

Spielbasiertes Lernen mit nutzergenerierten Inhalten

Michael A. Herzog, Christian Kiefer, Jürgen Sieck
FHTW Berlin

Abstract: Das Education 3.0 Zeitalter ermöglicht mit neuen Web-Technologien die Wiederentdeckung historischer Lernkulturen in einer ganz neuen Qualität. Studierende wie Wissensarbeiter erobern, organisieren und archivieren sich ihre vernetzte Welt der Fakten, sie erschaffen sich damit ihren persönlichen Wissenskosmos aus kleinteiligen Inhalten (Microcontent). Der Trend zum nutzergenerierten Open Content fördert eine neue Art der Informationsbeschaffung und des Austausches zwischen den Lernenden. Ausgehend von einer Bewertung dieses didaktischen Umfelds wird in diesem Artikel ein neues Anwendungsszenario zum spielbasierten elektronischen Lernen mit nutzergenerierten Spielinhalten entwickelt. Die technische Umsetzung des leicht zugänglichen Web-Portals auf der Basis flexibler Rich Internet Technologien wird hinsichtlich der Architektur und der Realisierung wichtiger Funktionen diskutiert. Mit der vorgeschlagenen Lösung wird ein wirkungsvoller Weg aufgezeigt, wie ein offen konzipiertes Quiz-Portal mit stationären wie mobilen Nutzungsmöglichkeiten in spielerischer Herangehensweise neue Zielgruppen für das E-Learning motivieren kann.

Ausgangssituation und didaktisches Umfeld

E-Learning wird oft aus demselben Grund wie traditioneller Unterricht als langweilig angesehen: Es fokussiert eher auf die Inhaltspräsentation als auf ein Lernerlebnis [E104]. So fehlt es häufig an Interaktivität und hat in seiner immer noch meist verbreiteten Form – dem Bereitstellen von Kursmaterialien auf statischen Websites oder in Learning Management Systemen – das E-Learning in seinem Anliegen eher verfehlt und damit dessen Ansehen geschadet [PP01].

Eine Art von Interaktivität, die in vielen E-Learning-Anwendungen vertreten ist, sind Multiple- und Single-Choice-Aufgaben. Neben der Möglichkeit Abfragen zum Vermitteln und Festigen von Sachwissen zu nutzen, bieten sie gute Voraussetzungen zur relativ einfachen und schnellen Herstellung auf Web- und Lernplattformen. *Effizienz* ist einer der wichtigsten Akzeptanzkriterien bei der Herstellung und Nutzung von Services, insbesondere dann, wenn nicht Medienprofis die Angebote erstellen sollen, sondern die Lehrenden und Lernenden selbst nach den Paradigmen von Web 2.0-Anwendungen an der Medienherstellung maßgeblich beteiligt werden.

Innerhalb interaktiver Quiz-Spiele können sich Lernende ihr Wissen selbst bestätigen. Wirkungsvolle Lernspiele verleihen Selbstsicherheit und zeigen Wissenslücken auf. Anhand eines aussagekräftigen Feedbacks können zudem Sachwissen vermittelt und aufgezeigte Wissenslücken unmittelbar geschlossen werden. Besonders in der Prüfungsvorbereitung sind solche interaktiven Lernanwendungen nicht nur eine spannende und unterhaltsame, sondern auch eine sehr effektive Art des Lernens. Ein

besonderer didaktischer Vorteil der Wissensvermittlung in Quiz-Form ist die Möglichkeit, als eintönig angesehene Lerninhalte in einer spielerischen Form zu erschließen. Quiz können deshalb im Sinne des Game Based Learning genutzt werden, um auch Zielgruppen für das E-Learning zu erreichen, die wenig auf andere didaktische Formen und schon gar nicht auf Web-Based Trainings oder das Lernen aus Büchern und Skripten ansprechen (vgl. auch [Pr01]). Eine für das Game Based Learning prädestinierte Zielgruppe stellen Jugendliche der „Generation Playstation“ dar, die mit Spielkonsolen aufgewachsen sind und teilweise kaum noch mit herkömmlichen pädagogischen Mitteln zum Lernen motiviert werden können. Ebenso zählen zu den Zielgruppen auch Lernende mit eher geringer Medienkompetenz und fehlender Fähigkeit zur Selbststeuerung, was oft für einen produktiven Umgang mit Web Based Trainings oder Computer Based Trainings erforderlich ist [MS02]. Neben Jugendlichen und jungen Erwachsenen werden mittlerweile auch weitere Altersgruppen mit Computerspielen erreicht. Casual Games¹, zu denen im besonderen auch Wissensspiele zählen, erfreuen sich etwa bei „Silver Gamern“, Spielern fortgeschrittenen Alters, zunehmend großer Beliebtheit.²

Der in diesem Beitrag nachfolgend vorgestellte technische Ansatz zum Spielbasierten elektronischen Lernen mit nutzergenerierten Spielinhalten wurde vor dem Hintergrund entwickelt, dass sich im E-Learning ein Paradigmenwechsel abzeichnet. Eine knappe phänomenologische Systematik zur Entwicklung des E-Learning aus heutiger Sicht, wurde in Tabelle 1 zusammengestellt. Die Aufstellung bringt zum Ausdruck, wie sich Lehren und Lernen mit den Möglichkeiten des offenen Informationszugangs und der elektronischen Kommunikation verändert. Mittel- bis langfristig ist zu erwarten, dass sich Mischformen des Lernens aus den Kategorien 1 bis 3 bilden und etablieren werden, allerdings mit einem deutlichen Trend zu Individualisierung und vom aktiven Lernenden ausgehenden Lernprozess, d.h. einer Verschiebung hin zum »E-Learning 3.0« Paradigma.

In Tabelle 1 spiegelt sich eher der von Hochschulen getragene Entwicklungspfad wieder, als eine *umfassende Veränderung der Lernkultur*. Allerdings wird gezeigt, dass gerade lebenslanges Lernen durch kleinteiligere, natürlich wirkende Lernangebote unterstützt wird, ohne dass entsprechend breit konzipierter Content aufwändig produziert werden muss. Diese Form liegt auch dem »Learning on the job« Prinzip am Nächsten. Nicht alle Lerninhalte lassen sich in einem solchen Konzept transportieren, aber für viele Bereiche scheint sich dieser Trend abzuzeichnen. Empirische Forschung zum Vergleich der Wirksamkeit der skizzierten Ansätze steht derzeit noch aus. Der in diesem Artikel skizzierende Systemansatz wurde für die Kategorie *E-Learning 3.0* entwickelt.

¹ Vgl. http://www.igda.org/casual/IGDA_CasualGames_Whitepaper_2005.pdf, Abruf am 20.3.2006

² Besonders im mobilen Bereich sind Wissensspiele wie „Dr. Kawashimas Gehirnjogging“ (für Nintendo DS) derzeit in den Listen der meistverkauftesten Videospiele vertreten. In den Videospiele-Verkaufscharts vom 30.1.2007 für Deutschland war der Nachfolger „Dr. Kawashima: Mehr Gehirntraining“ auf Platz 1 und mit „Practise Englisch“ sogar ein vollwertiges Lernspiel auf Platz 6.

	E-Learning 1.0	E-Learning 2.0	E-Learning 3.0
Technische Komponenten	Courseware Learning Management Systeme (LMS), Autorenwerkzeuge	Learning Content Management Systeme (LCMS), Diskussionsgruppen, Blogs	Wiki, Social networking & bookmarking, Add-ins, Mash-ups
Akteure	Top-Down Lehrer-getrieben	Kollaborativ kooperativ	Bottom-up Lerner-getrieben Peer-learning
Gegenstand	Faktenwissen (Know That)	Prozedurales Wissen (Know How)	Kombination aus Fakten-, prozeduralem und sozialem Wissen
Entwicklung	Lang und aufwändig	Schnell und effizient	Nahezu kein Entwicklungsaufwand
Rezeptionsumfang Nutzungszeit	60-90 min Vor oder nach der Arbeit	10-20 min In Pausen	1-3 min Während der Arbeit
Distribution Zugriff	In einem Stück LMS	In vielen Teilen E-Mail, Forum, Blog	Bei Bedarf Suche, RSS Feed
Treiber und Contentersteller Inhalt	Lehrer Traditionelles, urheberrechtlich zuzuordnendes Lehrmaterial	Partizipativ (Mehr Lehrer als Lerner) Traditionelles Material und OpenContent	Lerner, Community OpenContent, User generated Content
Rolle des Lehrers	Wissensquelle	Rahmengestalter, Arrangeur	Kritiker und Helfer
Rolle des Lerners	Wissenssenke, eher passive Lernhaltung	Mix aus klassischen und kollaborativen, aktiven Lernformen	Gestalter eigener Wissensräume, selbstbestimmter Lernprozess

Tabelle 1: Genese des E-Learning³

Forschungsziel

Vor dem Hintergrund des beschriebenen Paradigmenwechsels ist die Begeisterung für Wissensspiele wie auch Bereitschaft zur Online-Zusammenarbeit sehr hoch. Quiz-Spiele bieten eine ideale Voraussetzung, diese Trends für E-Learning zu nutzen: Sie können in einfachster Form schnell ohne spezielle Kenntnisse erstellt und genutzt werden. Aufgabe war es, ein flexibles Lernszenario und eine adäquate prototypische

³ Systematik und Inhalt dieser Tabelle wurden inspiriert von Tony Karrer, Artikeln von Derek W. Keats und J. Philipp Schmidt [Ke07] sowie von Peter Baumgartner [Ba06].

Softwareumsetzung zu finden, die unter Maximierung von Einfachheit, Effizienz und Flexibilität eine breitestmögliche Anwendungsperspektive bietet.

Forschungsteoretisch wurde dabei nach der Methodologie der Designwissenschaften (Design Research) verfahren (vgl. [VK07], [Jä07]), die ein neu zu entwickelndes IT-Artefakt in den Mittelpunkt stellt. Für das technische Experiment wurde die Web-Anwendung "Quizzer" konstruiert, die es über ein Online-Portal ermöglicht, Quiz für das E-Learning in Form von Single- und Multiple-Choice-Fragen zum spielebasierten Lernen innerhalb einer virtuellen Community zu nutzen. Ein extrem leicht zugängliches System soll seinen Nutzern erlauben, mit geringem Aufwand grafisch anspruchsvolle E-Learning-Quiz herzustellen und diese innerhalb der Community anzubieten und zum Spielen bzw. Lernen zu nutzen. Das System hat nicht den Anspruch, komplexe E-Learning-Portale zu ersetzen. Vielmehr wird mit diesem Ansatz untersucht, wie eine spielerische Form des E-Learning für Lernende wie Lehrende ohne die Barriere monolithischer und komplexer Architekturen zugänglich gemacht werden kann.

Da sich Quiz zum kurzfristigen Lernen oder Abfragen von Sachwissen anbieten, wird die Anwendung neben der Nutzung auf stationären Computern deshalb auch für mobile Geräte getestet. So wird es den Nutzern ermöglicht, quasi *jederzeit* spielerisch zu lernen. Für maximale Ortsunabhängigkeit soll die Anwendung auch ohne Netzverbindung weitestgehend alle Funktionalitäten, die für das entsprechende Endgerät sinnvoll sind, zur Verfügung stellen. Umgesetzt werden sollten dazu mindestens die didaktisch erprobten Entitäten Frage, Antwortoption, Feedback „Richtig oder Falsch“ und optional erklärendes Feedback.

Bestehende Lösungen

Bei der Entwicklung des inhaltlichen und technischen Konzepts wurden vor Allem verschiedene Ansätze in *Online-Quiz-Spielen* wie „quizgamemaker.com“ oder „Qmania“⁴ ausgewertet. Neben vielen positiven Aspekten, wie der Systemoffenheit für nutzergenerierte Inhalte, ist aufgefallen, dass Möglichkeiten zum gezielten Spielen von Themengebieten hier ebenso fehlen, wie ausführliche, erklärende Feedbacks. Um eine sinnvolle Nutzung für das Lernen zu ermöglichen, sind in diesen Systemen Erweiterungen oder konzeptionelle Änderungen erforderlich.

Learning Management Systeme wie Moodle, Ilias oder Clix⁵ bieten zwar die Möglichkeit, Quiz zu erstellen und den Lernenden zu Übungszwecken anzubieten bzw. als Aufgabe zu stellen. Eine **aktive** Mitwirkung der Lernenden im Sinne einer schöpferischen, kreativen, gestaltenden Rolle ist hier jedoch nicht intendiert, da das Anlegen und Verwalten von Quiz allein in der Hand von Nutzern mit administrativen

⁴ Vgl. <http://www.qmania.de/geschichte.php> bzw. <http://www.quizgamemaker.com>

⁵ Open Source LMS Moodle, <http://www.moodle.org/>, ILIAS, <http://www.ilias.de/>; Kommerzielles LMS Clix, <http://www.im-c.de/>

Rechten liegt. Darüber hinaus sind ergonomische Barrieren bei der Quiz- Herstellung eher die Regel. Dem Charakter der LMS als Verwaltungssystem für Lerninhalte adäquat, sind die umfangreichen Auswertungsfunktionen für Lehrende besonders augenfällig. Charakteristisch an den generierten Quiz der untersuchten Systeme ist zudem die optisch eher schwache Präsentation, die Fragebögen ähnelt und keinerlei spielerische Charakteristik zeigt.

Das Game Based Learning Szenario »Quizzer«

Beteiligung der Lernenden

Der entwickelte Systemansatz soll in erster Linie Nutzer *über Spielen zum Lernen motivieren*. Ein wichtiger Punkt, um Spieler zu erreichen, ist dabei die visuelle Präsentation, die hierfür unbedingt einen starken Spielcharakter vermitteln sollte, ohne allzu einfach oder infantil zu wirken und damit den didaktischen Hintergrund der Anwendung zu diskreditieren.

Die Plattform soll *aktives Mitwirken auf einfache Weise* ermöglichen. Um in Quizzer eigene Quiz-Fragen anzulegen ist lediglich das Eintragen einer Frage und der Antwortmöglichkeiten nötig. Dabei ist ein allgemeines "Richtig/Falsch"-Feedback vordefiniert, das verändert bzw. überschrieben werden kann.



Abbildung 1: Quizzer auf verschiedenen mobilen Geräten

Auch wenn im ersten Schritt noch keine ideale Quiz-Frage für das Lernen erzeugt wird – dazu wäre ein ausführliches Feedback nötig – können Nutzer aufgrund der Leichtigkeit im Erzeugen von Quiz schnell dazu bewegt werden, eine aktive Rolle zu übernehmen. Das Hinzufügen eines ausführlichen Feedbacks in Form von Texten (incl. Hyperlinks) ist auf ebenso einfache Art und Weise möglich, wie das Einbinden von Bildern, interaktiven Animationen, Videos und Audio-Dateien. Neben dem großen Nutzen für die

Spieler der Quiz-Fragen mit ausführlichem Feedback, erzielt der Autor durch den Prozess beim Erstellen der Quiz-Fragen und speziell der Feedback-Informationen erhebliche Lerneffekte durch die aktive Auseinandersetzung mit den Inhalten. Das Ergänzen von weiterführenden Erklärungen oder Verweisen zu externen Quellen durch andere Nutzer soll ebenso ermöglicht werden. Diese Erklärungen werden dem Spieler im Anschluss eines gespielten Quiz angezeigt.

The screenshot shows the 'Quizzer' website interface. At the top, there is a navigation bar with links for 'Startseite', 'Mein Quizzer', 'Quiz Generator', 'Community', 'Service', and 'Forum'. The main content area is titled 'Quiz-Resultate' and displays a table of quiz questions with their results and actions. The table has columns for 'Frage', 'Resultat', 'Informationen', and 'Aktionen'. Below the table, there are sections for 'Legende anzeigen', 'Resultat', 'Informationen', and 'Aktionen'. The left sidebar contains 'Mein Quizzer' and 'Help-Center' sections. The right sidebar contains 'Werbung' and 'Freunde online' sections.

Frage	Resultat	Informationen	Aktionen
1. Wie ist der Fachausdruck Lernen anhand von Spielen?	🟢	W 📄 📺 📺	👍 🗣️ 📄 📺
2. Wie heißt ein weit verbreitetes Open Source Learning Management...	🟡	W 📄 📺 📺	👍 🗣️ 📄 📺
3. Wie werden Variablen in Java deklariert?	🟡	W 📄 📺 📺	👍 🗣️ 📄 📺
4. Zu welcher Form des Lernens gehört das Kartellkartenprinzip?	🟡	W 📄 📺 📺	👍 🗣️ 📄 📺
5. Wie heißen die Standard zum Austausch von Lerninhalten mit Learning ...	🟢	W 📄 📺 📺	👍 🗣️ 📄 📺
6. Was versteht man unter Blended Learning?	🟢	W 📄 📺 📺	👍 🗣️ 📄 📺
7. Was versteht man unter Blended Learning?	🟡	W 📄 📺 📺	👍 🗣️ 📄 📺

Abbildung 2: Weiterführende Informationen zu einem gespielten Quiz

Über Wettbewerb motivieren

Der Trend zum Multiplayer-Game oder gar zu Community-Games ist derzeit in der Computerspieleindustrie sehr stark und die Anforderung der Spieler, gegen Freunde und Familienmitglieder spielen zu können, hat heute sehr hohe Priorität [JK07]. Diesem Trend folgend wurden für das Szenario sowohl Einzeler- als auch Mehrspielermodi konzipiert, so dass Wettbewerbe und ggf. auch Turniere eingerichtet werden können. In Wettbewerben fordern sich Nutzer so gegenseitig zu Quiz-Spielen heraus. Besonders die Möglichkeit, sein Wissen mit anderen Nutzern messen zu können, sorgt für eine dauerhafte Motivation.

Qualitätssicherung

Bei Community-Plattformen wie Wikipedia ist zu beobachten, dass deren Nutzer in der Regel einen gesunden Qualitätsanspruch an eingestellte Inhalte mitbringen, auch wenn diese These immer wieder kritisch diskutiert wird⁶. Dennoch ist es nach unserer

⁶ Vgl. z.B. Jimmy Wales in PC Welt: <http://www.pcwelt.de/it-profi/englishnews/53570/>

Auffassung notwendig, den Nutzern alle Möglichkeiten der Contenterstellung und -bearbeitung an die Hand zu geben, dazu gehören auch Funktionen, um fehlerhafte oder fragwürdige Inhalte (z.B. mit diskriminierendem Hintergrund) aus dem System zu entfernen und damit eigenverantwortlich Qualitätssicherung zu betreiben. Ein Mechanismus speist sich aus der Meldung kritischer Quiz-Inhalte durch die Nutzer, indem ab einer definierten Anzahl an Meldungen die Inhalte gesperrt werden. Administratoren entscheiden dann über die Löschung oder Freigabe gesperrter Quiz-Fragen, ggf. auch über den Ausschluss von Nutzern.

Neben der Qualitätssicherung durch Ausschluss ist auch die Bewertung der Inhalte ein inzwischen bewährtes Mittel der Qualitätskennzeichnung von Beiträgen und Autoren. Dieses „User Rating“ kann als Filter auf Ergebnisse von Suchanfragen angewendet werden (z.B. über Collaborative Filtering Algorithmen) und damit bei der Auswahl der Fragen hilfreich sein.

Weitere wesentliche Funktionen

Zum komfortablen spielerischen Lernen gehören neben einem geeigneten Szenario auch eine Reihe weiterer wichtiger Systemfunktionen.

Quiz anlegen und spielen. In einem Quiz werden Fragen auf Basis eines Kataloges, einer Kategorie oder der Ergebnisse von Suchanfragen ausgewählt. Die Anzahl der Fragen für ein Quiz wird durch den Benutzer festgelegt. Als Kontrahenten können Freunde, Mitglieder einer gemeinsamen Gruppe oder auch aus Suchanfragen resultierende Nutzer ausgewählt werden. Um im Mehrspieler-Modus eine maximale Chancengleichheit und im Einzelspielermodus einen möglichst hohen Lerneffekt (nach „Karteikarten-Methodik“) zu erzielen, wählt das System bevorzugt selten gestellte oder falsch beantwortete Quiz-Fragen für die Spiele.

Quiz-Fragen organisieren. Quiz-Fragen können in Katalogen gesammelt und geordnet werden. Diese Fragen-Kataloge können optional von Ihren Autoren öffentlich bereitgestellt oder auch nur für definierte Benutzer (-gruppen) veröffentlicht werden. Für eine intensive Zusammenarbeit sind auch Kataloge zur Bearbeitung und Erweiterung freizugeben.

Gruppen verwalten. Innerhalb des Portals wird das Abbilden von Lerngemeinschaften ermöglicht. In Gruppen können z.B. Studiengänge einen geschlossenen Bereich erstellen um Quiz-Fragen oder Quiz-Kataloge intern zu nutzen. In diesen internen Bereichen sind auch kollaborative Techniken wie Foren, Wikis bis hin zu (Video-)Chats einsetzbar. Speziell Lehrende können als Gruppenadministratoren Ihren Lernenden Quiz zum Training anbieten und anhand der gewonnenen Resultate Rückschlüsse über deren Wissensstand ziehen. Auch können Lernende direkt mit dem Erstellen von Quiz beauftragt werden. Auch Quiz-Turniere sind denkbar.

Statistiken anzeigen. Neben Auswertungsfunktionen für Gruppenadministratoren werden verschiedene Statistiken auch für Nutzer angeboten, die nach gespielten

Katalogen, Kategorien, Gegnern und Zeiträumen selektiert werden können und die den eigenen Lernfortschritt dokumentieren.

Kontextsensitive Hilfe. Ein kontextsensitives Hilfesystem bietet auf Basis der Auswertung von Benutzerfeedback, Suchbegriffen und Zugriffsraten fundierte Erweiterungs- und Optimierungsanregungen für Administratoren und Nutzer.

Mobile Funktionen. Aufgrund der eingeschränkten Bildschirmauflösung und Eingabemöglichkeiten für mobile Geräte werden einige Funktionalitäten für bestimmte Gerätetypen aus Usability-Gründen nicht freigegeben. So beschränkt sich die Nutzung am Mobiltelefon beispielsweise auf das Suchen und Spielen von Quiz, das Suchen und Herausfordern von Gegnern sowie das Einsehen von Ergebnissen und Statistiken.

Technische Umsetzung und Architektur des Systems

Bei der **Technologieauswahl** wurde sehr viel Wert auf die Nutzung moderner und zukunftsgerichteter Rahmenbedingungen unter Aspekten wie Modularisierbarkeit, Bereitstellungsmöglichkeiten von Schnittstellen und Performance gelegt. Um eine hochgradig interaktive Benutzeroberfläche im Stile traditioneller Desktop-Anwendungen auf Portal-Ebene zu ermöglichen, wurde für die Frontend-Implementierung die Rich-Internet-Application (RIA) Technologie *Adobe Flex* als Basis gewählt.⁷ Die Nutzung von *Adobe Flash* auf der Distributionsseite bietet herausragende Möglichkeiten zur Bereitstellung multimedialer und grafisch anspruchsvoller Inhalte⁸ (vgl. auch [KL08]). Im Gegensatz zu HTML-basierten RIA-Technologien wie Ajax werden mit dem Flash-basierten Flex Browserinkompatibilitäten und plattformspezifische Probleme umgangen (vgl. auch [Wen06] [Do06]). Der Einsatz von *Action Script 3* als klassenbasierte objektorientierte Programmiersprache liefert zudem eine performante Programmierung nach aktuellen Konzepten. Eine weitere Stärke von Flex ist die Möglichkeit, auch komplexe Daten Clientseitig (über so genannte Shared Objects) zu speichern und damit Server-Requests zu optimieren. Mit der kürzlich erschienenen Adobe Integrated Runtime⁹ können Flash-Anwendungen auch lokal installiert und – mit oder ohne Web-Anbindung – lokal eingesetzt werden, was für die Quizzer-Plattform in Kürze genutzt werden soll.

Eine höchstmögliche Modularität für das Portal wurde durch den Einsatz des Microarchitektur-Frameworks *Cairngorm* erzielt. Cairngorm ist die Implementierung

⁷ Das komponentenbasierten RIA Framework Flex erzeugt Shockwave Flash Dateien für den Flash Player 9, der laut aktuellen Statistiken bei über 93% der Internetnutzer (gesättigte Märkte) genutzt wird und kostenlos für alle gängigen Betriebssysteme verfügbar ist.

⁸D'Amore, S., Schrader, F.: Flex & LiveCycle – Im Gespräch mit Flex-Spezialist Christoph Rooms, VISUAL-X Magazine, Software und Support Verlag Frankfurt, 15.12.2006, <http://visualxmag.de/itr/news/psecom,id,32688,nodeid,370.html>, Abruf am 27.2.2008

⁹ AIR – Adobe Integrated Runtime, <http://labs.adobe.com/technologies/air/>

von Best-Practice Ansätzen und Design Patterns zur Entwicklung skalierbarer und wartbarer RIAs unter Flex [SW06]. Anwendungslogik und das Erscheinungsbild des Portals können durch die Nutzung von Cairngorm pragmatisch gepflegt und modular erweitert werden. Für erfahrende Design Pattern Entwickler ist der Programmcode zudem leicht zugänglich (vgl. auch [Ei06]).

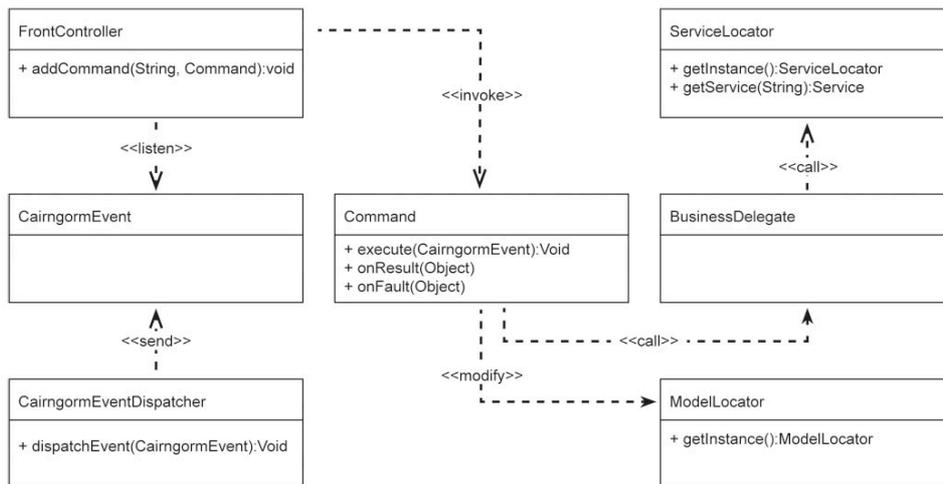


Abbildung 3: Cairngorm Übersicht

Im Folgenden werden die wichtigsten Aufgaben der generalisierten Klassen des **Flex-Frontends** erläutert. *MainApplikation* stellt die Programmoberfläche in speziellen Views bereit, initialisiert die für den Programmstart nötigen Prozesse (z.B. RMI) und organisiert die Seiteninhalte innerhalb des erzeugten „Hauptfensters“. Über *Views* werden programmablaufspezifische Layoutelemente dargestellt und Benutzerinteraktionen an das System mittels „CairngormEvents“ versendet. Innerhalb der *Events* werden Objekten mit ablaufspezifischen Daten weitergereicht, die für die Weiterverarbeitung durch die *Commands* genutzt werden. Im *Front Controller* werden die registrierten *Commands* auf die zugehörigen *Events* gemappt. *Commands* behandeln Ereignisse anhand der empfangenen *Event-Objekte* und rufen serverseitige Funktionen über *BusinessDelegates* auf. Das Interface zwischen *Commands* und *Remote-Services* wird mittels *BusinessDelegates* bereitgestellt. Über diese werden entfernte *Services* aufgerufen, die durch *Commands* angefragt werden. Die Resultate der *Service-Abrufe* werden an das aufrufende *Command* weitergeleitet und dort verarbeitet. Anschließend wird das *Model* in den aktuellen Zustand überführt. Das *Model* hält die für die *Applikation* notwendigen Daten und den *Anwendungszustand*. Mittels „Datenbindung“ werden *Views* anhand der Änderungen im *Model* aktualisiert. In *ValueObjects* werden spezifische Datenstrukturen mittels *Klassen/Objekten* abgebildet. Die Bereitstellung und Registrierung von *Services* (HTTP und Web Services) werden für einen globalen Zugriff innerhalb der *Anwendung* mittels eines *ServiceLocator* realisiert.

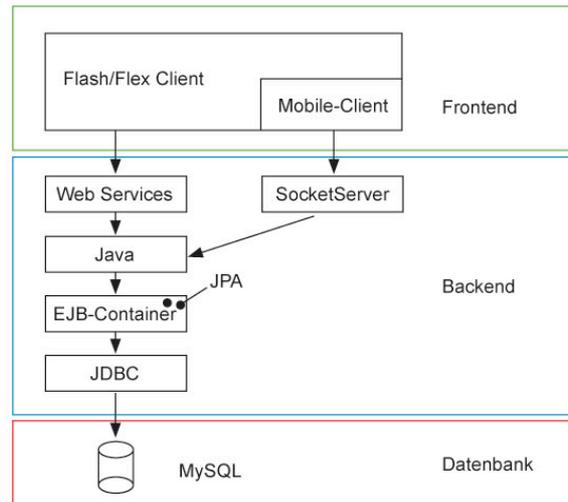


Abbildung 4: Quizzer-Architekturüberblick

Für viele **mobile Endgeräte** ist derzeit noch kein vollwertiger Flash Player verfügbar. Flash9-Dateien können häufig nicht wiedergegeben werden. Somit war es nicht möglich, die Funktionalitäten des Portals einheitlich für alle mobilen Geräte umzusetzen. Deshalb ist zusätzlich ein gesonderter Client auf Basis des Flash Player 7 erstellt worden. Dieser Client kann auf mobilen Playern (Flash Lite und Flash Pocket PC Player 7) unter einer Vielzahl von Smartphones, PDAs und Internet fähigen Spielekonsolen genutzt werden. Innerhalb des mobilen Client werden alle Elemente der grafischen Benutzeroberfläche über die Zeichenmethoden der Flash API visualisiert. Dadurch kann das komplette Layout mittels Stildefinitionen für alle Displayauflösungen bei geringem Aufwand optimiert dargestellt und gleichzeitig die Dateigröße des mobilen Client gering gehalten werden. Da Quiz-Fragen ein Feedback in Form von Medien mit hohem Datenaufkommen (Flash-, Video- oder Audio-Feedback) beinhalten können, kann der Anwender einzelne Feedback-Medien-Typen deaktivieren, um mangelnde Performance wie auch hohe Kosten durch eine datenintensive Medienübertragung zu vermeiden.

Der **Spielmodus** von Quizzer wird für die verschiedenen stationären und mobilen Endgeräte einheitlich in Flash 7 umgesetzt. Die Nutzung von Flash 9 (Action Script 3) würde für den Spielmodus keinen Vorteil bringen, da die benötigten Funktionalitäten bereits in der 7. Generation implementiert wurden. Aufgrund der Gestaltung des Layouts, das sich der Displaygröße des Clients anpasst, kann die nur 50 kb große mobile Anwendung in das Portal eingeladen werden und mittels Parametern direkt den Quiz-Modus aufrufen. So war es möglich, über einmalige Programmierung den Herstellung- und Wartungsaufwand gegenüber einer Geräte-spezifischen Lösung erheblich zu reduzieren.

Die **Client-Server-Kommunikation** des Portals wurde aus Kompatibilitätsgründen mittels *Java Web Services* realisiert. Die Geschwindigkeit in der Datenübertragung

unterscheidet sich unwesentlich im Vergleich zur Nutzung der Alternative Flash RemoteObjects.

Mobiler Echtzeit-Datenaustausch. Im simultanen Mehrspielermodus muss das „pushen“¹⁰ von Daten vom Server zum Client für sofortige Kommunikationsanforderungen ermöglicht werden. Hierfür können Web Services nicht genutzt werden. Für diesen Zweck wurde ein Java-basierter Socket Server entwickelt, über den XML-basiert nahezu in Echtzeit Daten ausgetauscht werden können. Im mobilen Portal werden auch Anforderungen von Daten im XML-Format über den Socket Server abgewickelt. Dieser kommuniziert mit den Java-Diensten der Web Services, um Daten aus der Persistenzschicht zu beziehen.

Persistenzschicht für Performance und Flexibilität. Um eine hohe Performance für die Datenbankabfragen zu erreichen wurde eine Persistenzschicht unter Nutzung der Java Persistence API (JPA)¹¹ realisiert. Das System erreicht durch Abfragen auf den Second-Level-Cache enorme Geschwindigkeitsvorteil gegenüber direkten Datenbankzugriffen [GPT07]. Als positiver Nebeneffekt wird implizit eine Abstraktion des Datenbankschemas und eine damit verbundene Unabhängigkeit vom Datenbankmanagementsystem erreicht. Zudem werden durch die Nutzung der Persistenzschicht die Applikationskomplexität und Entwicklungszyklen reduziert sowie die Wartbarkeit erhöht.

Die **Datenbank** wurde in MySQL realisiert. Aufgrund der flexiblen Systemarchitektur sind die Datenbank aber auch andere Frontend-Technologien einfach austauschbar.

Fazit und Ausblick

Mit dem entwickelten Quizzer-Portal¹² wurde eine umfangliche Architektur für das hier vorgestellte spielbasierte Lernszenario entwickelt, das einen leichten, offenen Zugang für Lernende und Lerngruppen im stationären wie mobilen Kontext bietet. Die Integration der Anwendung in andere kollaborative Umgebungen oder Verknüpfungen ist auf Basis der XML-Schnittstellen leicht realisierbar. Neben der Verknüpfung mit verschiedensten SocialSoftware-Technologien (Wikipedia, YouTube, Delicious) werden für das Portal verschiedene Erweiterungspläne verfolgt. So sollen etwa personalisierbare Avatare (im Stile von avatars.yahoo.com) für die Identifizierbarkeit der Nutzer mit den Spielfiguren sorgen und kollaborative Tools in Form von Chats und Foren integriert werden, um den Nutzern die Diskussion der Inhalte und Wettbewerbe zu ermöglichen.

¹⁰ Gemeint ist hier das Senden von Daten ohne explizite, direkte Anforderung

¹¹ Als JPA wurde hier die Referenzimplementierung „Toplink Essentials“ eingesetzt. Vgl.: <http://oss.oracle.com/toplink-essentials-jpa.html>

¹² Eine Testversion des Systems ist hier nutzbar: <http://norfolk.f4.fhtw-berlin.de:8180/QuizzerWebService/Quizzer.html>

Über einen SCORM-Exporter könnten viele Inhalte des Quizzer-Portals auch für andere Lernsysteme zur Verfügung gestellt werden.

Literaturverzeichnis

- [Ba06] Baumgartner, P.: The Zen Art of Teaching - Communication and Interactions in eEducation. Proceedings of the International Workshop ICL2004, Villach / Austria 29 September-1 October 2004, Villach, Kassel University Press. CD-ROM, online abrufbar <http://www.peter.baumgartner.name/article-de/the-zen-art-of-teaching>
Abruf am 27.2.2008
- [Do06] Domenig, M.: Rich Internet Applications und AJAX. In Entwickler-Magazin, 04/2006 <http://www.canoo.com/news/entwickler.pdf>
- [Ei06] Eisman, D.: Vortrag „Einführung in die Cairngorm Architektur für Flex 2“ im Rahmen der Flashforum-Konferenz Düsseldorf, 11/2006 <http://www.flexperten.de/wp-content/uploads/2006/11/CairngormDemoFFK06.zip>
Abruf am 14.2.2008
- [EI04] Ellis, R.: Down with boring e-learning! Interview with e-learning guru Dr. M. W. Allen, <http://www.learningcircuits.org/2004/jul2004/allen.htm>, Abruf am 27.2.2008
- [GPT07] Ghadir, P., Pagop, S., Tigges, O.: Die JavaPersistence API, in JavaSpectrum 2/2007, http://www.sigs.de/publications/js/2007/02/ghadir_pagop_JS_02_07.pdf
Abruf am 14.2.2008
- [Ke07] Keats, D.W., Schmidt, J.P.: The genesis and emergence of Education 3.0 in higher education and its potential for Africa, First Monday, online journal, volume 12, number 3 (March 2007), http://firstmonday.org/issues/issue12_3/keats/index.html,
Abruf am 27.2.2008
- [KL08] Kazoun, C., Lott, J.: Programmieren mit Flex 2, O'Reilly Verlag, Köln, 2008
- [MS02] Meier, M., Seufert, S.: Game-based Learning, in Hohenstein, A., Wilbers, K.: Handbuch E-Learning, Verlag Deutscher Wirtschaftsdienst, Köln 2003
- [Pr01] Prensky, M.: Digital Game-Based Learning, McGraw-Hill, New York, 2001
- [Pr05] Prensky, M.: EDUCAUSE review, September/Oktober 2005 <http://www.educause.edu/ir/library/pdf/erm0553.pdf>, Abruf am 14.2.2008
- [VK07] Kuechler, W., Vaishnavi, V. Design (Science) Research in IS: A Work in Progress. In proceedings of 2nd International Conference on Design Science Research in Information Systems and Technology (DESRIST '07), May 13-16, 2007, Pasadena, CA. <http://www.isworld.org/Researchdesign/drislSworld.htm>. Abruf am 27.2.2008
- [Jä07] Järvinen, P.: On Reviewing of Results in Design Research. In Österle, H., Schelp, J., Winter, R. (eds.): Proceedings of the Fifteenth European Conference on Information Systems, pp. 1388-1397, University of St. Gallen, 2007
- [Web06] Webster, S.: Developing Flex RIAs with Cairngorm Microarchitecture. Adobe Systems 2006, http://www.adobe.com/devnet/flex/articles/cairngorm_pt1.html,
Abruf am 28.2.2008
- [Wen06] Wenz, C.: JavaScript und AJAX, Galileo Computing, 2006