

ALS-GameCreator für be-greifbare Miteinander-Lernspiele

Thomas Winkler, David Bouck-Standen, Michael Herczeg

Institut für Multimediale und Interaktive Systeme, Universität zu Lübeck

Abstract

In diesem Paper wird der ALS-GameCreator beschrieben, ein auf Webtechnologien basiertes System zur grafischen Entwicklung von be-greifbaren Miteinander-Lernspielen, der im Rahmen des DFG-geförderten Projekts Ambient Learning Spaces (ALS) entwickelt wird. Ambiente Lernumgebungen sind durch vernetzte digitale Systeme erweiterte Lernumgebungen, in dem die Lernenden ihren Körper im physischen Raum in geräteübergreifender Interaktion einsetzen. Technologisch wird dies durch das Framework Network Environment for Multimedia Objects (NEMO) ermöglicht, welches u.a. die zentrale Haltung der Logik von Lernspielen, eine gemeinsame semantische Datenhaltung, die Konvertierung von Daten sowie die Benutzerauthentifizierung ermöglicht. Anhand zweier Szenarien wird der praktische Einsatz des ALS-GameCreator veranschaulicht, mit dem ohne Programmierkenntnisse Miteinander-Lernspiele in ALS erstellt werden können.

1 Einleitung

Gegenstand unserer bisherigen Forschung sind unter anderem die Miteinander-Lernspiele *Spellit* und *AlgoFrogs*, deren Ziel es ist, bei Kindern im Vorschul- oder Grundschulalter kognitive Fähigkeiten und soziale Interaktion in Problemlösungsszenarien in Kombination mit fein- und grobmotorischen Fähigkeiten in einer realphysischen Umgebung zu fördern (Winkler et al., 2011; Hahn et al., 2012; Scharf et al., 2012). Die Lernspiele wurden in einem Co-Designprozess mit Kindern entwickelt. *AlgoFrogs* (Winkler et al., 2014) fokussiert die Förderung algorithmischen Denkens, während *Spellit* (Winkler et al., 2013; Winkler et al., 2015) dem Erlernen des Lesens und Schreibens im Vor- und Grundschulalter dient. Durch das Einbeziehen von Körperlichkeit und physischem Raum und das be-greifen sowie handhaben von kleinen elektronischen Geräten lernen die Kinder mit besonderer Freude in einem gemeinsamen Lernspiel nachhaltiges Wissen zu konstruieren (Prensky, 2003; Ritterfeld et al., 2009). Beide Lernspiele wurden in der neuesten Entwicklung auf Basis des NEMO-

Frameworks in die *Ambient Learning Spaces* (ALS) im gleichnamigen, von der DFG geförderten Erkenntnistransferprojekt integriert. Das Projekt schließt an zwei gleichnamige, ebenfalls von der DFG geförderte Grundlagen-Forschungsprojekte an, die nacheinander seit 2008 am Institut für Multimediale und Interaktive Systeme (IMIS) der Universität zu Lübeck durchgeführt wurden. Im Rahmen des Projektes wurden AlgoFrogs und Spellit webbasiert für *Tangicons* (Winkler et al., 2014) entwickelt, auch vorübergehend *ELBlocks* (Bouck-Standen et al., 2016) genannt. Tangicons sind kleine elektronische Spielsteine, deren webbasierte Technologie inzwischen ebenfalls auf anderen mobilen Geräten, die über einen Browser verfügen, eingesetzt werden kann. Intelligentes Wissen (Weinert, 2001) wird insbesondere durch komplexe Lernsituationen mit Lebensweltbezüge konstruiert (Höfer & Madelung, 2006). Unter den vorgenannten Gesichtspunkten ergibt sich im schulischen Kontext die Herausforderung, Miteinander-Lernspiele zu entwickeln, in denen sowohl der Lebensweltbezug, wie auch komplexe Lernsituationen abgebildet werden können (Gee, 2003). Um in den dieser Forschung zugrundeliegenden Szenarien, die nachstehend beispielhaft dargestellt werden, nicht lediglich eine bloße Akkumulation von Informationen zu fördern, bedarf es bereits während der Erstellung eines Lernspiels der nachhaltigen Reflektion des Pädagogen und auch das Einbeziehen der Schüler*innen in den Gestaltungsprozess selbst dient diesem Zweck (Huang et al., 2010; Dickey, 2005). Das nachstehende Beispielszenario veranschaulicht die Erstellung eines Levels für das Miteinanderlernspiel AlgoFrogs mit dem ALS-GameCreator.

Beispielszenario für die Levelerstellung

Lena Ergün unterrichtet in einer Grundschule und erstellt für das Miteinanderlernspiel AlgoFrogs mit dem ALS-GameCreator einen neuen Level. Ziel des Spiels ist es, algorithmisches Denken zu fördern (Winkler et al. 2014), indem die Kinder durch gemeinsame Aktionen mit den Tangicons den grünen Frosch über das Spielfeld zur gelben Fröschin bewegen. Nachdem die Kinder im zweiten Jahrgang mit den ersten vier Levels gut zurechtgekommen sind, erstellt sie nun einen fünften Level, siehe Abb. 1. Dabei übernimmt sie den Hintergrund des vorhergehenden Levels und ordnet per Drag-and-Drop Datenblöcke auf dem später beim Spielen nicht sichtbaren Schachbrettmuster an, um den neuen Level zu gestalten. Mit der Platzierung des grünen Froschs per Drag-and-Drop definiert sie den Startpunkt des Levels, während das Erreichen der gelben Fröschin das Ende des Levels bedeutet. Sie definiert innerhalb des Levels im Bedienfeld der Spielelogik (unten links) Ereignisse, wie etwa das Abspielen von Sound, die Bewegung des grünen Froschs (Overlay) oder den Level-Übergang. Dazu wird ein Popup-Menü eingeblendet, in dem sie die dazugehörigen logischen Funktionen definiert. Außerdem definiert sie in einem weiteren Popup-Menü Aktionen, die mit Funktionsblöcken verknüpft werden und auf den Bildschirmen der Tangicons erscheinen, die die Spieler im Spielverlauf ausführen können. Beim Erstellen falsch platzierter Elemente löscht Frau Ergün diese nach Selektion. Im Infobereich (zweite Spalte unterhalb des Spielfeldes) werden Hinweise zur Spiel- und Level-Erstellung angezeigt. Dort speichert Frau Ergün den neu erstellten Level. Würde Frau Ergün ein ganz neues Spiel erstellen, würde sie über den Button „Start“ (im Bereich Spielelogik, siehe Abb. 1) die Startsequenz und Hinweise zum Spiel festlegen. In der dritten Spalte unterhalb des Spielfeldes findet Frau Ergün Funktionen, mit denen sie Medien hochladen und die verwendete Hardware konfigurieren kann.

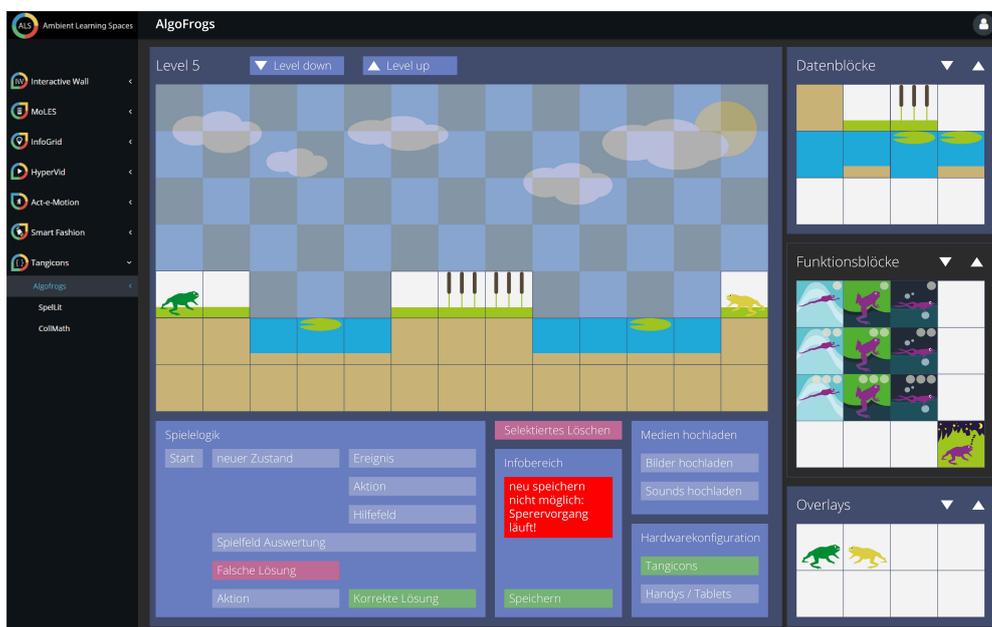


Abbildung 1: Ansicht eines Spielfeldes des ALS-GameCreator innerhalb des ALS-Portals. Im Spielfeld ist ein Level der AlgoFrogs dargestellt. Ziel ist es, durch gemeinsame Interaktion den grünen Frosch über das Spielfeld, das auf der InteractiveWall angezeigt wird, zum gelben Frosch zu bewegen. Auch dargestellt sind die Funktionsblöcke, die die Aktionen der Tangicons definieren.

2 Systeme anderer zur Erzeugung von Lernspielen

Ein zur Thematik verwandtes Projekt findet sich im z.B. im *Sifteo Creativity Kit* (Merrill et al., 2012), welches die webbasierte Erstellung von (Lern-)Spielen für die Sifteo Cubes ermöglichte. Das System ist mangels Unterstützung durch den Hersteller mittlerweile nicht mehr praktikabel einsetzbar. Inspiration für den ALS-GameCreator findet sich außerdem in den visuellen Programmiersprachen, wie sie Anwendung in den Projekten *Snap!*¹ oder *Vuo*² finden. Im Fokus unserer Entwicklung steht dabei, dem Anwender die Entwicklung eines Lernspiels unter der Nutzung von Medien und der im Projekt entwickelten Infrastruktur ohne Programmierkenntnisse zu ermöglichen.

3 Erzeugung be-greifbarer Miteinander-Lernspiele

Der ALS-GameCreator ermöglicht es dem Benutzer, webbasiert be-greifbare Miteinander-Lernspiele per Drag-and-Drop und ohne das Schreiben von Programmcode zu erstellen.

¹ Siehe <http://snap.berkeley.edu/>

² Siehe <http://vuo.org>

Dieser ist an das Framework *Network Environment for Multimedia Objects (NEMO)* angebunden, welches unter anderem die Infrastruktur für die Kommunikation zwischen Tangicons/mobilen Geräten sowie Multitouch-Bildschirmen/-Tischen und der Spielelogik realisiert und diese zu diesem Zweck über Web-Services bereitstellt. Durch die Verwendung der Tangicons, aber auch anderer Geräte, wie z.B. Handys, die im Sinne einer be-greifbaren Interaktion verwendet werden, werden die Lernspiele selbst be-greifbar. Hierzu wird die be-greifbare Interaktion mit den Tangicons in Gesten abstrahiert, die z.B. durch das Aneinanderstoßen, durch Auslösen der Lagesensoren durch Bewegung im Raum, Platzierung auf einem Tisch oder Nutzung als Fiducials auf einem Multitouch-Tisch ausgelöst werden. Der ALS-GameCreator ist modularer Bestandteil des *ALS-Portals*, der zentralen webbasierten Maske zum Einstellen und Verwalten aller ALS-Applikationen, Medien und Einstellungen im NEMO-gestützten ALS. Mit ihm wird innerhalb des ALS-Portals zuerst jedes Miteinander-Lernspiel mit Informationen zum Gegenstand des Spiels, des Fachbereichs und zum Schwierigkeitsgrad angelegt. Ein Lernspiels besteht aus mindestens einem oder mehreren Levels. Das Spielfeld eines neuen Levels, das später auf der InteractiveWall, dem Multitouch-Tisch oder einem anderen Gerät mit Browser angezeigt wird, ist zunächst leer und wird durch ein beliebig skalierbares Schachbrettmuster visualisiert, siehe Abb. 1. Per Drag-and-Drop werden auf dem Spielfeld Funktions- und Datenblöcke platziert, um den Level zu gestalten. Dabei beinhaltet ein Funktionsblock eine zuvor definierte, im Spielverlauf feste Funktion und Datenblöcke zuvor definierte Daten. Die Platzierung der Blöcke wird durch eine anpassbare Spielelogik ergänzt. In dieser definiert der Benutzer für einen jeden möglichen Ausgangszustand und alle Ereignisse, die er im Spielkonzept berücksichtigen möchte, den jeweiligen Folge- oder Zielzustand. Im Hintergrund wird dazu ein Regelgraph erzeugt. Für ungültige Interaktionen wird ebenfalls eine entsprechende Systemantwort definiert. Durch die Kombination von Funktionsblöcken, Datenblöcken und der anpassbaren Spielelogik lassen sich zahlreiche Interaktionen erstellen. Die Spielelogik ermöglicht durch das Einfügen von Übergängen (Transitionen) zwischen den Zuständen des Regelgraphen, den Ablauf des Spiels für jeden Level zu gestalten. Dazu wählt der Benutzer bei der Erstellung aus einem erweiterbaren Satz Bedingungen aus. Eine Transition wird durch eine Benutzerinteraktion (Geste) mit den Tangicons ausgelöst. Dies kann eine Interaktion mit einem Tangicon selbst, jedoch auch die Kombination zweier oder mehrerer Tangicons nebeneinander sein. Darüber hinaus sind u.a. zeitgesteuerte Transitionen implementiert, die z.B. nach Ablauf eines Timers automatisch auslösen. Außerdem können eigene Funktions- und Datenblöcke erstellt und mit diesen die Möglichkeiten für die Erstellung neuer Lernspiele erweitert werden. Hier können bereits vordefinierte Funktions- oder Datenblöcke durch neue Grafiken erweitert und genutzt werden. Für erfahrene Anwender mit Programmierkenntnissen ist es möglich, eigene Methoden zu implementieren und diese in Funktionsblöcken zur Verwendung zur Verfügung zu stellen. Neben den anpassbaren Inhalten eines Funktions- oder Datenblocks, der implizit Einfluss auf die Logik des Spiels nimmt, lassen sich beide Sorten von Blöcken frei mittels HTML, CSS, Bildern und Animationen gestalterisch definieren.

Der durch den ALS-GameCreator erzeugte Regelsatz, anhand dessen valide Spielzustände in der NEMO-Spielelogik geprüft werden, wird innerhalb des semantischen Modells von NEMO gespeichert und steht dort für die Erzeugung von Spieleinstanzen zur Verfügung.

Wird eine Instanz eines be-greifbaren Lernspiels gestartet, hostet die NEMO-Spielelogik das Spiel in einer eigenen Instanz und stellt die Infrastruktur für die Gerätekommunikation bereit. Über den *TangiconsWebService* melden sich die Benutzer mit ihren Tangicons an. Diese werden in der Spielinstanz registriert und erhalten aus dem Regelgraphen ihren Startzustand in Form eines Funktions- oder Datenblocks zugewiesen. Jede be-greifbare Interaktion eines Benutzers mit einem Tangicon führt zum Auslösen einer Geste, die an NEMO übertragen und von der NEMO-Spielelogik in der korrespondierenden Spieleinstanz innerhalb des Regelgraphen ausgewertet wird: Die NEMO-Spielelogik berechnet anhand des Spielzustandes und anhand des Zustandes, der dem Tangicon zugewiesen ist, ob es sich um eine valide Transition innerhalb des Regelgraphen handelt. Ist dies der Fall, wird ein Zielzustand erreicht, der in Form eines Funktions- oder Datenblocks an das oder die jeweiligen Tangicons ausgeliefert wird. So kann die Interaktion eines Benutzers mit einem Tangicon u.a. auch den Zustand aller übrigen Tangicons verändern. Wird ein nicht valider Zustand erreicht, so liefert die NEMO-Spielelogik den zuvor im ALS-GameCreator definierten Zustand in Form eines Funktions- oder Datenblocks aus. Wird innerhalb des Regelgraphens einer Spielinstanz durch die Spieler ein Endzustand eines Levels erreicht, so wechselt die NEMO-Spielelogik in den Startzustand des nächsten Levels. Zurzeit wird an der Umsetzung folgender Miteinander-Lernspiele gearbeitet: AlgoFrogs, SpellLit, CollMath und Energiewende.

4 Zusammenfassung und Ausblick

Während der Entwicklung des ALS-GameCreator in einem Co-Designprozess, wurde dieser mit Lehrkräften hinsichtlich der Erstellung der Spielelogik entwicklungsbegleitend qualitativ evaluiert. Die ersten qualitativen Ergebnisse zeigen, dass Lehrer mit dem ALS-GameCreator gut zurechtkommen. Die weitere Entwicklung widmet sich vor allem der Funktionserweiterung und Zuverlässigkeit sowie der Unterstützung des Benutzers bei der Spieleerstellung. Wir streben außerdem die Integration einer Unterstützung von Multitouch-Tischen an. In einer sich an die aktuelle Entwicklung anschließenden Evaluation ist geplant, die Hypothese der Unterstützung nachhaltigen Lernens durch das eigene Erstellen von Spielen durch Schüler*innen, die die zitierte Literatur nahelegt, näher zu evaluieren. Damit setzen wir im Bereich der Miteinander-Lernspiele einen neuen Fokus auf fortführende Schulen, in denen der ALS-GameCreator auch im Unterricht eingesetzt wird. Darüber hinaus versprechen wir uns durch die flexiblen Gestaltungsmöglichkeiten, die der ALS-GameCreator bietet, die Möglichkeit, den von uns vermuteten Mehrwert der Tangicons gegenüber anderen mobilen Geräten im Bereich der Vor- und Grundschule zu evaluieren.

Literaturverzeichnis

- Bouck-Standen, David; Schwandt, Marco; Winkler, Thomas; Herczeg, Michael (2016): ELBlocks - An Interactive Semantic Learning Platform for Tangibles. Mensch und Computer 2016 - Workshopband. Aachen: Gesellschaft für Informatik e.V.
- Dickey, Michele D. (2005): Engaging by design: How engagement strategies in popular computer and video games can inform instructional design: Kluwer Academic Publishers (Vol. 53, Issue 2), S. 67–83.
- Gee, James Paul (2003): What Video Games Have to Teach Us About Learning and Literacy: ACM Computers in Entertainment (1).
- Hahn, Claudia; Wolters, Christian; Winkler, Thomas; Herczeg, Michael (2012): Programmieren im Vorschulalter mit Hilfe von Tangicons. Mensch & Computer 2012 - Workshopband: 12. fachübergreifende Konferenz für interaktive und kooperative Medien. München: Oldenbourg Verlag, S. 135–140.
- Höfer, Christoph; Madelung, Petra (2006): Lehren und Lernen für die Zukunft. Unterrichtsentwicklung in selbstständigen Schulen: Bildungsverlag EINS.
- Huang, C.-C.; Yeh, T.-K.; Li, T.-Y.; Chang, C.-Y (2010): The Idea Storming Cube: Evaluating the Effects of Using Game and Computer Agent to Support Divergent Thinking. In: *Educational Technology & Society*, 13 (4), S. 180–191.
- Merrill, David; Sun, Emily; Kalanithi, Jeevan (2012): Sifteo Cubes. CHI. Austin (TX), USA.
- Prensky, Marc (2003): Digital Game-Based Learning: ACM Computers in Entertainment (1).
- Ritterfeld, U.; Cody, M.; Vorderer, P. (2009): Serious Games: Mechanisms and Effects. London: Routledge.
- Scharf, Florian; Winkler, Thomas; Hahn, Claudia; Wolters, Christian; Herczeg, Michael (2012): Tangicons 3.0: An Educational Non-Competitive Collaborative Game. IDC'12 Proceedings of the 11th International Conference on Interaction Design and Children. New York: ACM.
- Weinert, F. E. (2001): Leistungsmessung in Schulen. Weinheim und Basel.
- Winkler, Thomas; Scharf, Florian; Hahn, Claudia; Wolters, Christian; Herczeg, Michael (2014): Tangicons: Ein be-greifbares Lernspiel mit kognitiven, motorischen und sozialen Aktivitäten. i-com: Zeitschrift für interaktive und kooperative Medien: de Gruyter (13), S. 47–56.
- Winkler, Thomas; Scharf, Florian; Herczeg, Michael (2013): SpellLit – Tangible Cross-Device-Interaction beim Erlernen von Lesen Interaction beim Erlernen von Lesen und Schreiben. Workshopband Mensch & Computer 2013. München: Oldenbourg Verlag, S. 179–184.
- Winkler, Thomas; Scharf, Florian; Peters, Judith; Herczeg, Michael (2011): Tangicons Programmieren im Kindergarten. Workshop-Proceedings der Tagung Mensch & Computer 2011. Chemnitz: Universitätsverl. Chemnitz, S. 23–24.
- Winkler, Thomas; Stahl, Jacob; Jahr, Georg; Herczeg, Michael (2015): SpellLit 3.0 - Ein mobiles Miteinander-Lernspiel zum Erwerb der Schriftsprache. Mensch und Computer 2015 Workshopband. Stuttgart: Oldenbourg Wissenschaftsverlag, S. 543–550.

5 Autoren



Winkler, Thomas

Dr. Thomas Winkler ist wissenschaftlicher Mitarbeiter und Projektleiter/-koordinator im Bereich Computergestütztes Lernen für Kinder und Jugendliche am Institut für Multimediale und Interaktive Systeme (IMIS) der Universität zu Lübeck. Die Hauptaufgaben sind Interaktionsdesign, Augmented-, Mixed- und Virtual Reality, Mediendesign, Kunst- und Designtheorie, neue Kunstgeschichte und Kulturgeschichte sowie Ästhetik.



Bouck-Standen, David

David Bouck-Standen M.sc. studierte am Institut für Multimediale und Interaktive Systeme (IMIS) der Universität zu Lübeck. In seiner Masterarbeit entwickelte er das Framework „Network Environment for Multimedia Objects (NEMO)“ auf neuer technologischer Basis. Es dient der zentrale Haltung der Logik aller Lernprogramme im Forschungsvorhaben „Ambient Learning Spaces“ und von in diesen gemeinsam verwendeten Programmmodulen, der gemeinsamen semantischen Datenhaltung sowie diversen Modulen zur Konvertierung von Daten sowie der Benutzerauthentifizierung.



Herczeg, Michael

Prof. Dr. Michael Herczeg ist Universitätsprofessor für Praktische Informatik und Direktor des Instituts für Multimediale und Interaktive Systeme (IMIS) der Universität zu Lübeck. Die Hauptarbeitsgebiete sind Mensch-Maschine-Kommunikation, Software-Ergonomie, Interaktionsdesign, Multimediale und Interaktive Systeme, Computergestütztes Lehren und Lernen sowie Sicherheitskritische Mensch-Maschine-Systeme.