

Heuristiken für Second-Screen-Anwendungen

Valentin Lohmüller, Daniel Schmaderer, Christian Wolff

Lehrstuhl für Medieninformatik, Universität Regensburg

Zusammenfassung

In dieser Arbeit wurde auf Basis von Niensens (1994b) *Ten Usability Heuristics* in einem rechnerbasierten Ansatz anhand von spezifischer Literatur eine domänenspezifische Heuristik für Second-Screen-Anwendungen ausgearbeitet und zu einer Checkliste erweitert. Um die Qualität dieser Checkliste zu bewerten, wurde eine heuristische Evaluation einer Second-Screen-Anwendung mit fünf Anwendern durchgeführt und deren Ergebnisse mit einer Nutzerstudie mit 20 Teilnehmern verglichen. Dabei ergab sich eine mittlere Validität von 0.5 und eine hohe Vollständigkeit von 0.74. Das harmonische Mittel dieser Werte ergibt bei einer Gleichgewichtung ein F-Maß von 0.6. Dieser Wert spricht für eine ausreichende Validität der erstellten heuristischen Checkliste in der ersten Iteration.

1 Einleitung

Ziel von heuristischen Evaluationen ist die Erfassung des aktuellen Zustands einer Software anhand von *Regeln*, sogenannten Heuristiken, mit dem Ziel, die Gebrauchstauglichkeit des Untersuchungsobjekts zu verbessern. Dieser Prozess ist iterativ zu sehen, damit die Usability einer Anwendung ab einem frühem Entwicklungsstadiums zunehmend steigt (Guimaraes & Martins, 2015, S. 46). Heuristische Evaluationen gelten als besonders effiziente und kostengünstige Methode zur Bestimmung von Usability-Problemen und orientieren sich oft an den *Ten Usability Heuristics* von Nielsen (1994b). Um eine möglichst vollständige Liste an Usability-Problemen in einem bestimmten System zu generieren, ist es sinnvoll, ein angepasstes Set an Heuristiken für die jeweilige Domäne zu verwenden (Ling & Salvendy, 2005, S. 183). Beispiele hierfür sind Heuristiken für *Augmented-Reality-Anwendungen* (Guimaraes & Martins, 2015), *Information Appliances* (Böhm, Schneidermeier & Wolff, 2014) oder *Game Design* (Pinelle & Wong, 2008). Auch in dem Bereich *Second Screen und Smart-TV* existieren bereits Heuristiken (Mosqueira-Rey, Alonso-Ríos, Prado-Gesto & Moret-Bonillo, 2017; Solano et al., 2011) bzw. Guidelines (Pagno, Costa, Guedes, Freitas & Nedel, 2015; Weber, Mayer, Voit, Ventura Fierro & Henze, 2016), über die in den jeweiligen Arbeiten allerdings

keine Aussage über ihre Validität getroffen wird. Das übergeordnete Ziel von heuristischen Evaluationen ist eine hohe Validität, mithin die korrekte Vorhersage von schwerwiegenden Usability-Problemen des Untersuchungsgegenstands (Hvannberg, Law & Lárusdóttir, 2007, S. 226).

Heuristiken sind meist allgemein formuliert und werden deshalb von Anwendern oft unterschiedlich interpretiert (Böhm et al., 2014, S. 277). Andere, präzisere Heuristiken sind aufgrund ihres Umfangs weniger gut handhabbar. Deshalb wurde für die Heuristiken für Second-Screen-Anwendungen eine *Checkliste* ausgearbeitet, die konkrete und prägnante Anweisungen für den Anwender enthält, um eine effiziente und umfassende Identifikation von Usability-Problemen zu ermöglichen (Nielsen & Molich, 1992, 249f.).

Der nächste Abschnitt beschreibt den Gegenstandsbereich der neuen Heuristiken, *Second Screen*. Danach werden der Entstehungsprozess der rechnerbasierten Heuristiken erster Ebene und die Ableitung der Checklisten-Punkte näher beschrieben. Die dabei entwickelten Heuristiken werden in Abschnitt 4 erläutert. Die Validierung der heuristischen Checkliste wird in Abschnitt 5 beschrieben und abschließend diskutiert.

2 Second Screen als Heuristische Domäne

Unter *Second Screen* versteht man die Verwendung eines zweiten Bildschirms, wie z. B. ein Smartphone oder Tablet, während der Nutzung eines primären Bildschirms, wie z. B. eines Fernsehers (Cunningham & Weinel, 2015, S. 228). Eine präzise Einschätzung, wie oft und wie lange ein Second Screen verwendet wird, ist schwierig, aber Studien gehen davon aus, dass ein Großteil – zwischen 57% (Busemann & Tippelt, 2014, S. 410) und 83% (Johnen & Stark, 2015, S. 381) von Nutzern – wenigstens *selten* ein Smartphone, Tablet oder Laptop parallel zu einem primären Gerät verwendet. Durch das Aufkommen von Smart-TVs, Streaming-Sticks, Set-Top-Boxen und ähnlichen Geräten in den letzten Jahren hat Second Screen eine weitere Perspektive in Form von direkt verbundenen und bidirektional kommunizierenden Second-Screen-Anwendungen bekommen. Prominente Vertreter hierfür sind Anwendungen wie *Netflix*, *Amazon Prime Video* oder *YouTube*, die es Nutzern ermöglichen, über ihr mobiles Endgerät zusätzliche Informationen angezeigt zu bekommen und den Inhalt auf ihrem Fernseher zu steuern. Dadurch können die Vorteile beider Bildschirme in einer Second-Screen-Anwendung kombiniert werden.

Bei der Entwicklung von Second-Screen-Anwendungen müssen zahlreiche Besonderheiten gegenüber herkömmlichen Anwendungen berücksichtigt werden, um eine zufriedenstellende Nutzererfahrung zu schaffen, wie die sinnvolle Verteilung von Informationen und Zuständigkeiten beider Komponenten oder die Lenkung der Aufmerksamkeit durch Benachrichtigungen. Arbeiten wie Mosqueira-Rey et al. (2017) oder der verwandte Themenbereich bei Solano et al. (2011) haben bereits nach einem evaluationsbasierten Ansatz domänenspezifische Heuristiken entworfen. Darüber hinaus gibt es Empfehlungen zur Gestaltung von Second-Screen-Anwendungen wie die Pagno et al. (2015) oder Weber et al. (2016), die ähnlich wie Niensens *Ten Usability Heuristics* eine Grundlage für Heuristiken darstellen können (Hvannberg et al., 2007, S. 226). Die Validität der Richtlinien wurde in keiner der benannten Arbeiten geprüft,

weshalb diese hier zu einer heuristischen Checkliste synthetisiert werden, für die sich Validität, Vollständigkeit und Effektivität bestimmen lassen.

3 Entwicklung der Heuristiken

Nach Ling und Salvendy (2005, S. 186) können domänenspezifische Heuristiken nach zwei unterschiedlichen Ansätzen entwickelt werden: dem *evaluations-* und dem *recherchebasierten* Ansatz. Bei einem evaluationsbasierten Ansatz werden allgemeine Usability-Probleme mit bestimmten Systemen in Heuristiken kategorisiert. Diese Art der Ableitung basiert zwar auf empirischen Beobachtungen, ist jedoch oft abhängig von den untersuchten Objekten und daher weniger geeignet für die Ableitung generischer Heuristiken (Böhm et al., 2014, S. 278). Der recherchebasierte Ansatz identifiziert aufgrund geeigneter Literatur Anforderungen und Schlüsselfaktoren einer bestimmten Domäne. Dieser Ansatz wurde gewählt, um anhand von bestehender Literatur im Bereich Heuristiken und Guidelines im Themenbereich Second Screen eine domänenspezifische Heuristik zu entwickeln.

Dabei wird häufig ein Set an Heuristiken als Grundlage gewählt, das angepasst und mit Hilfe entsprechender Literatur ergänzt wird (Ling & Salvendy, 2005, 183ff.). Dabei wurde eine Fokusgruppe aus Experten der Zieldomäne gebildet, um die subjektiven Einflüsse möglichst gering zu halten. Dieses Ergebnis stellt die erste Stufe der Heuristiken dar, welche in einer zweiten Stufe weiter konkretisiert wurde. Abbildung 1 zeigt den Ablauf der Arbeitsschritte bei der Entwicklung der Heuristiken.

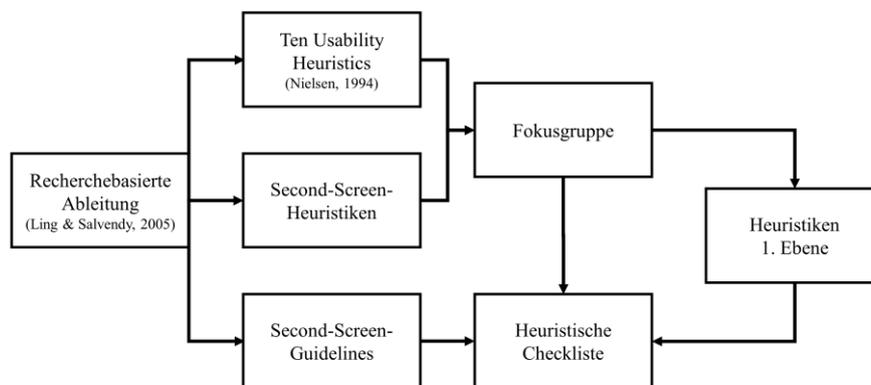


Abbildung 1: Vorgehen zu den Entwicklung einer heuristischen Checkliste für Second-Screen-Anwendungen

3.1 Recherchebasierte Ableitung der Heuristiken

Die Grundlage der Heuristiken im Kontext Second Screen bildeten aufgrund ihrer hohen Verbreitung die *Ten Usability Heuristics* von Nielsen (1994b). Diese wurden um die bereits existierende domänenspezifischen Heuristiken von Mosqueira-Rey et al. (2017) und Solano et al.

(2011) erweitert. Dieser Prozess wurde von einer Fokusgruppe aus zwei zusätzlichen Experten im Bereich der Second-Screen-Entwicklung unterstützt, um namentliche und semantische Duplikate zu identifizieren, Themenbereiche zu gruppieren, weniger relevante Heuristiken im Bereich Second Screen zu finden und die neuen Heuristiken um domänenspezifische Punkte zu erweitern. Dabei wurden acht der ursprünglichen Heuristiken von Nielsen als auf die neue Domäne übertragbar befunden und angepasst, sowie zwei neue Heuristiken speziell für Second-Screen-Anwendungen hinzugefügt. Dabei wurde bewusst darauf geachtet, die Anzahl von zehn Heuristiken nach dem Vorbild von Nielsen (1994b) nicht zu überschreiten. Auch wenn eine höhere Anzahl an Heuristiken potentiell mehr Usability-Probleme identifiziert, stellen diese eine höhere kognitive Belastung für den Anwender dar, weshalb die Gesamtanzahl von Heuristiken nicht zu groß werden sollte (Ling & Salvendy, 2005, S. 192). Diese zehn Heuristiken bilden die erste Stufe der Heuristik für Second-Screen-Anwendungen und sind auf Grund ihrer Spezialisierung umfassender formuliert als die ursprünglichen Heuristiken. Deshalb wurde in der zweiten Stufe eine heuristische Checkliste mit konkreten Anweisungen entwickelt, um die Nutzung der Heuristiken zu erleichtern.

3.2 Ableitung der Heuristischen Checkliste

Für die Erstellung der heuristischen Checkliste wurden aus den bereits existierenden Heuristiken (Mosqueira-Rey et al., 2017; Solano et al., 2011) und Guidelines (Pagno et al., 2015; Weber et al., 2016) in der Zieldomäne einzelne Checklistenpunkte erstellt und in die bereits existierenden Heuristiken erster Ebene eingeordnet. Guidelines eignen sich aufgrund ihrer konkreteren Natur, um Heuristiken präziser zu erweitern (Böhm et al., 2014, S. 277). Dabei wurden die zuvor erlangten Erkenntnisse hinsichtlich der Priorisierung aus der Fokusgruppe berücksichtigt. Insgesamt wurden 66 Punkte erstellt, von denen 51 in die heuristische Checkliste integriert wurden. Das Ergebnis der heuristischen Checkliste für Second-Screen-Anwendungen ist in Abschnitt 4.2 beschrieben.

4 Heuristiken für Second-Screen-Anwendungen

Auf Grundlage von Nielsen (1994b) *Ten Usability Heuristics* und den Arbeiten von Mosqueira-Rey et al. (2017), Pagno et al. (2015), Solano et al. (2011) und Weber et al. (2016) wurde durch eine Fokusgruppe ein Set von domänenspezifischen Heuristiken für Second Screen erstellt. Auf Basis der Heuristiken erster Ebene wurden mit Hilfe der Literatur konkrete Stichpunkte erstellt, um unterschiedlichen Interpretationen von Anwendern der Heuristik entgegenzuwirken. Diese wurden in englischer Sprache verfasst, um eine breitere Anwendergruppe zu erreichen.

4.1 Heuristiken für Second-Screen-Anwendungen Erster Ebene

Sichtbarkeit des Systemstatus: Das System sollte den Benutzer immer auf dem Laufenden halten, indem es angemessenes Feedback in einer angemessenen Zeit liefert. Der Nutzer sollte zu jeder Zeit Überblick über den aktuellen Verbindungsstatus zwischen First und Second Screen

haben. Der aktuelle Inhalt auf dem First Screen sollte jederzeit auf dem Second Screen erkenntlich sein, sofern eine Verbindung besteht, um den Nutzer einen guten Überblick über beide Systeme zu geben. Beide Teile des Systems sollten denselben Status anzeigen.

Übereinstimmung zwischen dem System und der realen Welt: Eine Second-Screen-Anwendung sollte die Sprache des Nutzers sprechen. Dabei sollen Wörter, Phrasen und das Konzept der Second-Screen-Anwendung in einer natürlichen Ordnung dargestellt werden. Falls dies nicht der Fall ist, werden sich Schwierigkeiten bei der Nutzung der Anwendung häufen. Vor allem bei der Verbindung der Second-Screen-Anwendung mit dem TV könnten komplizierte Fachbegriffe die Bedienung erschweren oder sogar unmöglich machen.

Benutzerkontrolle und -freiheit: Der Nutzer sollte zu jeder Zeit Kontrolle über den Inhalt auf dem First Screen haben, sofern eine Verbindung besteht. Eine Second-Screen-Anwendung sollte dem Nutzer die Möglichkeiten geben, Aktionen rückgängig zu machen, die fälschlicherweise oder aus Versehen ausgeführt wurden.

Konsistenz und Standards: Das Design und Layout des Interfaces, sowie die Nutzerinteraktion sollte auf beiden Bildschirmen konsistent sein. Außerdem sollten standardisierte Icons, Konventionen und Begrifflichkeiten verwendet werden.

Fehlervermeidung: Das Design und die Erklärungen einer Second-Screen-Anwendung sollte das Auftreten von Fehlern so gut es geht verhindern. Falls es jedoch doch zu Fehlern kommen sollte, ist es wichtig, dass diese so klar und präzise wie möglich beschrieben sind. Dadurch wird es dem Nutzer erleichtert, mit den Fehlermeldungen umzugehen.

Wiedererkennen statt Erinnern: Objekte, Optionen und Aktionen sollen bei Second-Screen-Anwendungen sichtbar sein. Der Nutzer soll in allen Bereichen seine Möglichkeiten vor Augen haben und sich nicht an diese erinnern müssen.

Ästhetik und minimalistisches Design: Eine Second-Screen-Anwendung sollte den Nutzer nicht mit irrelevanten Informationen beschäftigen, denn dadurch wird er von den relevanten Informationen abgelenkt. Am wichtigsten ist dies beim Dialog des Systems mit dem Nutzer.

Hilfe beim Erkennen, Diagnostizieren und Beheben von Fehlern: Beim Auftreten von Fehlern in einer Second-Screen-Anwendung sollte die Hilfe in der Sprache des Nutzers formuliert werden. Bei Fehlern, die der Nutzer selbst lösen kann, sollte der Fehlermeldung eine Anleitung zur Beseitigung des Fehlers angeboten werden. Bei Fehlern, die der Nutzer nicht lösen kann, sollte dies klar formuliert werden. Die Fehlerursache sollte immer klar erkenntlich sein.

Verbindungsprozess: Der Verbindungsprozess sollte möglichst einfach und von überall verfügbar gestaltet werden. Dem First und Second Screen sollten die richtigen Rollen zugewiesen werden, dabei sollte die Steuerung auf dem Second und die Mediendarstellung auf dem First Screen stattfinden.

Erzeuge einen Mehrwert durch den Second Screen: Eine Second-Screen-Anwendung sollte nur dann verwendet werden, wenn sie auch einen Mehrwert für Nutzer bietet.

4.2 Heuristische Checkliste für Second-Screen-Anwendungen

1. *Visibility of the System Status*
 - a. Does the application give the user feedback?
 - i. At performing key actions
 - ii. At reasonable time
 - b. Is the status of the connection kept updated?
 - c. Are the screens keep synchronized instantaneously?
2. *Match between the system and the real world*
 - a. Does the application speak the user's language?
 - i. Understandable terms/descriptions
 - b. Does the application show the information in a natural order?
 - c. Does the sequence of activities follow the user's mental processes?
3. *User control and freedom*
 - a. Is the navigation simple and intuitive for the operating system?
 - i. Menu
 - ii. Search bar
 - b. Does the application provide different options?
 - i. Return to top level
 - c. Is the user able to explore the application freely?
 - d. Is the user able to control the content of the TV at any time?
4. *Consistency and standards*
 - a. Does the application follow the design guidelines of the using platform?
 - b. Is the consistency between the two applications given?
 - i. Terminology
 - ii. Controls
 - iii. Graphics/Icons
 - iv. Focus on one guideline (if multiple apply)
5. *Error prevention*
 - a. Is there a help for novice users?
 - b. Does the application provide appropriate error messages?
6. *Recognition rather than recall*
 - a. Is the relationship between the controls and their actions obvious?
 - b. Does the user know what options he has and where to go?
 - i. Main elements of application always available
 - ii. Help available if needed
7. *Aesthetic and minimalist design*
 - a. Does the application only show relevant and necessary information to the user?
 - i. Titles and headlines short but descriptive
 - b. Is the application design appropriate?
 - i. Distance between elements
 - ii. Size
 - iii. Placement
 - c. Are the elements of the application visible?
 - i. At the visual range of watching TV

- ii. At various types of lighting
- 8. *Help users recognize, diagnose and recover from errors*
 - a. Does the application provide clear messages with indicating errors and solutions for errors?
 - i. Connection fails, application crashes, etc.
 - b. Are the error messages written in an accurate way?
 - i. Not blaming the user
 - ii. Non-derisory tone
 - c. Does the application provide users a clear and simple help, in their own language?
- 9. *Discovery and Connection*
 - a. Is the pairing of main and secondary display simple and intuitive?
 - i. 1-3 clicks needed
 - ii. Direct Response after pairing
 - b. Is the separation between the two applications and devices clear?
 - c. Is the main logic on the mobile device?
 - d. Is the content shown on the TV?
- 10. *Use a second-screen when it adds value*
 - a. Does the second-screen add value to the TV application?
 - b. Does the second-screen improve the content navigation?
 - c. Does the second-screen give the user a better user experience?

5 Validierung der Heuristiken

Bei der Bewertung von Heuristiken werden in der Regel drei Maße herangezogen, Validität, Vollständigkeit und Reliabilität (Böhm et al., 2014, S. 281; Hartson, Andre & Williges, 2003, S. 160; Sears, 1997, S. 214). Um diese Werte zu berechnen, wurde eine Second-Screen-Anwendung von fünf Experten mit Hilfe der erstellten heuristischen Checkliste evaluiert und diese Ergebnisse mit denen aus einer Usability-Studie von 20 Nutzern gegenübergestellt. Untersuchungsobjekt war eine Second-Screen-Mediatheken-Anwendung, die sich noch in Entwicklung befand. Ziel war die frühzeitige Identifikation von Problemen bezüglich der Gebrauchstauglichkeit, bevor diese bei Endnutzern eintreten. Für eine aussagekräftige Bewertung der Reliabilität müssten weitere heuristische Evaluationen an Second-Screen-Anwendungen durchgeführt werden, was in der ersten Iteration nicht möglich war. Nielsen und Molich (1992, S. 255) empfehlen zwischen drei und fünf Anwender einer heuristischen Evaluation für die effizienteste Bestimmung von Usability-Problemen. Die Experten untersuchten mit Hilfe der Checkliste die Anwendung, ordneten gefundene Probleme hinsichtlich der Heuristiken ein und vergaben einen Schweregrad nach Nielsen (1994a) von 0, *kein Problem*, bis 4, *muss behoben werden*. Abschließend wurden die Experten zu den verwendeten Heuristiken befragt.

Die in der heuristischen Evaluation vorhergesehenen Probleme in der Usability in der Anwendung wurden verwendet, um die Tasks der Nutzerstudie zu generieren (Hvannberg et al., 2007, 299f.). Hierfür wurden semantische und inhaltliche Duplikate in den gefundenen Problemen entfernt und thematisch gruppiert. Aus den resultierenden Gruppen *Feedback*, *Hilfe*, *Fehler*, *Verbindung*, *Suche*, *Menü*, *Video*, *Navigation* und *Playlist* wurden Tasks für die Nutzer so

konstruiert, dass sie nicht direkt mit den entsprechenden Problemen konfrontiert werden. Ziel war, zu überprüfen, inwiefern die prognostizierten Probleme der heuristischen Evaluation echten Problemen der Nutzer entsprechen. Abschließend wurde der *System Usability Scale* (SUS) (Brooke, 1996) von der untersuchten Anwendung erhoben und eine teilstrukturierte Befragung durchgeführt. Die Ergebnisse der Evaluation der erstellten heuristischen Checkliste werden im Folgenden dargestellt:

Validität

Heuristiken besitzen eine hohe Validität, wenn möglichst viele der prognostizierten Probleme aus der Evaluation mit den *echten / tatsächlichen* Problemen der Nutzer übereinstimmen (Hartson et al., 2003, 163f.). Dabei werden die *echten / tatsächlichen gefundenen Probleme* als die Schnittmenge der von den Nutzern und Experten gefunden Problemen gesehen.

$$\text{Validität} = \frac{\text{Anzahl echter gefundener Probleme (Experten} \cap \text{Nutzer)}}{\text{Anzahl Probleme von identifizierten Problemen (Summe Experten)}} = \frac{14}{28} = 0,5$$

14 der 28 der prognostizierten Fehler wurden von Nutzern bestätigt, was eine Validität von 0.5 der angewandten Heuristiken bei dem Untersuchungsobjekt ergibt. Dieser Wert zeugt von einer mittelmäßigen Validität und spricht dafür, dass eher unterschiedliche Fehler zwischen Nutzern und Experten gefunden wurde. Die Validität einer Heuristik sinkt in der Regel mit der Anzahl der Anwender, die in dieser Studie am oberen Ende anzusiedeln ist. Des Weiteren befand sich das Untersuchungsobjekt noch in der Entwicklung, was ein Grund für die eher hohe Anzahl von Usability-Problemen sein könnte. Probleme, die von Experten prognostiziert wurden und nicht von Nutzern bestätigt werden, bringen während der Entwicklung einer Anwendung trotzdem einen Mehrwert, da diese Fehler frühzeitig behoben werden können. Somit kann die hohe Anzahl von gefundenen Fehlern der Experten außerhalb der angewandten Maße als positiv für den Entwicklungsprozess gesehen werden, da hier der eigentliche Sinn einer heuristischen Evaluation liegt: die effiziente und kostengünstige Identifikation von Usability-Problemen, bevor ein Endnutzer mit diesen konfrontiert wird.

Vollständigkeit

Die Vollständigkeit beschreibt die Anzahl der vorhandenen Probleme, die durch die heuristische Evaluation identifiziert werden konnten. Dabei wird wieder die Schnittmenge der gefundenen Probleme von Experten und Nutzer als *echte / tatsächliche gefundene Probleme* gesehen und die Summe *aller* Nutzerprobleme als *Anzahl echter existierender Probleme*. Die nicht von Nutzern benannten Probleme der Experten stellten sich somit als *false positives* heraus und werden in der Validität berücksichtigt (Hartson et al., 2003, 163f.).

$$\text{Vollständigkeit} = \frac{\text{Anzahl echter gefundener Probleme (Experten} \cap \text{Nutzer)}}{\text{Anzahl echter existierender Probleme (Alle Nutzer)}} = \frac{14}{19} = 0,74$$

Mit einem Wert von 0,74 weist die heuristische Checkliste eine hohe Vollständigkeit auf. Diese beruht unter anderem auf der hohen Anzahl von identifizierten Problemen der Experten.

Effektivität und F-Maß

Anhand der Maße Vollständigkeit und Validität kann die Effektivität von Heuristiken berechnet werden. Diese ergibt sich aus einer einfacher Multiplikation der beiden Werte.

$$\text{Effektivität} = \text{Vollständigkeit} \times \text{Validität} = 0,74 \times 0,5 = 0,37$$

Der eher niedrige Wert 0.37 ist bedingt durch die mittelmäßige Validität der Heuristiken. Hartson et al. (2003, 165f.) bemerkt die starke Beeinflussung dieses Effektivitätsmaßes durch einen niedrigen Wert und beschreibt eine Berechnung eines gewichteten F-Maßes:

$$F = \frac{1}{\alpha(1 / \text{Validität}) + (1 - \alpha)(1 / \text{Vollständigkeit})}$$

Bei einer Gleichgewichtung beider Werte ($\alpha = 0.5$) ergibt sich ein F-Maß von 0.60, das ein gewichtetes Maß zwischen Validität und Vollständigkeit beschreibt. Dieser Wert liegt im mittleren bis positiven Bereich und beschreibt ein akzeptables Abschneiden der Heuristiken.

6 Diskussion und Ausblick

Die Ergebnisse der Evaluation deuten auf eine ausreichende Validität der recherchebasierten abgeleiteten heuristischen Checkliste für Second-Screen-Anwendungen hin. Diese Arbeit ist als erster Schritt einer gründlichen Validierung der hier erstellten Heuristiken zu sehen. Dieser Prozess kann auf weitere Second-Screen-Anwendungen und heuristische Evaluationen ausgeweitet werden, um noch besseren Einblick bezüglich der hier erstellten Heuristiken zu gewinnen. Auch inhaltlich kann die hier entstandene Grundlage weiter angepasst und erweitert werden. Heuristiken in Form von konkreten und prägnanten Stichpunkten wurden von den Evaluatoren als besonders positiv empfunden, was für weitere Untersuchungen spricht.

Danksagung

Wir danken dem Bayerischen Staatsministerium für Wirtschaft, Energie und Technologie (Förderkennzeichen IUK-1506-0005// IUK475/002) für die Finanzierung dieser Forschungsarbeit, sowie unserem Projektpartner MEKmedia für die Unterstützung bei dessen Durchführung.

7 Literaturverzeichnis

- Böhm, P., Schneidermeier, T. & Wolff, C. (2014). Heuristiken für Information Appliances. In A. Butz, M. Koch & J. Schlichter (Hrsg.), *Mensch & Computer 2014 - Tagungsband* (S. 275–284). Berlin: De Gruyter Oldenbourg.
- Brooke, J. (1996). SUS-A quick and dirty usability scale. *Usability evaluation in industry*, 189 (194), 4–7.

- Busemann, K. & Tippelt, F. (2014). Second Screen: Parallelnutzung von Fernsehen und Internet. *Media Perspektiven*, 7, 408–416.
- Cunningham, S. & Weinel, J. (2015). Second screen comes to the silver screen. A technology feasibility study regarding mobile technologies in the cinema. In R. Picking (Hrsg.), *2015 internet technologies and applications (ITA). Proceedings of the sixth international conference : Tuesday 8th-Friday 11th September 2015, Glyndwr University, Wrexham, Wales, UK* (S. 228–232). Piscataway, NJ: IEEE.
- Guimaraes, M. d. P. & Martins, V. F. (2015). A Checklist to Evaluate Augmented Reality Applications. In *2015 XVII Symposium on Virtual and Augmented Reality (SVR). 25 - 28 May 2015, São Paulo, Brazil* (S. 45–52). Piscataway, NJ: IEEE.
- Hartson, H. R., Andre, T. S. & Williges, R. C. (2003). Criteria For Evaluating Usability Evaluation Methods. *International Journal of Human-Computer Interaction*, 15 (1), 145–181. https://doi.org/10.1207/S15327590IJHC1501_13
- Hvannberg, E. T., Law, E. L.-C. & Lárusdóttir, M. K. (2007). Heuristic evaluation. Comparing ways of finding and reporting usability problems. *Interacting with Computers*, 19 (2), 225–240. <https://doi.org/10.1016/j.intcom.2006.10.001>
- Johnen, M. & Stark, B. (2015). Wenn der Fernseher nicht mehr ausreicht: Eine empirische Analyse der Second Screen-Nutzung When watching television becomes insufficient: An empirical analysis of second screen usage. *SC/ M Studies in Communication/ Media*, 4 (4), 364–405.
- Ling, C. & Salvendy, G. (2005). Extension of heuristic evaluation method: a review and reappraisal. *Ergonomics An International Journal of Ergonomics and Human Factors*, 2005 (27 (3)), 179–197.
- Mosqueira-Rey, E., Alonso-Ríos, D., Prado-Gesto, D. & Moret-Bonillo, V. (2017). Usability evaluation and development of heuristics for second-screen applications. In S. Y. Shin, D. Shin & M. Lencastre (Hrsg.), *Proceedings of the Symposium on Applied Computing - SAC '17* (S. 569–571). New York, New York, USA: ACM Press.
- Nielsen, J. (1994a). Usability inspection methods. In C. Plaisant (Hrsg.), *Conference Companion on Human Factors in Computing Systems* (S. 413–414). New York, NY: ACM.
- Nielsen, J. (Hrsg.). (1994b). *Usability inspection methods*. New York NY u.a.: Wiley.
- Nielsen, J. & Molich, R. (1992). Heuristic evaluation of user interfaces. In J. C. Chew, J. C. Carrasco & J. Carrasco Chew (Hrsg.), *Empowering people. CHI '90 conference proceedings* (Human factors in computing systems, Bd. 1990, S. 249–256). Reading, MA: Addison-Wesley.
- Pagno, B., Costa, D., Guedes, L., Freitas, C. D. S. & Nedel, L. (2015). Guidelines for Designing Dynamic Applications with Second Screen. In *2015 XVII Symposium on Virtual and Augmented Reality (SVR). 25 - 28 May 2015, São Paulo, Brazil* (S. 42–51). Piscataway, NJ: IEEE.
- Pinelle, D. & Wong, N. (2008). Heuristic evaluation for games. In M. Burnett (Hrsg.), *The 26th Annual CHI Conference on Human Factors in Computing Systems, CHI 2008. Conference proceedings ; April 5-10, 2008 in Florence, Italy* (S. 1453). New York, NY: ACM.
- Sears, A. (1997). Heuristic Walkthroughs. Finding the Problems Without the Noise. *International Journal of Human-Computer Interaction*, 9 (3), 213–234. https://doi.org/10.1207/s15327590ijhc0903_2
- Solano, A., Rusu, C., Collazos, C., Roncagliolo, S., Arciniegas, J. L. & Rusu, V. (2011). Usability Heuristics for Interactive Digital Television. In E. Borcoci & J. Bi (Eds.), *AFIN 2011. The Third International Conference on Advances in Future Internet : August 21-27, 2011, Nice/Saint Laurent du Var, France*. Wilmington, DE, USA: IARIA.
- Weber, D., Mayer, S., Voit, A., Ventura Fierro, R. & Henze, N. (2016). Design Guidelines for Notifications on Smart TVs. In P. Whitney, J. Murray, S. Basapur, N. Ali Hasan & J. Huber (Hrsg.), *TVX 2016. Proceedings of the 2016 ACM International Conference on Interactive Experiences for TV and Online Video : June 22-24, 2016, Chicago, IL, USA* (S. 13–24). New York, NY: ACM Association for Computing Machinery.