

Die ältesten und verbreitetsten Systeme sind **Hardware-Clicker** aus mehreren dedizierten Sendern und typischerweise einem einzelnen Empfänger. Beide Gerätetypen kommunizieren drahtlos über Infrarot oder Funktechnologien miteinander. Während die Sender an die Zuhörer verteilt werden, nutzt der Vortragende in der Regel den Empfänger zur Stimmerfassung und eine mitgelieferte Software zur Auswertung. Abbildung 2 zeigt beispielhaft eine aktuelle Marktauswahl. Einzelne Systeme sollen aufgrund der hohen konzeptuellen Ähnlichkeit an dieser Stelle nicht hervorgehoben werden. Preislich liegen derartige Clicker-Clients zwischen 20 und 60 USD pro Gerät [OK11].

Hardware-Clicker überzeugen als aufeinander abgestimmte Komponenten durch eine zuverlässige Funktionsweise. Allerdings bieten Sie oftmals nur wenige Abstimmungsformate, wie etwa Multiple-Choice an. Bildanzeigen und Textantworten sind meist nicht möglich. Weiterhin ist ihr breiter Einsatz mit hohen Investitionskosten verbunden und die Wahl eines konkreten Systems schließt typischerweise alternative Hardware-Clicker aufgrund mangelnder Interoperabilität aus.



Abbildung 1: Hardware-Clicker ¹

Software-Clicker skalieren bei verschiedenen Teilnehmerzahlen deutlich besser hinsichtlich der Investitionskosten. Sie erfordern die Installation einer speziellen Software auf Notebooks, PCs oder auch Smartphones der Teilnehmer sowie dem Rechner des Dozenten. Die Kommunikation erfolgt über lokale Netzwerke bzw. das Internet.

Einige Systeme, wie *Informa* [HA09] und das *TVremote System* [BRTL06] haben neben einem typischen Rechner weitere Systemanforderungen (z. B. Java). Im Gegenzug bieten sie teilweise erweiterbare Frameworks und ein breites Spektrum von Frage/Antwort-Formaten. Neben Multiple-Choice-Antworten sind beispielsweise textuelle und grafische Antworten sowie individualisierte Nachrichten zwischen Sender und Empfänger möglich. Vermehrt bieten Software-Clicker wie *eduVote* neben der PC-Software auch Smartphone-Apps sowie Web-Interfaces über den Browser an [Ley12]. Dadurch eröffnen sie sich einem breiteren Nutzerspektrum.

Die Klasse der **Web-Clicker** setzt ausschließlich auf Abstimmung, Auswertung und Verwaltung über den Browser. Dabei kommen auch Web 2.0-Technologien und -Plattformen zum Einsatz. Die neben *eduVote* in Abbildung 2 dargestellte Lösung *PowerPoint Twitter Tools* von SAP setzt beispielweise auf die Stimmabgabe über Twitter. Ein SAP-Server sammelt diese Stimmen, aggregiert sie und bietet das Resultat über ein Flash-Interface an. Dieses kann dann unter anderem unmittelbar in PowerPoint-Folien integriert werden und findet so live den Weg in den Seminarraum. Leider benötigen die Teilnehmer einen Twitter-Account, wobei die Abgabe mehrerer Stimmen über einen Account ebenfalls möglich ist.

Die Entscheidung für eines dieser Systeme ist in den meisten Fällen indirekt verbindlich. Hardware-Clicker arbeiten beispielsweise nicht mit Software- oder Web-Clickern zusammen, so dass jeder Teilnehmer einen Abstimmungs-Client erhalten muss obwohl er heute in der Regel mit PC bzw. Smartphone ausgerüstet ist und diese persönlichen Geräte direkt nutzen könnte. Im folgenden wird ein Framework vorgeschlagen, das dem Nutzer eine freie Client-Wahl ermöglicht, indem es die Clicker-Kernlogik hinter Web Services kapselt die standardisiert über ein Netzwerk ansprechbar sind.

¹Bildquellen: <http://aclasstechnology.com>, <http://edtechtools.wordpress.com>, <http://en.wikipedia.org>

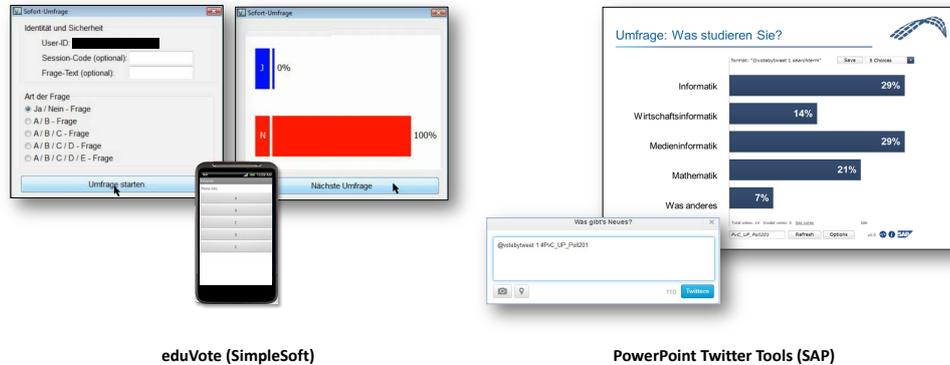


Abbildung 2: Software-Clicker von SimpleSoft und Web-Clicker von SAP

3 Click2Vote

Grundidee des Frameworks *Click2Vote* ist die standardisierte Trennung von Nutzungsschnittstellen und eigentlicher Kernlogik. Die Web Service-basierten Schnittstellen können dabei durch beliebige Hardware-, Software- oder Web-Clicker-Systeme genutzt werden.

3.1 Funktionen

Für den Lehrenden und die Lernenden unterscheidet sich die *Click2Vote*-Funktionalität kaum von der traditioneller Clicker. Unterstützte Funktionen sind unter anderem:

- Erstellung, Bearbeitung und Löschen von Umfragen
- Aktivierung und Archivierung von Umfragen
- Single-Choice-Umfragen (z. B. Ja/Nein)
- Multiple-Choice-Umfragen (z. B. A/B/C/D/E)
- Abgabe von Kommentaren zusätzlich zur Stimmabgabe (z. B. für demographische Angaben oder freie Kommentare)
- Anonymisierung der Abstimmenden
- Stimmenaggregation für den Dozenten

Sämtliche Funktionen können über entsprechende Web Services von beliebigen Clicker-Clients genutzt werden. Zur einfachen Nutzung stützt bereits Web-Anwendungen, die diese standardisierten Schnittstellen bedienen, zur Verfügung.

3.2 Architektur und Web Services

Die zentrale Herausforderung bestand darin, ein breites Client-Spektrum so zu integrieren dass diese trotz eigenständiger Ausprägung dieselbe funktionale Logik teilen. Nur so können beispielsweise Hardware- und Web-Clicker gleichberechtigt an derselben Umfrage teilnehmen. Die dafür erforderliche gemeinsame Datenbasis wurde zentralisiert gehalten, um den Funktionsumfang der Client-Anbindungen und somit den individuellen Entwicklungsaufwand zu reduzieren. Um die einzelnen Clients an die zentrale Datenbasis anzubinden wurde zugunsten standardisierter Web Services von einem proprietären Plugin-Ansatz abgesehen. Da Web Services von der Masse der im Web relevanten Programmiersprachen unterstützt wird, ist eine Nutzung der Datenbasis mit beliebigen Programmiersprachen und von jedem relevanten Betriebssystem aus möglich. Selbst proprietäre Hardware-Clicker könnten über systemspezifische Anwendungen auf dem Dozentenrechner Stimmen an das *Click2Vote*-System weiterleiten.

Die Stärke von *Click2Vote* liegt somit im Vergleich zu anderen Lösungen in den über SOAP standardisierten, offenen Web Services, über die alle Funktionen des Frameworks direkt nutzbar sind. Der Systemkern besteht aus einer typischen Java-Anwendung, die durch das JAX-WS-Framework Services zur Abstimmungsverwaltung, sowie Stimmenabgabe und Ergebnisabfrage anbietet. Die drei derzeit existierenden Services sind in leicht reduzierter Form in Abbildung 3.2 im Rahmen der Grobarchitektur dargestellt.

Der schlanke Kern des Frameworks erfordert eine Java-VM und könnte beispielsweise im universitären Umfeld auf einem zentralisierten Server des Rechenzentrums betrieben werden. Die Bedienung der drei *Click2Vote*-Services erfolgt auf hingegen auf Grundlage des SOAP-Protokolls klassischer Web Services. Der **Admin-Service** enthält dabei alle Methoden zur Verwaltung von Dozenten (Teacher) und Umfragen (Polls). Der deutlich kompaktere **Voting-Service** wird genutzt um Umfragen aus dem Systemkern zu laden und über die angebotenen Optionen abzustimmen (Vote). Zur Abfrage des Abstimmungsergebnisses nutzen Dozenten den **Result-Service**. Im Folgenden wird die Referenzimplementierung passender Service-Consumer in Form von Web-Interfaces betrachtet und dabei das Potential der einzelnen Services verdeutlicht.

3.3 Consumer-Referenzimplementierung

Die in Abbildung 3.2 dargestellten HTML-Seiten werden über einen Jetty-Webserver unter Nutzung des Google Web Toolkits (GWT) erzeugt. Sie implementieren intern die drei vorgestellten Web Services und ermöglichen eine Web-basierte Verwaltung, Abstimmung und Auswertung und dienen Entwicklern weiterer Clients als Referenzimplementierung.

Über die Abstimmungsverwaltung erstellen, bearbeiten, aktivieren, archivieren und löschen registrierte Dozenten ihre Umfragen. Sie geben dafür eine zur Umfrage gehörende Frage sowie die dazu passenden Antworten (derzeit noch beschränkt auf Multiple/Single Choice) an. Die Umfragen können vor der Lehrveranstaltung vorbereitet und dann zu Umfragezeitpunkt aktiviert werden. Dabei erhält jede aktive Umfrage eine Poll-ID zur Identifikation

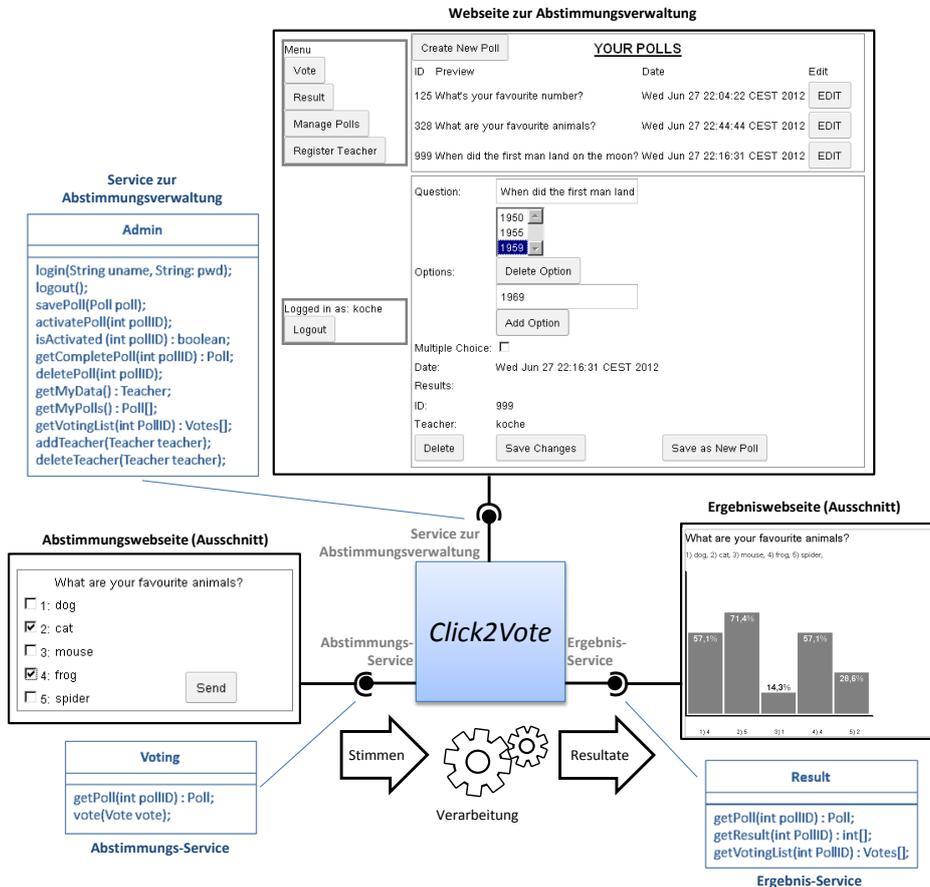


Abbildung 3: Architekturüberblick: Drei Web Services zur Verwaltung, Stimmabgabe und Auswertung und die Referenzimplementierung ihrer Nutzung in Form von Web-Interfaces

und Weitergabe an die Abstimmenden. Diese gaben die erhaltene ID ohne weitere Anmeldung auf der *Click2Vote*-Webseite an und werden zur Stimmabgabe weitergeleitet. Vor und während der Umfrage kann sich der Dozent die Zwischen- und Endergebnisse anzeigen lassen und ggf. seinen Zuhörern präsentieren.

Da sich HTML-Seiten in vielen Fällen bereits direkt in Medien wie Powerpoint-Folien und Lernplattformen einbetten lassen, ist schon jetzt eine breite Verteilung der Resultatswebseite möglich. Insbesondere aktivere Prozessschritte wie die Abstimmung und Verwaltung sollten jedoch aufgrund unerwünschter Wechselwirkungen zwischen Medium und HTML-Seite nicht über rudimentär eingebettete Seiten erfolgen. Erst die noch ausstehende Implementierung weiterer Service-Consumer eröffnet sich ein breites Spektrum an Anwendungsszenarien für *Click2Vote*.

4 Anwendungsszenarien

Durch die offenen Service-Schnittstellen eröffnet sich ein breites Spektrum an Integrationsmöglichkeiten von verbreiteten Smartphones über Lernmanagementsysteme bis hin zu Videokonferenzvorlesungen und virtuellen Online-Lernwelten. Die im folgenden betrachteten Szenarien bieten daher nur einen beschränkten Einblick in das eröffnete Potenzial.

In dem klassischen Anwendungsszenario einer durch Clicker bereicherten Präsenzveranstaltung bereitet der Dozent zunächst eine Umfrage zur einem passenden Thema vor. Während der Veranstaltung aktiviert er die Umfrage. Dazu müsste ein *Click2Vote*-Client genutzt werden, der die Verwaltung von Umfragen unterstützt. Die Studierenden erhalten die Poll-ID und geben diese auf einem beliebigen Voting-Client an, bekommen die Optionen angezeigt und geben daraufhin ihre Stimme ab. Abschließend deaktiviert der Dozent die Umfrage und könnte die aggregierten Stimmen z. B. in einer Powerpoint Folie darstellen, indem diese über ein Plugin den Result-Service abfragt.

Neben dieser synchronen Form der Clicker-Nutzung sind auch asynchrone Umfragen denkbar. Derzeit werden beispielsweise Moodle-Plugins für die drei *Click2Vote*-Services entwickelt. Der Dozent einer Vorlesung kann darüber direkt in der Lernplattform eine Umfrage erstellen, die z. B. während einer gesamten Woche ebenfalls im Moodle oder auch direkt von Studierenden-Smartphones aus beantwortet werden könnten. Die Auswertung kann entweder in einer anschließenden Vorlesung oder ebenfalls über Moodle erfolgen und so in eine Reihe von Online-Lernszenarien eingebettet werden.

Ein drittes Anwendungsszenario ergibt sich in Videokonferenz-Vorlesungen, die ein lokales und entferntes Publikum kombinieren. Sowohl lokale als auch live teilnehmende entfernte Teilnehmer können gleichberechtigt, mit beliebigen Endgeräten an Umfragen teilnehmen wodurch die meist ungewollte Trennung beider Zuhörerschaften aufgehoben wird und eine intensivere Kollaboration möglich ist. Dieses Szenario erlaubt auch Untersuchungen bzgl. der Nutzbarkeit durch Großgruppen.

Neben diesen Szenarien können durch die flexible Client-Auswahl weitere Anwendungsbereiche wie virtuelle Lernwelten, Blended Learning-Ansätze, Field Trips und kollaborative Projektarbeiten durch Clicker-Funktionalität bereichert werden.

5 Zusammenfassung und Ausblick

Clicker finden zunehmend Akzeptanz als nützliche Werkzeuge für Lerneraktivierungen in Präsenzveranstaltungen. Sie fördern aktive und kollaborative Lernprozesse und erhöhen die Lerneffizienz [OK11]. Heute gibt es eine Vielfalt verschiedener Clicker-Systeme mit unterschiedlichem Funktionsumfang, Systemvoraussetzungen und entstehenden Anschaffungskosten. Die Wahl eines konkreten Systems schließt andere Systeme in der Regel aus.

Die in diesem Beitrag vorgestellte Software-Lösung *Click2Vote* verfolgt die Philosophie einer systematischen Integration verschiedener Clicker-Ansätze über offene, standardisierte Web Services. Während die Kernlogik mitsamt der gemeinsamen Datenbasis zentral auf

einem Webserver betrieben wird, wurden die Voraussetzungen geschaffen, die Funktionen zur Abstimmung, Auswertung und Verwaltung in nahezu beliebige Systeme zu integrieren und für diese zu individualisieren. Als Referenzimplementierung wurden Web-basiertes Client-System implementiert. Darüber hinaus ist beispielsweise auch die Einbettung in Lernmanagementsysteme, Smartphone-Plattformen, Powerpoint-Folien oder sogar virtuellen Welten möglich. Die technologische und räumliche Mobilität der Studierenden bei der Nutzung derartiger Abstimmungen wird somit gefördert.

Während der Mehrwert des Clicker-Konzeptes für die Lehre bereits mehrfach untersucht wurde, steht eine Evaluierung der Bedienbarkeit des konkreten *Click2Vote*-Systems noch aus. Neben der Evaluierung in authentischen Lehrveranstaltungen mit variabler Teilnehmeranzahl im Wintersemester 2012/13 wird auch der Einsatz in verschiedenen Plattformen der universitären Aus- und Weiterbildung (z. B. Moodle, Second Life) analysiert.

Weiterentwicklungen der Lösung fokussieren vor allem die Entwicklung neuer *Click2Vote-Clients*. Dabei stehen konkrete Lernplattformen, virtuelle Klassenzimmer und Lernwelten sowie Web 2.0-Dienste im Vordergrund. Auch die Entwicklung weiterer Umfrageformate wie beispielsweise zur Abgabe von Fotos wird derzeit untersucht.

Nach erfolgreicher Evaluierung an der Universität Potsdam wird *Click2Vote* anderen Bildungseinrichtungen zur Verfügung gestellt, um die Systemverbesserung und den Client-Ausbau auf Grundlage einer breiten Nutzerbasis zu fördern.

Literatur

- [BRTL06] H. Baer, G. Roessling, E. Tews und E. Lecher. Bluetooth Interaction Support in Lectures. In *Proceedings of the IADIS International Conference Mobile Learning*, Seiten 360–364. IADIS, 2006.
- [HA09] Matthias Hauswirth und Andrea Adamoli. Solve & evaluate with informa: a Java-based classroom response system for teaching Java. In *Proceedings of the 7th International Conference on Principles and Practice of Programming in Java*, Seiten 1–10. ACM, 2009.
- [Ley12] Leydecker, F. and Diercks-O'Brien, G. and Frohwein, S. Einsatz von EduVote in der Lehrveranstaltung 'Mathematik I für Studierende der Wirtschaftswissenschaften' (Erfahrungsbericht). Bericht, Universität Hannover, 2012.
- [Mar07] M. Martyn. Clickers in the Classroom: An Active Learning Approach. *EDUCAUSE Quarterly*, 30(2):71–74, 2007.
- [OK11] James Oigara und Jared Keengwe. Students' perceptions of clickers as an instructional tool to promote active learning. *Education and Information Technologies*, Seiten 1–14, 2011.
- [Wes05] J. West. Learning Outcomes Related to the Use of Personal Response Systems in Large Science Courses. Academic Commons, 2005.

Studi.UP: Mobile Mehrwert-Dienste für konventionelle Legacy-Applikationen des Campus Management

Jan Bernoth¹, Alexander Kortsch¹, Volker Kaatz²,
Sander Lass¹, Raphael Zender², Ulrike Lucke²

Universität Potsdam

¹ Lehrstuhl Wirtschaftsinformatik

² Lehrstuhl Komplexe Multimediale Anwendungsarchitekturen

A.-Bebel-Str. 89, 14482 Potsdam

vorname.nachname@uni-potsdam.de

Abstract: Studierenden stehen zahlreiche Websites zur Verfügung, die die Organisation des Studiums erleichtern sollen. Die verbreitete Nutzung von mobilen Geräten erlaubt einen schnellen und ortsunabhängigen Zugang, jedoch sind viele dieser Sites nicht auf die speziellen Belange (Darstellungsform, Interaktionsmuster, Ressourcen, Sicherheit) der mobilen Plattformen abgestimmt oder nutzen deren Möglichkeiten nur unzureichend (Ortsbezug, persönliche Profile etc.). Die Integration derartiger mobiler Dienste unter einer modularen Oberfläche mit Single-Sign-On steigert die Usability und damit die Akzeptanz innerhalb der Zielgruppe. Herausforderung dabei ist es, den Zugriff auf alle Backend-Systeme zu realisieren, zumal nur wenige existierende Systeme technisch oder organisatorisch bedingt eine geeignete Schnittstelle zum Datenabruf bereitstellen. Der vorliegende Beitrag zeigt eine flexible Lösung – unabhängig von vorhandenen Schnittstellen, leicht wartbar und modular aufgebaut sowie unter Berücksichtigung der zugehörigen Sicherheitsaspekte.

1 Motivation

Moderne Applikationen der Hochschule bieten inzwischen einen mobilen Zugang über das Web oder über dedizierte Apps. Für ältere Systeme sind derartige Mechanismen jedoch i.d.R. noch nicht vorgesehen und auch nicht immer realisierbar (z.B. Probleme mit Anzeige klassischer Web-GUIs auf Smartphones). Beispielhaft seien Produkte aus den Bereichen Campus Management Systeme (HIS-GX), Raumplanung (TimeEdit), Bibliothek (OPAC) oder Studentenwerk/Mensa (häufig statische Websites) genannt.

Für mobile Dienste ergibt sich zudem ein Mehrwert aus der Orchestrierung mehrerer Datenquellen bzw. Anwendungen (sog. multi-sourced environment [CY05]). Darüber hinaus werden häufig ortsbezogene oder personalisierte Zugänge angeboten. Dies ist teilweise vergleichbar mit verschiedenen Ansätzen zur Bereitstellung von Campus-Portalen, die verschiedene Systeme der Hochschule koppeln.

Sollte die Strategie einer Hochschule, die ihren Mitarbeitern und Studierenden mobile Campus-Anwendungen zur Verfügung stellen will, also lauten: abwarten bis auch die konventionellen Lösungen mobile Zugänge anbieten? Angesichts von Projektvolumina im siebenstelligen Euro-Bereich und Projektlaufzeiten von mehreren Jahren allein im Bereich Campus Management kann ein einfach zu entwickelndes Provisorium sinnvoll sein. Es erlaubt zudem tiefere Einblicke in die Architektur, Schnittstellen und Handhabbarkeit vorhandener Systeme, die bei einer strategischen Erneuerung oder Erweiterung der IT-Infrastruktur [BB10] nützlich sein werden.

Der Beitrag beschreibt daher ein Konzept zur Bereitstellung von Mehrwert-Diensten für mobile Geräte/Benutzer, das auf verschiedene Legacy-Applikationen und -Dienste zurückgreift, sowie dessen beispielhafte Implementierung Studi.UP für Studierende an der Universität Potsdam.

2 Anforderungsanalyse

2.1 Nutzerseitige Anforderungen

Um den Mehrwert einer mobilen Anwendung zu gewährleisten, sollten durch verschiedene Schnittstellen zu vorhandenen IT-Systemen auf dem Campus folgende Anwendungsfälle [BGB11] abgedeckt werden:

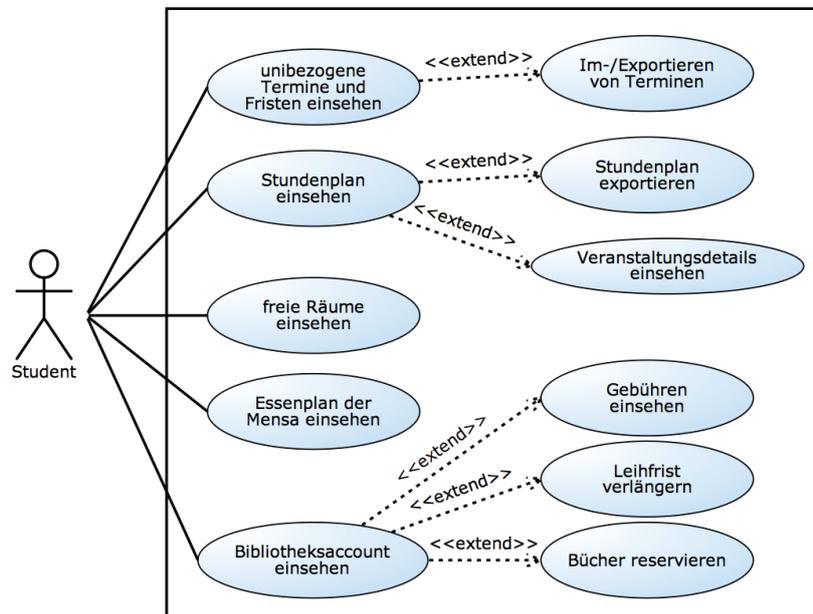


Abbildung 1: Anwendungsfälle der mobilen Campus-App im Use-Case-Diagramm

Laut einer Umfrage unter den Studierenden verschiedener Fachrichtungen und Hochschulen wird in der Benutzung besonderen Wert auf Veranstaltungsmanagement, Prüfungsmanagement und Informationsfunktionen gelegt [BGB11]. Somit sollten gerade in einer prototypischen Entwicklung die Implementierung vom Stundenplan und Zugriffsmöglichkeiten auf unibezogene Termine und Fristen in den Vordergrund rücken. Alle Funktionen müssen schnell zugreifbar sein.

Für eine breite Nutzung ist es wichtig, die App auf verschiedenen Plattformen (iOS, Android, etc.) anzubieten. Für eine gute Akzeptanz müssen Vorteile und das Nutzerempfinden der jeweiligen Plattform ausgenutzt werden. Ergänzend ist es hilfreich, die gesammelten und visualisierten Informationen auch für weitere Plattformen (z.B. als Portlet für modular zusammenstellbare, personalisierte Lernumgebungen [SK09]) exportierbar zu gestalten.

2.2 Systemseitige Anforderungen

Die zu schaffende Lösung muss sich nahtlos in bestehende Infrastrukturen einfügen. Neben einer Beachtung des Corporate Design (das betrifft vorwiegend Farben, Logos und Interaktionsmuster) sind damit auch technische Rahmenbedingungen gemeint.

Für die hier beschriebene Lösung sind die erforderlichen Daten über mehrere bestehende Systeme verteilt. Dazu zählen:

- Campus Management: das Potsdamer Universitätssystem für Lehre und Studium (PULS), eine umfangreiche Erweiterung der HIS-GX-Produkte
- Raummanagement: TimeEdit, eine aus dem Hotelgewerbe stammende Software, die eine überschneidungsfreie Stunden- und Raumplanung unter Beachtung von Studienordnungen und Prüfungsverknüpfungen vornimmt
- das Bibliotheksportal auf OPAC-Basis
- die Mensa-Informationsseiten des Studentenwerks

Diese Systeme bieten eine Reihe von unterschiedlichen Schnittstellen, die durch mobile Mehrwertdienste ansprechbar sind. Dazu zählen in beschränktem Umfang reguläre Programmierschnittstellen (APIs), aber auch unmittelbare Zugriffe auf Datenbanken (aus Konsistenzgründen nur lesend und generell nur wenn nötig) sowie die herkömmlichen Webseiten dieser Plattformen (die notfalls geparkt werden können). Bei allen Entwicklungen sind selbstverständlich bestehende Vorgaben der Systemsicherheit bzw. des Datenschutzes zu beachten.

3 Beschreibung der entwickelten Lösung Studi.UP

3.1 Systemarchitektur

Die Studi.UP ist in ihrer Architektur in drei Teile gegliedert und entspricht damit der gängigen Praxis [Su07][Ng10]. Der Viewer stellt das User Interface für die Benutzer dar, verarbeitet die Anfragen und leitet sie zum Server weiter. Der Server empfängt und verarbeitet die Anfragen. Als letzte Komponente in der Wertschöpfungskette stehen die Legacy-Anwendungen (bspw. das Campus Management System), aus denen die Daten ausgelesen werden.

Der Viewer bietet dem Nutzer mehrere modulare, abgekapselte Services an, die verschiedene Daten aus dem System widerspiegeln. Zum Beispiel können die Services den persönlichen Stundenplan abrufen, den aktuellen Mensaplan einsehen, Zugang zum Bibliotheksportal herstellen oder freie Räume in der näheren Umgebung anzeigen. Lokal werden die Daten dann persistent gespeichert und bei Änderungen aktualisiert. Auch werden Benutzerdaten sicher im Key Storage des Endgerätes verschlüsselt und gespeichert. Sobald der Service gestartet wird, reicht er Anfragen des Nutzers über die Sicherheitsschicht weiter, die mittels aktueller Standards die Netzwerktransaktion mit den verschiedenen Legacy-Anwendungen abwickelt.

Der Server bekommt die Anfrage aus der Sicherheitsschicht und kann diese dann dem passenden Plugin zuordnen. Je nach vorhandenen Schnittstellen müssen die Daten geparkt oder können direkt mittels API oder aus der Datenbank ausgelesen werden. Die Anfrageergebnisse werden dann in einer XML Datei aufbereitet und an den Viewer über die Sicherheitsschicht verschickt, wo diese dem Nutzer passend dargestellt werden.

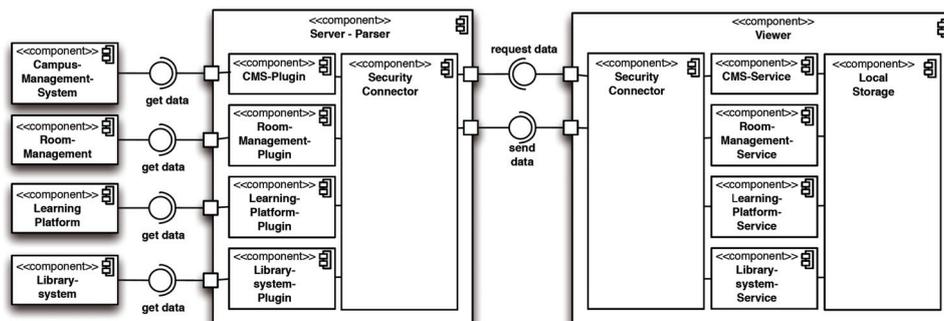


Abbildung 2: Eine zentrale Middleware entkoppelt Legacy-Applikationen von Benutzungsschnittstellen und bietet Ansatzpunkte für Mehrwert-Dienste

Hierbei wird auf die strikte Unabhängigkeit der Plugins verwiesen, die auch bei Änderungen in den angeschlossenen Legacy-Anwendungen eine schnelle und flexible Anpassung der Studi.UP ermöglichen, ohne dass andere Services oder Plugins beeinträchtigt werden. Auch die Einbindung weiterer Anwendungen oder die Bereitstellung auf verschiedenen Plattformen (nativ) bzw. als Web-Applikation sind so sauber gekapselt.

3.2 Anbindung von Legacy-Applikationen

Eine besondere Herausforderung stellte die Anbindung des Campus Management Systems PULS dar, das keine von den Entwicklern gepflegte, zukunftssichere API bietet. Der Studi.UP-Server bildet deshalb als Workaround die benötigte Schnittstelle mit Hilfe von Screen scraping nach. Dabei ist die Anbindung dynamisch konzipiert, um auf unerwartete Änderungen in der Web-Oberfläche dennoch mit passablem Aufwand reagieren zu können. Es wird daher möglichst wenig im HTML-Code von PULS geparkt; statt dessen werden die wichtigen Textfragmente mit Hilfe von separat abgelegten regulären Ausdrücken erfasst und in die Datenstruktur eingebunden. Das hat den Vorteil, dass bei einer Änderung in der Struktur des Seiteninhalts lediglich die Liste der gesuchten regulären Ausdrücke angepasst werden muss, ohne sich in den Code von Studi.UP einlesen zu müssen.

Weitere und nicht so aufwendige Optionen sind in Java vor allem durch vorhandene Projekte gegeben. Somit liefern Bibliotheken wie Tag Soup, Web-Harvest und HtmlUnit Funktionen die das Screen scraping und das Handling mit HTML dem Programmierer abnehmen.

Der Nachteil von vorhandenen Frameworks im Gegensatz zu einem eigenen System ist, dass man sich nur programmatisch an die Gegebenheiten des zu parsenden Systems anpassen kann. Falls wenige Änderungen in der Legacy-Applikation erfolgen oder gegebenenfalls der Support auch programmatische Änderungen einleiten kann, bieten Frameworks trotz Einarbeitungszeit immer eine effizientere Lösung.

3.3 Mobile Benutzungsschnittstelle

Nach dem Öffnen der Studi.UP gelangt der Benutzer auf eine Startseite, welche einen Überblick und einheitlichen Zugriff auf die Funktionen, die sog. Services, ermöglicht. Hier können zudem allgemeine Einstellungen bzw. Konfigurationen vorgenommen, z.B. Anmeldeinformationen für Legacy-Systeme hinterlegt werden. Die Stundenplan-Funktion (Abbildung 3) bietet beispielsweise eine Liste der tagesaktuellen Veranstaltungen und den Stundenplan der Woche. Durch Auswahl einer Veranstaltung gelangt der Benutzer zu einer Detailseite mit allen relevanten Informationen der ausgewählten Veranstaltung.

Die Oberfläche wurde mittels des Cocoa Touch Framework entwickelt, welche ein mobile Variante des Cocoa-Layer von Mac OS X ist. Von diesem wurden die Basisframeworks Foundation Kit und UIKit verwendet.

Das Design orientiert sich an den iOS Human Interface Guidelines und besteht aus den typischen iOS-Oberflächenkomponenten [AI12]. Lediglich die kompakte Darstellung des wöchentlichen Stundenplans ist eine Eigenentwicklung aus verschiedenen Oberflächenkomponenten.



Abbildung 3: Oberfläche der mobilen Benutzungsschnittstelle: Startseite und Stundenplan-Service

3.4 Sicherheitsaspekte

Der mobile Zugriff auf Daten der Hochschulsysteme wirft wie jede Nutzung dieser Systeme die Frage nach der Absicherung der Daten auf. Da die meisten Systeme personenbezogene Daten speichern oder zumindest personalisierte Daten bereitstellen (z.B. den individuellen Stundenplan), sind entsprechende Sicherheitsmechanismen vorzusehen [Eck08], die im Minimum die Nutzung eines verschlüsselten Transportweges über SSL/TLS beinhalten. Die Ziele dieser Sicherheitsarchitektur liegen in einer durchgehenden, transparenten Gewährleistung der datenschutzrechtlichen Anforderungen [DS11][BDS10] bei gleichzeitig hoher Usability für den Nutzer mobiler Geräte.

Um die Nutzer mobiler Systeme nicht mit der Einrichtung lokaler Maßnahmen zu belasten, werden in den neu entwickelten Anwendungen die Sicherheitsmechanismen weitgehend transparent integriert. Dazu zählen die Nutzung möglichst weniger einzugebender Identifikationsdaten auf dem Endgerät sowie die Verwendung modularer Entwicklungskonzepte durch eine eigenständige Sicherheitsschicht zwischen konventionellem Transport und der Anwendungslogik.

Im Hinblick auf eine spätere Umstellung auf eine serviceorientierte Architektur [Na06] stellt diese Schicht einen Mittelweg zwischen der Komposition von Services und Daten und den bestehenden klassischen Authentifizierungsmechanismen dar. Durch die Verwendung lokaler „digitaler Brieftaschen“ auf dem mobilen Gerät durch die Sicherheitsschicht kann dabei ein benutzerfreundliches Single-Sign-On simuliert werden, auch wenn solche Verfahren nicht in den Legacy-Systemen genutzt werden.

Um die Vertrauensstellungen der beteiligten Systeme sicherzustellen, kommen entsprechende Zertifikate zum Einsatz, welche die Server als Hochschulsysteme authentifizieren und durch deren Einsatz die Zugriffe auf personenbezogene Daten in verwendbarer, unverschlüsselter Form auf die berechtigten Stellen beschränkt werden können.

Die Verwendung von Daten wird dabei durch dienstspezifische Policies auf feinerer Granularitätsstufe gesteuert. Diese Policies werden maschinenlesbar angelegt und bestimmen, welche Nutzer und weitere Dienste bei der Verschlüsselung der Daten berücksichtigt werden und somit die Daten verarbeiten können. Die Policies greifen dabei in erster Linie bei der Weitergabe der Daten. Inwieweit diese Policies aber im eigenen Dienst, der ja bereits Zugriff auf die Daten hat, auch respektiert werden liegt in der Verantwortung des Entwicklers.

4 Zusammenfassung und Ausblick

Der Beitrag stellt eine Fallstudie für die Bereitstellung verschiedener Campus-Applikationen auf mobilen Geräten vor. Dabei wurde besonderes Augenmerk auf die Integration von herkömmlichen Legacy-Anwendungen gelegt, die in absehbarer Zeit noch keine mobile Schnittstelle bieten werden, aber dennoch in mobile Mehrwertdienste eingehen sollen. Neben einer sauberen architektonischen Trennung und damit künftiger Wartbarkeit und Erweiterbarkeit sind auch Sicherheitsaspekte von Bedeutung.

Der vorgestellte Prototyp wurde als studentisches Projekt entwickelt und ist noch nicht marktreif. Sowohl in der Robustheit als auch in den Wartungsmöglichkeiten im System müssen noch Verbesserungen erfolgen. Aktuell wird daran gearbeitet, das System an einzelnen Stellen noch dynamischer zu gestalten, damit bei Veränderungen der Randsysteme weniger Anpassungsbedarf entsteht.

Nach dem erfolgreichen Entwickeln des Prototyps sind einige Erweiterungen geplant. Darunter fällt unter anderem die Verbreitung der App auf weitere Betriebssysteme, wie zum Beispiel Android und Windows, woran derzeit im Rahmen eines Anschlussprojekts gearbeitet wird. Eine interessante Option ist darüber hinaus, das derzeit auf den Clients eingesetzte Polling durch Push-Benachrichtigungen seitens der Server zu ersetzen.

Außerdem können mit der App künftig die freien Räume in der näheren Umgebung eingesehen werden. Diese werden je nach Campus (die Universität Potsdam ist auf mehrere Standorte im gesamten Stadtgebiet verteilt) einzeln aufgelistet. Auch die Navigation über eine Karte des Campus ist vorgesehen.

Der Mensaplan wird pro Campus einsehbar sein. Hierbei werden die einzelnen Menüs und Inhaltsstoffe aufgeführt; künftig ist auch eine Bewertung durch die Nutzer denkbar. Hinzu kommt die Idee, die Universitätsbibliothek und deren Verwaltungssystem in die App zu integrieren. Dadurch hat man immer den Überblick über die reservierten oder ausgeliehenen Bücher sowie auch möglicherweise angefallene Gebühren.

Sukzessive soll so für die Studierenden ein Funktionsumfang entstehen, der den universitären Alltag auch auf mobilen Geräten abdeckt. Das System wird dabei schrittweise Akzeptanztests im universitären Praxisbetrieb unterzogen, um bislang unerkannte Probleme auch aus nicht-technischer Perspektive identifizieren und beseitigen zu können. Eine Frage ist hierbei etwa, ob die im System bekannten Informationen zur Raumbelastung tatsächlich aktuell genug sind, um den angestrebten Mehrwert zu bieten. Eine praktikable Antwort auf schlechte Datenqualität könnte hier z.B. das Crowdsourcing [MKK11] aktueller Daten bilden, indem mobile Nutzer selbstständig auf veraltete Informationen im Raumplanungssystem hinweisen können.

Literaturverzeichnis

- [AI12] Apple Inc. (2012): iOS Human Interface Guidelines.
<https://developer.apple.com/library/ios/#documentation/UserExperience/Conceptual/MobileHIG/Introduction/Introduction.html>
[Eingesehen am 20.06.2012]
- [BB10] A. Bode, R. Borgeest: „Informationsmanagement in Hochschulen“, Springer, 2010.
- [BGB11] Bührig J.; Guhr N.; Breitner M.: „Technologieakzeptanz mobiler Applikationen für Campus-Management-Systeme“, in Proc. Informatik 2011, Bonn : Köllen, S. 461.
- [BDS10] Die Landesbeauftragte für den Datenschutz und für das Recht auf Akteneinsicht Brandenburg: Brandenburgisches Datenschutzgesetz; in: Brandenburgisches Informationsgesetzbuch, Teil 1: Datenschutzgesetze, Heft 1; 6. Auflage, 2010.
<http://www.lda.brandenburg.de>
- [DS11] Die Landesbeauftragte für den Datenschutz und für das Recht auf Akteneinsicht Brandenburg: Bundesdatenschutzgesetz; in: Brandenburgisches Informationsgesetzbuch, Teil 1: Datenschutzgesetze, Heft 2; 4. Auflage, 2011. <http://www.lda.brandenburg.de>
- [CY05] L. Cohen, A. Young: “Multisourcing: Moving Beyond Outsourcing to Achieve Growth and Agility“, Gartner, 2005.
- [Eck08] Eckert C.; IT-Sicherheit: Konzepte – Verfahren – Protokolle; Oldenbourg Wissenschaftsverlag GmbH, München; 5. Auflage, 2008.
- [MKK11] Man-Ching Yuen; King, I.; Kwong-Sak Leung: “A Survey of Crowdsourcing Systems”, in Proc. 3rd IEEE Int. Conf. on Social Computing, 2011 , S. 766-773.
- [Na06] A. Nadalin, C. Kaler, R. Monzillo, P. Hallam-Baker (Ed.s): “Web Services Security. SOAP Message Security 1.1”, OASIS Standard Specification, Februar 2006.
<http://docs.oasis-open.org/wss/v1.1/>
- [Ng10] Ngu, A.H.H.; Carlson, M.P.; Sheng, Q.Z.; Hye-young Paik: “Semantic-Based Mashup of Composite Applications”, IEEE Transactions on Services Computing, Vol. 3 , Nr. 1, 2010 , S. 2-15.
- [SK09] S. Schaffert & M. Kalz: „Persönliche Lernumgebungen: Grundlagen, Möglichkeiten und Herausforderungen eines neuen Konzepts“. In: Handbuch E-Learning. Vol. 5, Nr. 5.16, Köln : Deutscher Wirtschaftsdienst, 2009, S. 1-24.
- [Su07] Sucharov, T.: “Mainframe makeovers”, in: Information Professional, Vol. 4 , Nr. 6, 2007, S. 36-38.

Lufthansa Privacy Quiz

Udo Sonne, Prof. Dr. Bettina Harriehausen-Mühlbauer

PL/B
Lufthansa AG
Airportring 1
60546 Frankfurt am Main
udo.sonne@dlh.de
bettina.harriehausen@h-da.de

Abstract: Die Lern-App „Quiz Lounge“ entstand im Rahmen eines Kooperationsprojekts der Hochschule Darmstadt und der Lufthansa AG. Ziel war es, eine mobile Lernanwendung zu entwickeln, mit der das Lufthansa-Management Inhalte zu Datenschutz und -sicherheit lernen kann. Aufgrund der interdisziplinären Zielsetzung des Projekts, wurden Studierende aus drei Fachbereichen (Informatik, Media, Wirtschaft) an diesem Projekt beteiligt. Ihre Motivation und Aufgabe bestand darin, nach einem originellen Weg zu suchen, die Lerninhalte für das Themengebiet „Data Privacy“ auf spielerischem Weg zu vermitteln. Für die Studierenden war es eine große Herausforderung, nicht nur den Ansprüchen der betreuenden Professoren gerecht zu werden, sondern auch denen eines namhaften Wirtschaftspartners. Die Nähe zur Industrie war ein zusätzlicher Ansporn, einen produktreifen Prototyp am Ende des Projekts übergeben zu können. Im Januar 2012 wurde die App erfolgreich innerhalb der Lufthansa AG unter 4000 Managern ausgerollt.

1 Privacy App

Die Privacy App beginnt mit einem Willkommen-Bildschirm, der einen sog. "Nice-to-know fact" aus dem Themenbereich der Datensicherheit anzeigt und mit einem einleitenden „Wussten Sie schon?“ auf einen Missstand in der Welt des Datenschutzes hinweist. Diese Information wird angezeigt, um den Benutzer zu informieren und auch mit einer Information zu überraschen, die er selbst noch nicht gewusst hat. Dies soll sein Interesse für das Themengebiet und auch die Neugier für die App wecken.

Nachdem das Interesse des Benutzers geweckt wurde, gelangt er in das Hauptmenü, aus dem er entweder ein Spiel starten kann, ein bereits begonnenes Spiel fortsetzen kann, einen Fachbegriff in einem Glossar nachschlagen oder die App beenden kann.

Entscheidet sich der Benutzer für das Spielen und wählt ein Quiz aus, so gelangt er in den Spielmodus, der mit der ersten Frage eines Quiz beginnt. Jedes Quiz besteht aus 10 Fragen unterschiedlicher Schwierigkeitsstufen: leicht, mittel und schwer. Wird eine Frage richtig beantwortet, so erhält der Spieler über eine grün hinterlegte Antwort nicht

nur ein optisches Feedback, sondern textuell wird die Frage auch detailliert erläutert. Eine falsche Antwort führt zu einer entsprechend rot hinterlegten Erklärung.

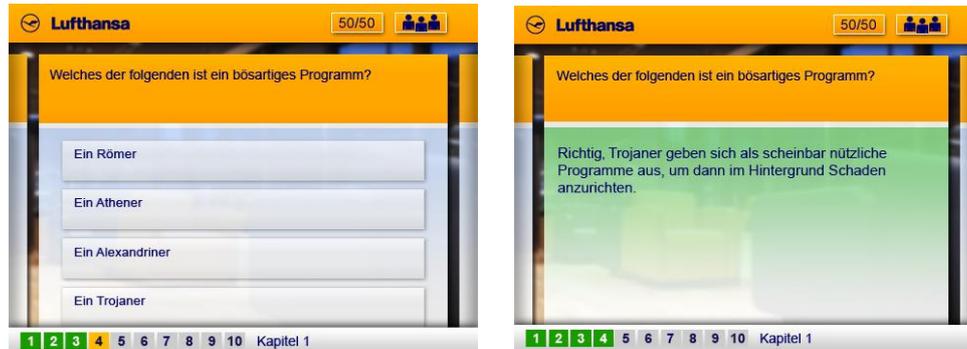


Abbildung 1: Screenshot der Privacy App

Um dem Spiel einen weiteren Anreiz zu geben, wurden zwei unterschiedliche Joker implementiert, die auch aus der verwandten Fernsehserie bekannt sind: der Zuschauerjoker und der 50:50 Joker. Wird der Zuschauerjoker gewählt, so wird aus den Antworten, die andere Spieler zu dieser Frage gegeben haben, eine Wahrscheinlichkeit errechnet und angezeigt.

2 QuizLounge Autorentool

Da das Projekt zeitlich begrenzt war und der Lufthansa AG auch die Flexibilität gegeben werden sollte, die App inhaltlich in der Form zu verändern, dass bestehende Quizze erweitert und neue mobile Quizze mit anderen Inhalten einfach und ohne Programmieraufwand erstellt werden können, wurde ein individuelles, Web-basiertes Autorentool (PHP) konzipiert und entwickelt, welches es dem Benutzer durch eine selbsterklärende Benutzeroberfläche ermöglicht, direkt auf die Inhalte zuzugreifen, bestehende Inhalte zu verändern, neue Fragen hinzuzufügen oder ein komplett neues Quiz zu einem anderen Thema, z.B. Klima & Umwelt, zu erstellen. Das Autorentool stellt neben Inhalte für die Blackberry Version des Spiels gleichzeitig die Inhalte für die iPhone- und Android-Versionen der Privacy App zur Verfügung.

Nachdem Änderungen vorgenommen wurden, konvertiert das Autorentool die Inhalte in XML Dateien, die über einen Server auf das Smartphone geschickt werden, wo die App sie lokal als Java-Objekte in einer SQLite Datenbank ablegt. Dieser Update-Mechanismus reagiert dynamisch auf Änderungen, indem er mit dem Apache Server kommuniziert, sobald eine Internetverbindung besteht. Somit wird garantiert, dass die Inhalte der App immer auf dem aktuellen Stand sind.

Die Zukunft des E-Learning

Martin Kapp, Lieven Leue, Tina Beigel, Ulrike Lucke

Universität Potsdam, Institut für Informatik, Lehrstuhl Komplexe Multimediale
Anwendungsarchitekturen, A.-Bebel-Str. 89, 14482 Potsdam

ulrike.lucke@uni-potsdam.de

Das Video stellt die momentane Lage von Studierenden einer Hochschule einer denkbaren Vision für die Zukunft gegenüber, die wesentlich durch E-Learning geprägt ist. Neben orts- und zeitunabhängigen Abläufen sowie einer Integration verschiedener Werkzeuge und Inhalte spielen dabei auch mobile Geräte eine wichtige Rolle.

Das Video wurde von drei Bachelor-Studierenden des Instituts für Informatik an der Universität Potsdam im Rahmen der Lehrveranstaltung „Multimedia-Technologie“ als wahlobligatorische Praxisaufgabe erstellt. Es war lediglich das Thema gegeben (eine Zukunftsvision des E-Learning an der Hochschule); die Studierenden erarbeiteten sich selbstständig eine konkrete Vorstellung von der Zukunft und setzten sie als Video um.

The screenshot shows a Moodle forum thread with three posts. The first post is the assignment, the second is a confirmation of the task, and the third is the submission of a video.

Post 1: E-Learning-Vision
von [Ulrike Lucke](#) - Dienstag, 29. November 2011, 16:47
Erstellen Sie einen ca. 3-minütigen Film, der eine Zukunftsvision von E-Learning an der Hochschule entwickelt!
[Ursprungsbeitrag](#) | [Bearbeiten](#) | [Thema teilen](#) | [Löschen](#) | [Antwort](#)

Post 2: Re: E-Learning-Vision
von [Martin Kapp](#) - Mittwoch, 30. November 2011, 23:24
Martin Kapp
Lieven Leue
Tina Beigel
übernehmen diese Aufgabe.
[Ursprungsbeitrag](#) | [Bearbeiten](#) | [Thema teilen](#) | [Löschen](#) | [Antwort](#)

Post 3: Re: E-Learning-Vision
von [Martin Kapp](#) - Montag, 30. Januar 2012, 19:38
[MMT.pdf](#)
Es ist ein wenig mehr geworden als 3 Minuten aber man verbringt nunmal in der Zukunft viel mehr Zeit mit E-Learning. ;)
Viel Spaß beim Gucken:
Die Zukunft des E-Learning!
<http://dl.dropbox.com/u/5788397/Die%20Zukunft%20des%20E-Learning.mp4>
[Ursprungsbeitrag](#) | [Bearbeiten](#) | [Thema teilen](#) | [Löschen](#) | [Antwort](#)

Abbildung 1: Aufgabenstellung und Abgabe im Moodle-Forum

Zunächst wird in dem Video das *UPbook* als zentrale Plattform für die Verwaltung von Lehrveranstaltungen und Inhalten, für die Kommunikation und Kollaboration gezeigt. Hierbei handelt es sich um eine persönliche Lernumgebung [SK09], die neben einer situativen Einbettung der o.g. Aspekte [LS12] durch Sprachsteuerung und geräteübergreifende Interoperabilität begeistert. Besonders interessant ist schließlich die

Darstellung einer holografischen E-Lecture, d. h. einer Projektion von Dozent und Folien ausgehend vom Smartphone auf dem Küchentisch, während sich der Student sein Müsli zubereitet.



Abbildung 2: Die holografische Vorlesungsaufzeichnung in einer Szene des Videos

So wird als ein Vorteil des E-Learning-Einsatzes durch die studentischen Autoren des Videos insbesondere der Wegfall von zeitintensiven ÖPNV-Fahrten gezeigt. Darüber hinaus sind aber auch vereinfachte Organisations- und Kommunikationsprozesse deutlich erkennbar – insgesamt ein Gewinn an Zeit, Komfort und Zuverlässigkeit durch den Einsatz von E-Learning auf stationären wie mobilen Geräten.

Es mag sich bei diesem Beitrag zwar um kein existierendes oder auch nur in Kürze vollständig realisierbares Projekt handeln, doch zeigt das Video interessante Möglichkeiten für technische Weiterentwicklungen auf. Als Ideengeber kann es somit helfen, die Attraktivität des Forschungsfelds E-Learning für künftige DeLFI-Tagungen und -Workshops zu erhöhen.

Das Video ist unter folgender URL zum Download verfügbar:

http://apache.cs.uni-potsdam.de/de/profs/ifi/mm/video_04_eLearning.mp4

Referenzen

- [LS12] U. Lucke, M. Specht: „Mobilität, Adaptivität und Kontextbewusstsein im E-Learning“. In: i-com, Vol. 11, Nr. 1, München : Oldenborug : 2012, S. 26-29.
- [SK09] S. Schaffert & M. Kalz: „Persönliche Lernumgebungen: Grundlagen, Möglichkeiten und Herausforderungen eines neuen Konzepts“. In: Handbuch E-Learning. Vol. 5, Nr. 5.16, Köln : Deutscher Wirtschaftsdienst, 2009, S. 1-24.

Angry Students: Mobile Multiplayer Action Game

Marc O. Kneifel, Ulrike Lucke

Universität Potsdam, Institut für Informatik,
Lehrstuhl Komplexe Multimediale Anwendungsarchitekturen,
A.-Bebel-Str. 89, 14482 Potsdam

ulrike.lucke@uni-potsdam.de

Gegenstand des Videobeitrags ist ein push-basiertes Messaging-System, das in großen und heterogenen Umgebungen bei der zeitnahen Verteilung von Daten durch/an mobile Geräte helfen soll [Sc12], ohne den jeweiligen Applikationsserver zum Flaschenhals werden zu lassen. Es wurde im Rahmen einer studentischen Arbeit am Institut für Informatik der Universität Potsdam entwickelt.

Das sog. Pushfork-System bietet mobile Broadcasts und Multicasts in Echtzeit, z B. für GeoLocation-Dienste [GG12]. Es wird zwischen den Clients und beliebigen Anwendungsservern platziert und synchronisiert deren Datenabgleich ohne die Anwendungsserver zu überlasten. Voraussetzung ist lediglich eine Registrierung sowohl der Anwendungen als auch der Clients bei Pushfork. Das System bietet Schnittstellen für Android, iOS, Windows Phone und Java. Einsatzbereiche sind alle Anwendungen, die für eine große Zahl an Clients häufige Updates z. B. der einzelnen Geo-Koordinaten benötigen.

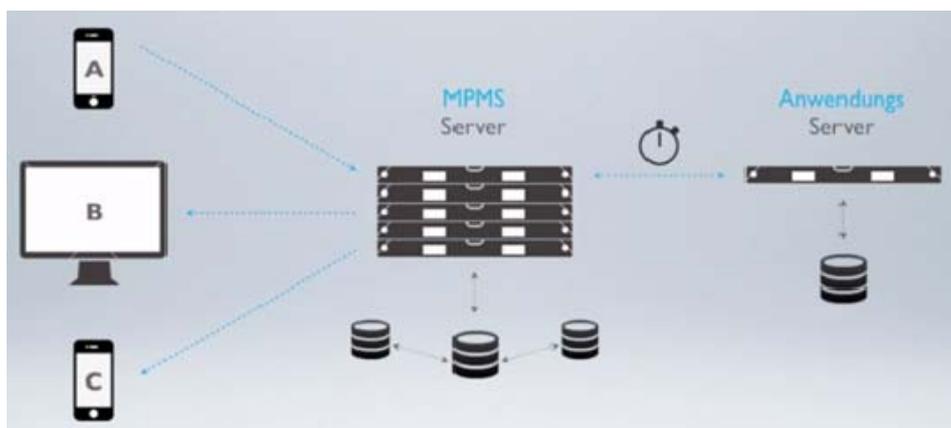


Abbildung 1: Clients erhalten schnell Datenupdates, ohne den Anwendungsserver zu belasten

Dieses System wird am Beispiel eines mobilen Spiels demonstriert, das Studierenden bei der Frustrbewältigung helfen soll. Ähnlich einem EgoShooter werden andere Spieler

aufgespürt und vernichtet, jedoch wird hierbei zusätzlich die reale Position des Spielers ausgewertet. Es sind also diejenigen Spieler auf dem Spielfeld sichtbar, die sich in der näheren Umgebung – und damit vermutlich in einer ähnlich stressigen Situation, langweiligen Veranstaltung, endlosen Wartepause, ... befinden. Ähnlich wie etwa bei einer Content Sharing Plattform handelt es sich somit um keine unmittelbare Unterstützung von Lernprozessen an sich, sondern um eine allgemeinere Anwendung die u. a. auch im Umfeld des Lernens sinnvoll eingesetzt werden kann.



Abbildung 2: Das mobile Spiel Angry Students basiert auf Messaging mittels Pushfork

Pushfork sorgt hier dafür, dass unabhängig von der jeweiligen Plattform und ohne zusätzliche Last auf dem Spieleserver die Geokoordinaten aller Spieler zeitnah ausgetauscht werden. Die Infrastruktur ist dabei unabhängig von der verwendeten Anwendung; beliebige ortbewusste Applikationen bzw. mit Geosensoren ausgestattete Geräte können hier angebunden werden. Konkret befindet sich die Nutzung im Rahmen des pervasiven Lernspiels RouteMe [ZML12] in Vorbereitung.

Referenzen

- [GG12] L. D. Gardner, J. Grigsby: „Mobiles Web von Kopf bis Fuß“, O'Reilly 2012.
- [Sc12] D. M. Scott: „Die neuen Marketing- und PR-Regeln im Social Web“, MIT Press 2012.
- [ZML12] R. Zender, T. Moebert, U. Lucke: „RouteMe – Routing in Ad-hoc-Netzen als pervasives Lernspiel“, in Proc DeLFI 2012, Bonn : Köllen, vorauss. September 2012 (im Druck).