

Digitale Skripte mit Markdown und elearn.js als Basistechnologie für OER

Zusammenfassung

An deutschen Hochschulen werden in zahlreichen Lehrveranstaltungen Skripte zur Wissensvermittlung genutzt und meist den Studierenden als PDF zum Download angeboten. Digitale Skripte ermöglichen es, die Vorteile von elektronischen Lehrmedien mit denen eines papierbasierten Veranstaltungsskriptes zu vereinen. Die Meta-Sprache Markdown dient hierbei als Autorensprache zur Generierung verschiedener Ausgabeformate wie PDF, HTML oder ePUB. Der Vorteil von Markdown liegt in dem einfachen und quelloffenen Format, welches sich problemlos teilen und adaptieren lässt, ohne spezielle Editoren oder Plattformen vorauszusetzen. Das frei lizenzierte elearn.js ermöglicht es, die HTML-Ausgabe von Markdown zu paginieren und mit Interaktionsmöglichkeiten auszustatten, wie zum Beispiel Navigation per Maus, Tastatur oder Wischgeste, Inhaltsverzeichnis, Bildergalerie, Quiz und weitere. Das dazugehörige Stylesheet passt die Darstellung der Inhalte responsive an alle geläufigen Bildschirmgrößen an. Markdown und elearn.js eignen sich somit bestens zur nachhaltigen Entwicklung von digitalen, offenen Lernressourcen.

1 Skripte in der alltäglichen Hochschulpraxis

Veranstaltungsskripte werden an Hochschulen meist begleitend zu Präsenz-Lehrveranstaltungen für die Teilnehmenden als Lernmaterial bereitgestellt, zum Beispiel als Vorlesungsskript oder Praktikumsskript. Die Skripte bieten den Studierenden in der Regel lehrbuchartige Texte zur Vor- oder Nachbereitung der Lehrveranstaltung, die von den DozentInnen inhaltlich speziell auf diesen Bedarf zugeschnitten werden. Dies geschieht häufig in Inhaltsbereichen, für die es kein passendes Lehrbuch gibt oder in denen die verfügbaren Lehrtexte zu umfangreich oder zu verstreut vorliegen, um sie in der gegebenen Zeit bearbeiten zu können. Ein Handzettel-Ausdruck von Präsentationsfolien stellt kein vollwertiges Skript dar, da die Folien ohne das gesprochene Wort nicht den gesamten Inhalt widerspiegeln. Aus manchen Vorlesungsskripten sind im Laufe der Jahre Lehrbücher entstanden (vgl. Skriptum, Wikipedia).

Skripte werden in der Regel in einem Office-Programm wie MS Word oder LibreOffice Writer verfasst. In technischeren Fächern ist auch die Nutzung

des Textsatzsystems LaTeX verbreitet. Für die Bürokommunikation und einfache Drucksachen sind Office-Programme sehr gut geeignet. Ist jedoch das Ziel, auch im Internet zu veröffentlichen, macht die Überführung der Inhalte aus einem Textverarbeitungsprogramm in HTML unnötig viel Arbeit. Aus LaTeX ist die Konvertierung nach HTML problemlos möglich, doch die meisten Autoren scheuen die große Anfangshürde beim Erlernen der LaTeX-Syntax und Aufsetzen des Satzsystems.

Die Möglichkeiten, die sich bieten, wenn Skripte vornehmlich digital als HTML-Seiten veröffentlicht werden, sind vielfältig. Vor allem die Einbindung von dynamischen und interaktiven Medien sind hier hervorzuheben (vgl. Skript, eTeaching.org). Schon länger ist man im E-Learning auf der Suche nach einem geeigneten Ansatz, um komfortabel Online- und Print-Versionen von Lehrtexten aus einer Quelle generieren zu können. Doch entweder sind die Autorenumgebungen zu technisch und schwierig zu erlernen (z.B. eLML), oder sie setzen auf WYSIWYG-Editoren, die leider nicht in der Lage sind, wirklich sauberen HTML-Code zu generieren. Einige Wiki-Engines sind zwar in der Lage, Artikel auch als PDF auszugeben, doch werden die Autoren dadurch beschränkt, dass sie nur im Browser online arbeiten können und auf eine Installation eines solchen Wiki-Servers angewiesen sind.

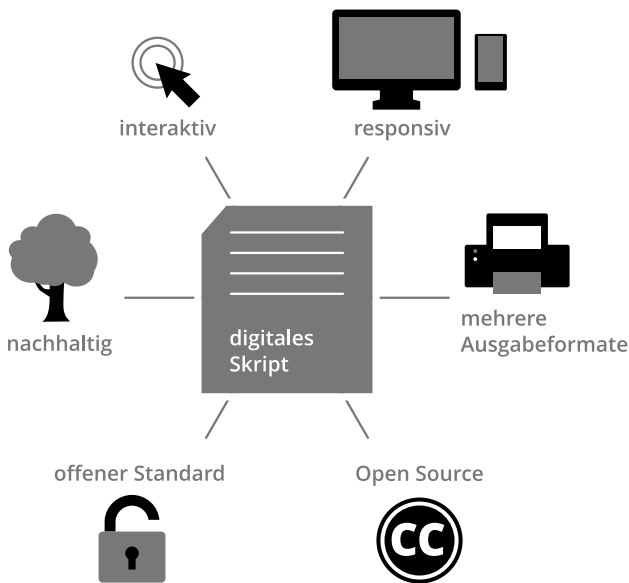


Abb. 1: Eigenschaften von zukunftsfähigen Digitalen Skripten.

2 Anforderungen an offene digitale Skripte

Soll das Skriptum in ein digitales zukunftsfähiges Format gebracht werden, so können verschiedene Anforderungen daran gestellt werden (vgl. Abb. 1). Die Anforderungen sind zum einen aus der Tatsache der steigenden Mobilität und dem einhergehenden Mobile Computing (Horizon Report 2011, JIM-Studie 2015), zum anderen aus den praktischen Erfahrungen mit E-Learning-Standards und den Problemen bei der Wiederverwendung von OER abgeleitet (Amiel, 2013, Tabuenca, 2015).

Folgende Anforderungen werden an die digitalen Skripte gestellt:

- einfach und überall zu erstellen
- ohne speziellen Editor
- in offenem Quell-Format
- leicht weiterzugeben, anzupassen und zu überarbeiten
- mit interaktiven und dynamischen Medien
- in Webseiten und in Lernplattformen leicht integrierbar
- mit druckbarer Version
- auf allen gängigen Gerätetypen gut zu nutzen
- leicht und intuitiv zu navigieren
- offline nutzbar
- mit möglichst wenig Abhängigkeiten von externen Ressourcen
- möglichst kleine Dateigrößen, dadurch mobil besser nutzbar

Die Probleme gängiger E-Learning-Autorenworkflows werden anhand dieser Liste schnell deutlich. Vor allem die flexible Erstellung und Anpassung der Skripte ohne speziellen Editor sind herausstechende Merkmale, die offene Lernressourcen tragen sollten. Eine Basistechnologie hierzu bietet Markdown.

3 Entwicklung von Texten mit Markdown

Markdown ist eine Meta-Auszeichnungssprache, die sich bei der Auszeichnung von Textabschnitten sehr einfacher Markierungen bedient (Gruber, 2004). Dabei wird Markdown im Quelltext in einer einfachen Textdatei geschrieben. Die Art und Weise der Auszeichnung orientiert sich dabei an der gewachsenen Praxis, wie solche Markierungen beispielsweise auch in Text-E-Mails vorgenommen werden. Dabei ist das Ziel, den Quelltext weiterhin leicht schreib- und lesbar zu halten, und dennoch Textabschnitte eindeutig und leicht verständlich auszuzeichnen. Ursprünglich wurde Markdown entwickelt, um HTML-Inhalte effizient schreiben zu können, ohne einen WYSIWYG-Editor nutzen zu müssen. Das direkte Schreiben einer Markup-Sprache wie HTML wird als zu fehleranfällig und störend für den Schreibprozess angesehen. Dieser Ansatz wurde in letzter Zeit auch

für verschiedene Schreibprogramme aufgegriffen, um mit Hilfe von Markdown eine möglichst ablenkungsfreie Umgebung für konzentriertes Arbeiten am Text zu ermöglichen (writeMonkey, Byword).

Die erste Markdown-Implementierung bestand aus einem Pearl-Skript, welches die Auszeichnungen im Markdown in HTML-Tags umwandelt (Gruber, 2004). Nach und nach folgten weitere Implementierungen mit Erweiterungen des Funktionsumfangs und der Ziel-Formate (Multimarkdown, GitHub, Pandoc). Markdown wird nicht mehr nur als Mittel zum Schreiben für das Internet angesehen, sondern als Sprache für das konzentrierte Schreiben im Allgemeinen, auch für Printmedien oder wissenschaftliche Arbeiten. Tools zur Umwandlung von Markdown in PDFs oder in LaTeX machen das strukturierte Vorbereiten von für den Druck bestimmten Texten mit Markdown möglich. Ebenso ist das Generieren von E-Books im EPUB-Format möglich. Um die verschiedenen Dialekte von Markdown wieder zu vereinheitlichen und um Uneindeutiges der ursprünglichen Markdown-Implementation zu klären, wurde mit CommonMark eine Standardisierung von Markdown angestrebt (CommonMark).

Markdown stellt damit eine etablierte Lösung zum konzentrierten Schreiben dar, welches verschiedene Ausgabeformate wie HTML, PDF und EPUB ermöglicht. Dabei ist für Markdown kein spezieller Editor nötig, ein einfacher Quelltext-Editor reicht aus. Die Konvertierungsprogramme stehen unter freien Lizenzen, daher bietet sich Markdown insbesondere als Quellformat für offene Lernressourcen an.

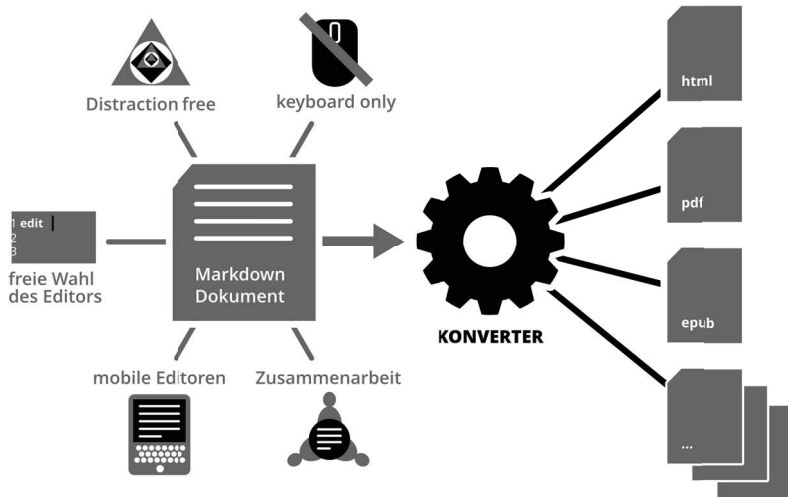


Abb. 2: Eigenschaften von Markdown als Autorensprache.

4 *elearn.js* – OpenSource-Software für Interaktionen

Als Basis für digitale Skripte kann also ein Markdown-Dokument genutzt werden. Aus diesem werden verschiedene Formate generiert, zur Anzeige auf einem Bildschirm vorzugsweise HTML. Die HTML-Ausgabe kann nun mit Inhalten angereichert werden, die nur bei der Anzeige auf einem Bildschirm genutzt werden können, vor allem dynamische und interaktive Medienelemente wie Videos, Animationen, interaktive Grafiken oder Quiz-Fragen. Auch die Navigation durch den HTML-Content sollte optimiert werden. Neben der langen Scroll-Seite wird eine seitenweise Darstellung des Inhalts angestrebt. Die Navigation innerhalb der Seiten sollte auch über eine Inhaltsübersicht möglich sein. Das Aussehen der HTML-Inhalte sollte möglichst optimal das Lesen unterstützen und, angelehnt an PDF- oder eBook-Reader, zurückgenommen sein und nicht ablenken. Schließlich soll der Inhalt auf möglichst vielen Endgeräten mit unterschiedlichen Screen-Größen funktionieren und gut lesbar sein (Zillgens, 2013).

Um all diese Anforderungen und Aufgaben zu erfüllen, wurde das Java-Script *elearn.js* entwickelt, welches für digitale Skripte den Inhalt um verschiedene Interaktionsmöglichkeiten erweitert und die Navigation durch den Inhalt optimiert. Dazu passend wurde das responsive Stylesheet *elearn.css* entwickelt. Wie in Abbildung 3 zu sehen ist, wird derselbe Inhalt auf stark unterschiedlichen

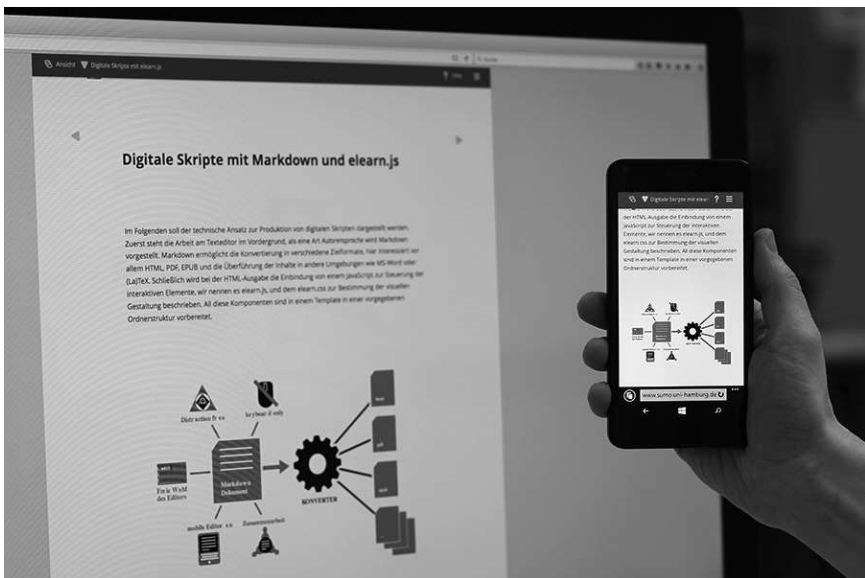


Abb. 3: Ein digitales Skript auf einem 27“-Bildschirm und einem Smartphone.

Bildschirmgrößen angepasst und gut lesbar angezeigt. Folgende Eigenschaften werden mit *ellearn.js* und *ellearn.css* ermöglicht:

- Dynamische Medien werden mittels HTML5-Tags in das Skript eingebunden.
- Das HTML kann in Sektionen unterteilt und so auf mehreren ‚Seiten‘ angezeigt werden (Paginierung).
- Zwischen den Seiten kann über Pfeiltasten, per Tastatur oder Wischgeste navigiert werden.
- Ein Inhaltsverzeichnis bietet die Navigation anhand einer Seiten-Übersicht.
- Ein Side-Menü bietet zusätzliche Funktionen wie Drucken, QR-Code und Download der Quelldateien.
- Listen von Abbildungen können als Slideshow angezeigt werden.
- Mit dem optionalen *quiz.js* sind Forced-Choice-, Multiple-Choice- und Short-Answer-Fragen möglich.
- Mit dem optionalen *clickimage.js* sind interaktive Grafiken möglich.
- Das Layout passt sich in vier Stufen responsiv an die Screen-Größe an.
- Ein Print-css bietet einen sauberen Druck aus dem Browser.

Ein nach HTML konvertiertes Markdown-Dokument kann demnach mit *ellearn.js* und den dazu gehörigen Komponenten in ein interaktives digitales Skript umgewandelt werden (vgl. Abb 4). Zusätzliche Inhalte wie Videos, Quiz-Fragen, interaktive Grafiken oder Galerien können bereits auch schon in das Markdown-

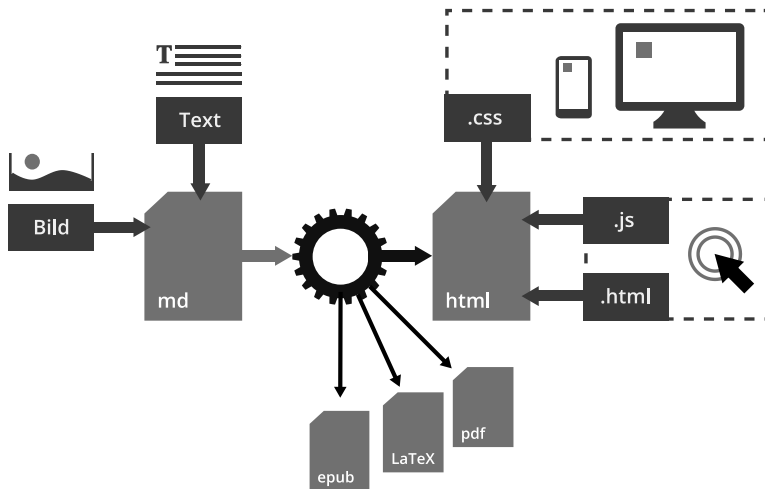


Abb. 4: Zusammenspiel der Komponenten bei digitalen Skripten.

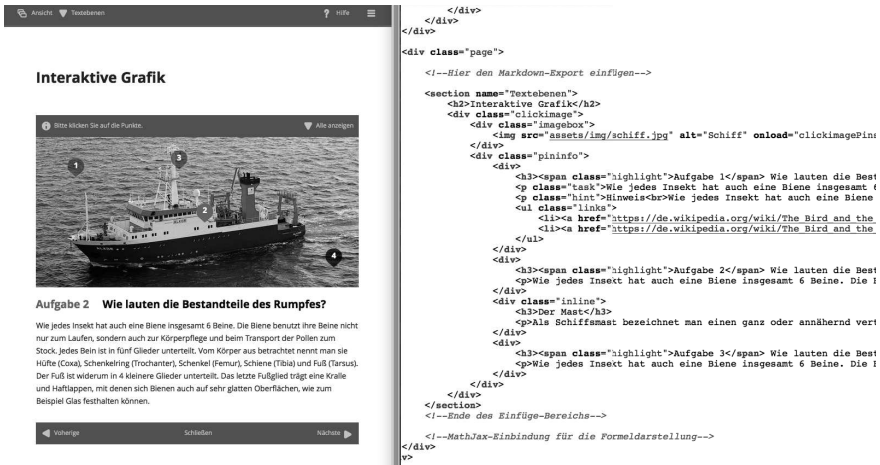


Abb. 5: Interaktive Grafik (Dummy) mit zugehörigem HTML-Code.
Die Steuerung der Interaktionen wird durch das *clickimage.js* gesteuert.

Dokument eingepflegt werden. Diese zusätzlichen Inhalte werden in relativ einfachem HTML-Code eingefügt, der Konverter reicht die HTML-Inhalte dann unverändert in die Ausgabe durch. Die Skripte *quiz.js* und *clickimage.js* erweitern den Funktionsumfang von *elearn.js* und ermöglichen es, mit einfachem HTML-Code interaktive Bausteine zu generieren (siehe Abbildung 5).

Der Ansatz des *elearn.js* in Kombination mit Markdown vereint viele Möglichkeiten von GitBook (friendcode), Adapt Learning (adaptlearning.org) oder hp5 (h5p.org). Das *elearn.js* benötigt jedoch, im Gegensatz zu den genannten alternativen Lösungen, keinerlei Server-Komponenten. Der Vorteil liegt bei *elearn.js* somit darin, dass der Workflow auch für Laien transparent und nachvollziehbar bleibt und der Nutzenden die volle Kontrolle über seine Inhalte behält. Es bestehen bei *elearn.js* fast keine Abhängigkeiten zu anderen Komponenten und die Erzeugnisse sind extrem flexibel anwendbar und erweiterbar, Sackgassen für die nachhaltige Nutzung der Inhalte werden so vermieden.

5 Erprobung des Workflows für digitale Skripte

Der Workflow und das *elearn.js* für digitale Skripte wurden im Rahmen des Teilprojekts *Studier- und Medienkompetenz Online – SuMO* des Universitätskollegs der Universität Hamburg entwickelt. In diesem Teilprojekt wurden bis zum jetzigen Zeitpunkt zehn Online-Tutorials auf dieser Basis entwickelt. Ebenso konnte mit Hilfe der Erweiterung *quiz.js* ein Großteil des Online-Selfassessments *MIN-Check* erstellt werden, welches auch Teil des Universitätskollegs der UHH ist.

Weitere neun digitale Skripte wurden für die Seminare *Visual Interface Design* und *Proseminar Human Computer Interaction* auf Basis des beschriebenen Workflows erstellt. Auch ein Vorlesungsskript aus den Wirtschaftswissenschaften wird zur Zeit als digitales Skript umgesetzt.

Ein neues interessantes Feld für die Erprobung des Ansatzes bietet sich im Rahmen der Hamburg Open Online University (HOOU). Hier konnten schon einige Projekte vom Format der digitalen Skripte überzeugt werden. Insgesamt scheint sich in der HOOU Markdown als ein häufiger genutztes Format zu bewähren.

In Zukunft sollen Daten über die Nutzung des Ansatzes gewonnen werden, um konkrete Aussagen über die Effektivität und Benutzerfreundlichkeit treffen zu können.

6 Diskussion

Die Entwicklung des Formats für digitale Skripte ist soweit, dass es für erste Projekte genutzt wird und damit weitere Erfahrungen und Anforderungen gesammelt werden können. Insbesondere bietet sich das Format für die Entwicklung von OER an, da es vollständig offen ist und ausschließlich lokal zu installierende Open-Source-Software benötigt.

Als problematisch kann angesehen werden, dass das Konzept einer Single-Page nicht für sehr umfangreichen Inhalt geeignet ist, da der Browser die gesamte Seite im Speicher vorhalten muss. Ferner ist der Ansatz eher für linear gestaltete Inhalte sinnvoll, für stark verzweigte oder vernetzte Inhalte wären andere Tools zu Zeit besser geeignet. Schließlich mag es für manche E-Learning-Autoren ungewohnt sein, im Quellcode zu arbeiten. Der Schritt von den WYSIWYG-Editoren zum Texteditor könnte von manchen als rückschrittlich bewertet werden. Der Komfort einer grafischen Benutzerschnittstelle könnte vermisst und die Arbeit im Text als zu fehleranfällig bewertet werden. Dem gegenüber steht beispielsweise die Effizienz bei der Arbeit im Quelltext oder die freie Wahl des Editors. Diese Aspekte sollen in einer zukünftigen Evaluation des Workflows näher betrachtet werden.

Es stehen bereits einige Weiterentwicklungen des *ellearn.js* an, die vor allem die Anwendung für den Endnutzer vereinfachen und zusätzliche Interaktionen ermöglichen sollen.

Geplant sind zunächst:

- Online-Konverter mit automatischer Zusammenstellung aller benötigten Komponenten
- Interaktiver Zeitstrahl zur Vermittlung zeitlicher Abläufe
- Anleitung für Lehrende mit Screencasts, Checklisten, Templates

Langfristig wäre die Integration in ein E-Learning-Repository wünschenswert. Auch könnte das Einbauen von interaktiven Elementen durch einen eigenen Markdown-Dialekt optimiert werden. So würde das Schreiben der teilweise noch etwas umständlichen HTML-Fragmente überflüssig und die Schritte zur Finalisierung eines Skripts könnte noch weiter vereinfacht werden. Ebenso wäre eine Erweiterung der Druckausgabe in Richtung Augmented Textbook (vgl. Chen et al, 2011) oder ein Präsentationsmodus für Vorträge denkbar.

Markdown und elearn.js bieten eine sehr flexible und nachhaltige Basistechnologie, um digitale Skripte und OER allgemein zu entwickeln. Eine funktionsfähige Grundlage ist hierfür bereits geschaffen (Heinecke, 2016). Zukünftige Entwicklungen werden noch weitere Interaktionen möglich machen und die Konvertierung für Endnutzer vereinfachen.

Förderhinweis

Das *elearn.js* wird zum Teil im Rahmen des Teilprojekts „Studier- und Medienkompetenz Online“ des Universitätskollegs der Universität Hamburg entwickelt.

Dieses Vorhaben wird innerhalb des gemeinsamen Bund-Länder-Programms für bessere Studienbedingungen und mehr Qualität in der Lehre aus Mitteln des Bundesministerium für Bildung und Forschung unter dem Förderkennzeichen 01PL12033 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den AutorInnen.

Literatur

- Adapt Learning. (o. J.). Abgerufen 22. Februar 2016, von <https://www.adaptlearning.org>
- Amiel, T. (2013). Identifying Barriers to the Remix of Translated Open Educational Resources. *The International Review of Research in Open and Distributed Learning*, 14(1), 126–144.
- Bleisch, S. & Fisler, J. (2005). *eLesson Markup Language eLML – eine XML basierte Applikation für die beschreibende Auszeichnung von nachhaltigen und flexiblen e-Learning Inhalten*. Muttentz, Schweiz: Fachhochschule beider Basel (FHBB). Abgerufen 22. Februar 2016, von http://www.elml.org/website/en/download/publications/DeLFI2005_eLML_Paper.pdf
- Chen, N.-S., Teng, D. C.-E., Lee, C.-H. & Kinshuk (2011). Augmenting Paper-based Reading Activity with Direct Access to Digital Materials and Scaffolded Questioning. *Computers & Education*, 57(2), 1705–1715.
- CommonMark. (o. J.). Abgerufen 22. Februar 2016, von <http://commonmark.org/>

- e-teaching.org. (o. J.). Skript. Abgerufen 26. Januar 2016, von <https://www.e-teaching.org/lehrszenarien/vorlesung/skript/>
- Feierabend, S., Plankenhorn, T. & Rathgeb, T. (o. J.). *JIM-Studie 2015 – Jugend, Information, (Multi-) Media (Basisstudie zum Medienumgang 12- bis 19-Jähriger in Deutschland)*. Medienpädagogischer Forschungsverband Südwest (mpfs). Abgerufen 20. Februar 2016, von <http://www.mpfs.de/index.php?id=676>
- Friendcode Inc. (2015). GitBook. Abgerufen 18. November 2015, von <https://www.gitbook.com/>
- Gruber, J. (2004). Markdown. Abgerufen 20. Februar 2016, von <https://daringfireball.net/projects/markdown/>
- H5P – Create, share and reuse interactive HTML5 content in your browser. (o. J.). Abgerufen 11. September 2015, von <https://h5p.org>
- Heinecke, M. (2016). *Digitale Skripte*. Abgerufen 22. Februar 2016, von <http://www.sumo.uni-hamburg.de/DigitaleSkripte/>
- Johnson, L., Smith, R., Levine, A. & Haywood, K. (o. J.). *The 2011 Horizon Report*. Austin, Texas: The New Media Consortium. Abgerufen 18. November 2015, von <http://www.nmc.org/publication/nmc-horizon-report-2011-higher-ed-edition/>
- MacFaralaine, J. (2014). Pandoc. Abgerufen 22. Februar 2016, von <http://pandoc.org/>
- MetaClassy. (o. J.). Byword • Simple and efficient text editor for Mac, iPhone and iPad. Abgerufen 22. Februar 2016, von <https://bywordapp.com/>
- pomarancha. (2014). WriteMonkey. Abgerufen 22. Februar 2016, von <http://writemonkey.com/>
- Tabuenca, B. (2015, Januar 26). *10 Limitations of Mobile Authoring Tools with Regard to Universal Access of OER*. Abgerufen 4. Februar 2016, von <http://oer-unescochair-ounl.ning.com/forum/10-limitations-of-mobile-authoring-tools-with-regard-to-universal>
- Wikipedia. (o. J.). Skriptum. Abgerufen 22. Februar 2016, von <https://de.wikipedia.org/wiki/Skriptum>
- Zillgens, C. (2013). *Responsive Webdesign: reaktionsfähige Websites gestalten und umsetzen*. München: Hanser.