

Volker Stuckmann

Was Game Design und Mensch-Computer-Interaktion voneinander lernen können

Game Design, Mensch-Computer-Interaktion, Usability, Playability



Die Disziplinen Game Design und Mensch-Computer-Interaktion trennt noch immer ein tiefer Graben. Während sich auf der einen Seite die Vertreter von Anwendungssoftware gegen die in ihren Augen nutzlosen digitalen „Spielzeuge“ verwahren, scheuen auf der anderen Seite Game Designer vor allzu wissenschaftlichen Erkenntnissen zurück, die auf den ersten Blick nichts zum Spielspaß beitragen können. Zu zeigen, dass es sich für beide Disziplinen lohnt, eine Brücke über diesen Graben zu schlagen, ist das Ziel dieses Beitrags.

1. Einleitung

Computerspiele haben sich seit ihren Anfängen zu einer äußerst erfolgreichen Form von Software entwickelt. Die digitalen Spiele verzeichnen kontinuierlich steigende Umsätze und erreichen immer weitere Teile der Gesellschaft. So konnte sich auch die Wissenschaft nicht länger vor ihnen verschließen. Als anerkannte Medien stehen Computerspiele heute im Mittelpunkt der Ludologie, der Lehre vom Spiel, und beschäftigen Anthropologen, Soziologen, Psychologen, Künstler und Informatiker gleichermaßen. Als Software fallen digitale Spiele außerdem in den Bereich der Mensch-Computer-Interaktion (MCI) und sollten somit auch unter entsprechenden Gesichtspunkten betrachtet werden. Bislang konnten sich MCI-Forscher und Game Designer allerdings nur schwerlich mit der jeweils anderen Disziplin anfreunden – obwohl beide Seiten voneinander lernen und profitieren können.

2. Wie Game Design von der MCI profitieren kann

Mit dem wachsenden Markt nimmt auch der Konkurrenzkampf zu. Daher erscheint es umso wichtiger, Spiele zu entwerfen, die mit eingängiger Steuerung, ausgefallenen Spielwelten oder packenden Geschichten aus der Masse herausstechen. Tatsächlich erkannten Spieleentwickler die Bedeutung guter Spielbarkeit für den Spielspaß schon Mitte der achtziger Jahre und prägten den Begriff der „Playability“. Abgesehen

von informellen Game Design-Prinzipien existieren allerdings bis heute keinerlei Richtlinien darüber, wie diese „Spielbarkeit“ genau zu erreichen ist. Stattdessen orientieren sich die meisten Game Designer noch immer an Erfahrungen, erfolgreichen Spielen oder schlicht ihrer Intuition. Andererseits beschäftigt sich ein ganzes Forschungsfeld der Informatik mit der Gestaltung der Schnittstelle zwischen Benutzer und Computer: das Fachgebiet MCI brachte unter dem zentralen Begriff der „Usability“ zahlreiche innovative Ideen hervor, um interaktive Systeme möglichst benutzergerecht zu gestalten. Zudem zollt man mit dem Ansatz der „User Experience“ inzwischen auch den emotionalen Aspekten von Software Anerkennung und nähert sich damit dem „Spielspaß“ digitaler Spiele an. Die Computerspiel-Branche verweigert sich bislang diesen Erkenntnissen, obwohl das Game Design durchaus und in hohem Maße von Konzepten der MCI profitieren kann.

2.1 Playability

Das Game Design kann zunächst Nutzen aus dem theoretischen Hintergrund der MCI ziehen: Er ermöglicht ein Grundgerüst für die Gestaltung „spielbarer“ Computerspiele. Ein zentraler Aspekt hierbei ist die Usability, die Gebrauchstauglichkeit von Hard- und Software. In ihrem Mittelpunkt stehen die drei Komponenten Effektivität, Effizienz und Zufriedenstellung. Computerspiele unter diesen drei Gesichtspunkten zu untersuchen, wäre allerdings zu kurz gegriffen: Es ist etwas vollkommen anderes, ob jemand möglichst schnell und effektiv hundert Listen für einen Vorgesetzten ausdrucken muss, oder eine unterhaltsame und dennoch anspruchsvolle Methode sucht, seine Freizeit zu verbringen. Zudem stellt sich die Interaktion in Computerspielen nicht einheitlich dar, sondern gliedert sich in die drei Ebenen „Game Interface“, „Game Mechanics“ und „Gameplay“. Jeder dieser Teile umfasst andere Aspekte der Interaktion und trägt auf seine Weise zum Gesamterlebnis Computerspiel bei. So entspricht die Interaktion auf den Ebenen der Benutzungsschnittstelle (Game Interface) und der Spielmechanik (Game Mechanics) weitestgehend der Benutzung einer Anwendungssoftware. Demzufolge profitieren digitale Spiele in diesen Berei-

chen durchaus von einer gebrauchstauglichen Gestaltung. Eine Benutzungsoberfläche, die das Speichern des Spielstands benutzerfreundlich gestaltet, oder eine Funktion zum einfachen „Schnellreisen“ innerhalb der Spielwelt tragen eindeutig zum Spielspaß bei und veranschaulichen, dass die klassische Usability zumindest einen Teil der Playability ausmacht. In Bezug auf das Gameplay, das Zusammenspiel von Herausforderungen, Belohnungen, Spielwelt und Geschichten, zeigt sich jedoch, dass „Spielbarkeit“ und Gebrauchstauglichkeit keineswegs deckungsgleich sind. Denn während man Anwendungssoftware zugleich effektiv, effizient und zufriedenstellend gestalten kann, laufen Effektivität und Effizienz bei Computerspielen in vielen Fällen der Zufriedenstellung zuwider. So können Game Interface und Game Mechanics gerade mittels einer gebrauchstauglichen Gestaltung das eigentliche „Spielen des Spiels“ zunichtemachen, indem sie das Gameplay all seiner Hindernisse und Herausforderungen und somit seines Reizes berauben. Bei der Betrachtung der Playability sollte demnach stets die Zufriedenstellung im Zentrum stehen. Dies bedeutet, dass dem Gameplay Vorrang vor Game Interface und Mechanics eingeräumt werden muss, da hier das eigentliche Spiel stattfindet.

Die Playability ergibt sich somit zum einen aus der Usability, einer effektiven und effizienten Bedienbarkeit, und zum anderen aus einem zufriedenstellenden Gameplay, den Herausforderungen, den Belohnungen, dem Tempo, der Spielwelt und der Geschichte. Wie und in welchem Ausmaß diese Aspekte die Playability beeinflussen, unterscheidet sich jedoch von Spieler zu Spieler und Spiel zu Spiel. So gibt es zum einen verschiedene Spielertypen mit unterschiedlichen Vorlieben. Zum anderen haben Genre und Hardwareplattform des jeweiligen Computerspiels großen Einfluss auf seine Playability.

Genau wie die Usability kann die Playability also nur unter Berücksichtigung des Nutzungskontexts beurteilt werden. Angelehnt an die Definition der Gebrauchstauglichkeit wird die Playability hier somit wie folgt definiert: „Die Playability eines Computerspiels ist das Ausmaß an Spielspaß und Gebrauchstauglichkeit, mit dem ein Computerspiel durch bestimmte



Bild: Obgleich sie sowohl wirtschaftlich als auch gesellschaftlich von immer größerer Bedeutung sind, gelten Computerspiele noch immer als Zeitvertreib oder gar -verschwendung.

Spieler in einem bestimmten Nutzungskontext gespielt werden kann.“

2.2 Spielerzentriertes Design

Wie aber lässt sich die Theorie der Playability in der Praxis der Spielgestaltung umsetzen? Der weit verbreitete „Designerzentrierte“ Gestaltungsansatz birgt die Gefahr, an der Zielgruppe vorbei zu gestalten und Probleme der Playability, die vor allem unerfahrene Spieler betreffen könnten, zu übersehen. Game Designer sollten sich stattdessen der MCI-Philosophie des „User-centered Designs“ bedienen. Im Gegensatz zum Designerzentrierten Ansatz steht beim UCD der Benutzer im Mittelpunkt. Indem er Rückmeldung über die zu entwickelnde Software beisteuert, hilft er dem Entwickler, die Anwendung gebrauchstauglich zu gestalten. Game Designer könnten auf diese Weise ein höheres Maß an Playability sicherstellen. Ein entsprechendes „Spielerzentriertes Design“ sollte sich dabei wie das UCD in die Phasen Nutzungskontextanalyse, Definition der Anforderungen, Konzeption und Entwurf sowie

Evaluation gliedern und auf Methoden und Techniken der MCI zurückgreifen.

So sollten Spielegestalter im Zuge des Spielerzentrierten Designs zunächst Informationen über Spieler und Nutzungskontext einholen und auf dieser Basis die Anforderungen an ihr Computerspiel definieren. Da ein „Musterspieler“ nicht existiert, sollten Spieleentwickler zunächst mittels Befragungen oder Fokusgruppen Daten über die Zielgruppe erheben, diese analysieren und in Nutzerprofilen zusammenfassen. Unter weiterer Berücksichtigung von Genre und Hardwareplattform des Spieleprojekts lassen sich so direkte Anforderungen an das Game Design ableiten.

Um die Anforderungen der Spieler verifizieren und präzisieren zu können, müssen die bislang ermittelten Anforderungen in der dritten Phase des iterativen Prozesses in Konzepte umgesetzt werden. Rein beschreibende Entwürfe, wie etwa Design-Dokumente, reichen dabei jedoch nicht aus: Möchte man die Playability eines Computerspiel beurteilen, muss man es auch tatsächlich

spielen. Das „Prototyping“, eine klassische Methode des benutzerzentrierten Produktdesigns, stellt zunächst ebenfalls nur einen Entwurf dar, bietet jedoch gegenüber rein beschreibenden Konzepten den Vorteil, dass ein Prototyp tatsächlich ausprobiert werden kann. Für die Evaluation von Computerspielen eignen sich schon simple „Low Fidelity“-Prototypen aus Papierzetteln, Karton und Haushaltsgegenständen, die gerade wegen ihrer Einfachheit von hohem Nutzen für das Game Design sind (Fullerton et al., 2004). Ebenfalls kann auf „High Fidelity“-Prototypen zurückgegriffen werden: Diese vorläufigen Spielversionen kommen dem Endprodukt bereits sehr nah, ohne zwangsläufig einen hohen Produktionsaufwand zu erfordern. Prototyping ist eine Grundvoraussetzung des Spielerzentrierten Designs, da es die Evaluation der Playability erst ermöglicht.

Auch in der Evaluationsphase des Spielerzentrierten Designs können und sollten Spielegestalter auf Techniken der MCI vertrauen. So können gut geplante Usability-Tests Hinweise auf kritische



Bild: Computerspiele bieten effiziente Methoden, mit denen sie ihre Benutzer langanhaltend zu motivieren wissen – ein Aspekt, der auch in Anwendungssoftware eine große Rolle spielt.

Probleme in einem Computerspiel geben. Für die Messung der Empfindungen der Spieler, eine Schwäche der Usability-Tests, eignet sich wiederum die Technik der Befragung. In Kombination mit einem Testspiel kann sie als sogenannter „Playtest“ die Empfindungen der Spieler in Form konkreter Zahlen zugänglich machen (Davis et al., 2005).

Neben diesen empirischen Methoden eignet sich auch die analytische Technik der heuristischen Evaluation für Computerspiele. Beim Vergleich mit klassischen Usability-Tests erwies sich diese Evaluation anhand einer Liste von Game Design-Heuristiken als eine valide Methode, welche die empirische Evaluation zwar nicht ersetzen, sie aber gerade zu Beginn des Designprozesses sinnvoll ergänzen kann (Desurvire et al., 2004). Die Evaluationsmethoden der MCI können Game Designer somit in hohem Maße unterstützen, vorausgesetzt sie berücksichtigen die Andersartigkeit der Computerspiele sowie die Eigenheiten der unterschiedlichen Methoden.

3. Was Anwendungssoftware von Computerspielen lernen kann

Auch Forscher der MCI und Entwickler von Anwendungssoftware haben den Computerspielen bislang nur wenig Aufmerksamkeit geschenkt. Dabei bieten Computerspiele effiziente Methoden, mit denen sie ihre Benutzer langanhaltend zu motivieren wissen – ein Aspekt, der auch in Anwendungssoftware eine große Rolle spielt. Darüber hinaus finden sich in digitalen Spielen auch innovative Design-Konzepte, die schlicht der Gebrauchstauglichkeit zugutekommen.

3.1 Spielend arbeiten

Traditionsgemäß stellt die MCI bei der Betrachtung interaktiver Systeme die reine Zweckmäßigkeit eines Produkts in den Mittelpunkt. Neuere Ansätze der MCI treten mittlerweile für eine neue Sichtweise auf interaktive Systeme ein, die über Nützlichkeit und Benutzbarkeit hinausgeht. Marc Hassenzahl etwa spricht von „hedonischen Qualitäten“

eines Softwareprodukts. Herausforderung an die Softwaregestalter sei es, „eine Erleichterung der Bedienung zu erreichen, ohne langweilig zu werden“. So entstehe ein „attraktives Produkt“, das nicht nur Zufriedenheit, sondern sogar Freude bei seinen Benutzern auslöse (Hassenzahl, 2003).

Gute gestaltete Computerspiele schaffen es, den Spielern alle nötigen Fähigkeiten beizubringen, ohne sie dabei zu langweilen. Besser noch: Sie motivieren sogar beim Lernen. Möglich wird dies durch didaktische Mittel, welche die Spieler nicht als Belehrung wahrnehmen. Dies sind zum einen die Tutorials, Einführungen in den Grundlagen des Spiels, die idealerweise bereits Teil des Spielgeschehens sind und die Spieler aktiv ihre eigenen Lernerfahrungen machen lassen, statt sie mit Vorträgen zu belehren. Auch in Anwendungssoftware könnten Tutorials dieser Art den Einstieg erleichtern. Zum anderen tragen in Computerspielen Herausforderungen zum Lernen bei: Da die Spieler mit jeder bestandenen „Challenge“ ihre Fähigkeiten fortentwickeln, müssen die Her-

ausforderungen immer anspruchsvoller werden – auf diese Weise treiben sie den Lernprozess an. Anwendungssoftware kann sich dieses Prinzip zu Nutze machen, um mit fairen Herausforderungen einen motivierenden Lernprozess in Gang zu setzen. Das könnte in Form von „Multi-Layer Interfaces“ geschehen, die mit immer komplexeren Interface-Schichten nicht nur einen Kompromiss zwischen Zugänglichkeit und Leistungsvermögen ermöglichen würden. Die Benutzer könnten auch – wie in einem Spiel – Selbstachtung und Freude daraus ziehen, immer anspruchsvollere Ebenen der Software zu meistern. Darüber hinaus könnten konkrete „Challenges“ in Form spielerischer Lernangebote den Benutzern neue Funktionsbereiche der Software näher bringen.

Geschichten, Mikrowelten und Belohnungen

Ein weiteres Game Design-Prinzip, von dem die MCI profitieren kann, stellen Geschichten dar, die in Computerspielen einen großen Beitrag zum Spielspaß leisten, indem sie unzusammenhängende Spielelemente zu einem großen Ganzen verbinden. Genau diesen Zusammenhang könnten Softwaregestalter laut Hassenzahl auch in „ernsthafter“ Software herstellen. So wäre in seinen Augen etwa eine Textverarbeitung mit „storytelling“-Funktionen denkbar, in der ein Assistent dem Benutzer Mut macht und ihm Ratschläge gibt, ihn an bisherige Erfolge und „dramatische“ Abschnitte vergangener Arbeitssitzungen erinnert und schließlich „eine Zusammenfassung der schwierigsten Momente, der dramatischsten Schreibblockaden und ihrer Lösungen“ bietet (Hassenzahl, 2003). Nach Hassenzahl lassen sich Erzählungen somit durchaus auf eine Art und Weise in Softwareprodukte integrieren, welche die Gebrauchstauglichkeit nicht beeinträchtigt und den Benutzer zusätzlich motiviert.

Ein Grund für die Faszination vieler Computerspiele sind die von ihnen entworfenen Spielwelten, die das nackte Regelwerk mit Fantasie ausschmücken und so zusätzlich motivieren. Die Kohärenz spielt dabei eine wichtige Rolle: In sich schlüssige fiktionale Welten sind glaubwürdiger und erleichtern das

Eintauchen in das Spielerlebnis. Game Designer legen daher viel Wert auf die spielerische und visuelle Konsistenz ihrer „Mikrowelten“: Spieler können einmal erlernte Techniken im gesamten Spiel anwenden, und sämtliche Elemente des Spiels wirken wie „aus einem Guss“ (Hassenzahl, 2003). Viele Spielgestalter machen zudem nicht bei Grafik und Gameplay halt, sondern betten auch Spielelemente, die eigentlich nicht zur Spielwelt gehören, in die Mikrowelt ein. Anwendungssoftware könnte von diesen Bemühungen um Konsistenz und Aufmerksamkeit für Details lernen und so an Attraktivität gewinnen. So bemühen sich Usability-Experten zwar redlich um Konsistenz im Sinne der „Erwartungskonformität“, verwechseln dabei aber häufig Konsistenz mit Uniformität (Hassenzahl 2003). So kann Anwendungssoftware durchaus visuell neue Wege, jenseits des typischen Windows-Designs, beschreiten. Solange sie dabei in sich konsistent bleibt, widerspricht sie nicht der Erwartungskonformität sondern wird für den Nutzer attraktiver.

Alltäglich greifen Menschen auf Belohnungen zurück, um ihre Mitmenschen in bestimmten Verhaltensweisen zu bestärken. Auch Game Designer wissen um dieses Prinzip und setzen gezielt auf extrinsisch motivierende Designelemente, um ihre Spieler zu dauerhaftem Spielen zu motivieren. Für bestandene Herausforderungen belohnen sie die Spieler mit aufwendigen Animationen, lobenden Worten, Punkten, neuen Fähigkeiten oder weiteren Spielabschnitten. Hinzu kommt die Komponente des Wettbewerbs: Die Highscore-Liste anzuführen, einen höheren Rang zu besitzen oder mehr „Achievements“ gesammelt zu haben als ihre Mitspieler birgt für viele Spieler eine große intrinsische Motivation. Auch diese Erfahrungen könnte sich die MCI zu Nutze machen, um Anwendungssoftware attraktiver zu gestalten. Ob Auszeichnungen, neue Funktionen oder ein simples „Gut gemacht!“, die denkbaren Einsatzmöglichkeiten von Belohnungen in Anwendungssoftware sind vielfältig. Und auch die Komponente des Wettbewerbs ließe sich – dank der immer stärkeren Vernetzung von Arbeitsplätzen und interaktiven Systemen – ohne weiteres in

Anwendungssoftware integrieren. Belohnungen und Wettbewerb stellen somit weitere Aspekte des Game Designs dar, die nicht nur zum Spielspaß eines Computerspiels, sondern auch zur Freude an der Nutzung von Anwendungssoftware beitragen können.

3.2 Innovationen aus der Welt der Computerspiele

Computerspiele setzen schon immer Techniken ein, welche für eine möglichst reibungslose und effiziente Interaktion der Spieler mit dem Computerspiel sorgen sollen (vgl. hierzu: Dyck et al. 2003). Eine dieser Techniken ist das Einbinden der Online-Community: Anstatt sie wie Anwendungssoftware außen vor zu lassen, integrieren Mehrspieler-Spiele die natürliche Gemeinschaft der Anwender bzw. Spieler. Dies stellt sicher, dass die Nutzer am selben Ort zusammenkommen, und gewährleistet damit eine „kritische Masse“ an Nutzern. Da Softwareanwender genau wie Computerspieler Erfahrungen austauschen, Fragen stellen und Rückmeldung bekommen möchten, könnten Softwareentwickler dieses Prinzip aufgreifen. Das Potential wäre enorm: Auf der großen Kundenbasis von Microsoft Word ließe sich etwa eine Online-Community mit Millionen von Mitgliedern aufbauen. Diese könnten ohne Zeitverlust miteinander kommunizieren, Probleme lösen und von erfahreneren Nutzern lernen.

Eine weitere Technik, die sich Anwendungssoftware von Computerspielen abschauen kann, ist das Lernen durch Zuschauen. Diese effektive Methode, die Menschen von der Geburt an nutzen, um sich neue Fähigkeiten anzueignen, findet sich in vielen Mehrspieler-Spielen wieder. Diese unterstützen ihre Spieler beim Erlernen des Spiels, indem sie sie ihren Mitspielern virtuell über die Schulter schauen lassen. Lernen durch Zuschauen ließe sich auch in Anwendungssoftware angemessen umsetzen, würde man den Nutzern das Beobachten anderer Anwender erlauben, sie dabei mit detaillierten Informationen über deren Vorgehen versorgen und ihnen damit die Interpretation der Aktionen erleichtern.

Der hohe Grad an Anpassbarkeit, das Verändern und Erweitern der Benutzungsoberfläche durch die Spieler, ist eine weitere Innovation der Computerspiele, von der Anwendungssoftware lernen kann. Viele Spielegestalter halten es für elementar, den Spielern die Möglichkeit zu geben, Layout und Abbildung von Spielfunktionen zu verändern, die bestehende Benutzeroberfläche zu modifizieren und zu erweitern, und die von ihnen entworfenen Interfaces ihren Mitspielern zugänglich zu machen. Diese Techniken tragen zur Playability bei, indem sie dem effizienten Erfüllen wiederkehrender Aufgaben und somit der Zufriedenstellung der Spieler dienen. Dies sollten auch die Ziele jeder Anwendungssoftware sein. Genau wie Computerspiele den Spielern wechselnde Spielweisen abverlangen, erfordern in Anwendungssoftware spezifische Aufgaben unterschiedliche Vorgehensweisen der Nutzer. Ein hoher Grad an Anpassbarkeit würde diesem Umstand Rechnung tragen.

Flüssige System-Mensch-Interaktion

Softwareentwickler können außerdem von Methoden der Kommunikation lernen, die in digitalen Spielen für eine flüssige System-Mensch-Interaktion sorgen. Da die größte Herausforderung für Spieler nach wie vor ihre begrenzte Aufmerksamkeitskapazität ist, reduzieren diese Techniken die von den Spielern abverlangte Aufmerksamkeit auf ein Minimum. So werden Nachrichten unaufdringlich mittels Sprache, Geräuschen, Textmitteilungen und Animationen übermittelt, ohne eine explizite Bestätigung durch die Spieler zu erfordern und den Spielfluss zu unterbrechen. Teile der Benutzungsoberfläche verändern sich abhängig von der Aufmerksamkeit, die ein Spieler ihnen schenkt, und werden etwa transparent, und Spiele passen den Blickwinkel auf die Spielwelt an die jeweilige Spielsituation an. Auch die Effizienz von Anwendungssoftware ließe sich auf diese Weise nachhaltig steigern: Setzen Softwareentwickler für nichtkritische Nachrichten statt der allgegenwärtigen Dialogfenster die unaufdringlichen Methoden der Computerspiele ein, würden sie Zeit und Aufmerksamkeit der Benutzer weniger beanspruchen. Aufmerksamkeits-

sitive Interfaceelemente würden Nutzern das zeitraubende Öffnen, Schließen, Verschieben und Skalieren von Fenstern ersparen. Und auch die kontextabhängige Darstellung wäre in Anwendungssoftware der Effizienz zuträglich, da sie den für das Ändern der Ansicht und das Navigieren nötigen Aufwand reduzieren würde.

Sicherlich ließen sich die vorgestellten Innovationen aus der Welt der Spiele nicht vollkommen ohne Probleme in Anwendungssoftware einbinden und bedürfen genauerer Untersuchungen bezüglich Fragen u.a. der Ablenkung, des Datenschutzes oder der Sicherheit. Diese potentiellen Risiken stellen aber keinesfalls unüberwindbare Hürden dar. Alle Methoden lassen sich grundsätzlich auf Anwendungssoftware übertragen und sind in der Lage, die Interaktion zwischen System und Anwendern flüssiger zu gestalten und die Produktivität zu steigern. Softwareentwickler müssen lediglich sorgfältig prüfen, wo und

wann sie welches Konzept angemessen einsetzen können.

4. Fazit

Für Game Designer, Vertreter der MCI und Entwickler von Anwendungssoftware ist es an der Zeit umzudenken. Die bislang unterschätzten digitalen „Spielzeuge“ können durchaus einen konstruktiven Beitrag zur Gestaltung interaktiver Systeme leisten, und die als zu theoretisch geltenden Erkenntnisse der Wissenschaft sehr wohl zum Spielspaß beitragen. Sicherlich gibt es viele Unterschiede zwischen Computerspielen und konventioneller Software. Die Differenzen dürfen Game Design und Mensch-Computer-Interaktion aber nicht davon abhalten, sich einander zu öffnen. Ansonsten würden beide Seiten das große Potential verschenken, das einem Brückenschlag zwischen den Disziplinen innewohnt.



Volker Stuckmann

ist Junior Game Designer bei der Blue Byte GmbH in Düsseldorf. Er studierte Medienwissenschaft mit dem Schwerpunkt Medieninformatik an der Universität Paderborn. Seine Diplomarbeit (eingereicht 2008), deren Ergebnisse hier zusammengefasst sind, befasste sich mit der Frage, was Game Design und Mensch-Computer-Interaktion voneinander lernen können.

E-Mail: volker@stuckmann.com

Literatur

- Davis, J. P.; Steury, K. R.; Pagulayan, R. J.: A survey method for assessing perceptions of a game: The consumer playtest in game design, 2005. http://www.gamestudies.org/0501/davis_steury_pagulayan/ (Letzter Zugriff: 31.12.2009).
- Desurvire, H. W.; Caplan, M.; Toth, J. A.: Using Heuristics to Evaluate the Playability of Games. In: CHI '04 extended abstracts on Human factors in computing systems. New York: ACM, 2004.
- Dyck, J.; Pinelle, D.; Brown, B.; Gutwin, C.: Learning from Games: HCI Design Innovations in Entertainment Software. In: Graphics Interface Proceedings. (Hrsg. Moller, T.; Ware, C.) Wellesley: A K Peters, 2003.
- Fullerton, T.; Swain, C.; Hoffman, S.: Game Design Workshop – Designing, Prototyping, and Playtesting Games. San Francisco: CMP Books, 2004.
- Hassenzahl, M.: Attraktive Software – Was Gestalter von Computerspielen lernen können. In: User Interface Tuning. Benutzungsschnittstellen menschlich gestalten. (Hrsg. Machate, J.; Burmester, M.) Frankfurt: Software & Support Verlag, 2003.