

User Experience Design - Wie Sie die Zufriedenheit Ihrer Anwender planen, messen und steigern können

DI Dr. Robert Pucher

University of Applied Sciences Technikum Wien, Wien, Österreich
User Experience Professionals Association – Austria, Wien, Österreich

robert.pucher@uxpa-austria.org
robert.pucher@technikum-wien.at

Abstract-Usability und User Experience (UX) sind zu den wichtigsten Qualitätsmerkmalen von Software geworden. Immer mehr Unternehmen versuchen ihre Software Produkte im Bereich User Experience zu verbessern. Obwohl in vielen Fällen der subjektive Eindruck besteht, dies sei ganz einfach, gehört gutes UX Design innerhalb etablierter Softwareentwicklungsprozesse zu den komplexen Aufgaben. Der Grund dafür ist die Abhängigkeit von User Experience von drei Faktoren, vom Menschen, vom Kontext, in dem die Software genutzt wird, und der Software an sich. Besonders der Mensch erweist sich in diesem Zusammenhang als die am schwierigsten zu behandelnde Komponente. Der gegenständliche Artikel versucht einen kurzen Überblick über diese Problematik zu geben und einen Lösungsansatz zu skizzieren.

I. EINLEITUNG

Usability und User Experience gehören zu den wichtigsten Qualitätsmerkmalen von Software. Umso verwunderlicher ist es, dass auch heute noch diese nicht-funktionale Eigenschaft von Software bzw. User Interfaces in so vielen Bereichen als mangelhaft erlebt wird. Ein Beispiel ist in Abbildung 1 dargestellt. Betrachten Sie das Bild einige Zeit und stellen Sie sich die Frage, „Wie wird die Uhrzeit korrekt eingestellt?“.



Abb. 1: Wie wird die Uhrzeit korrekt eingestellt?

Negative Beispiele wie dieses lassen sich ohne Mühe endlos aufzählen. Der Online Check-in-Prozess von Fluglinien, oder die Reservierung eines Sitzplatzes in der Bahn, aber auch Anwendungen, die entwickelt wurden um Geschäftsabläufe zu

vereinfachen, stellen sich oft als ausgesprochen anwenderunfreundlich heraus.

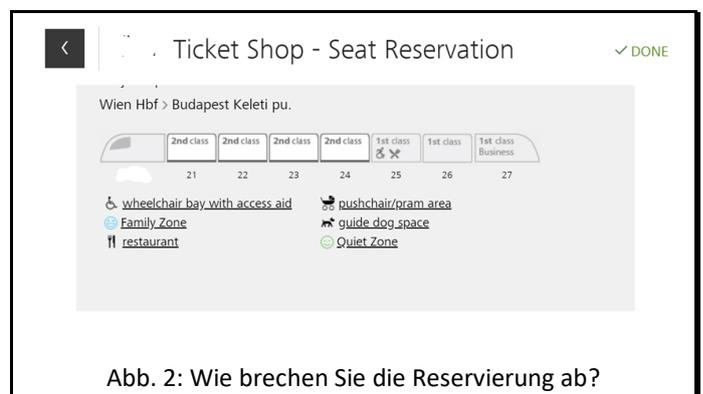


Abb. 2: Wie brechen Sie die Reservierung ab?

In Abbildung 2 ist die Reservierung eines Sitzplatzes in der Bahn dargestellt. Die Anwendung ist für Computer, für Smartphones und Tablets optimiert (responsive design). Allerdings sind viele Menschen im dargestellten Dialog nicht in der Lage herauszufinden, wie die Reservierung *abgebrochen* werden kann. Die Ursache für dieses Problem kann schnell gefunden werden. Auf Grund von sinnesphysiologischen und wahrnehmungsphysiologischen Vorgängen ist die Gestaltung der beiden klickbaren Elemente für „zurück“ und für „bestätigen“ problematisch gelöst. Die angewandte Darstellung wird fehlinterpretiert, da die Fähigkeiten von Menschen, die wahrgenommene Sinneseindrücke zielgerichtet zu verstehen, nicht berücksichtigt ist.

II. NEUROPHYSIOLOGIE UND USABILITY

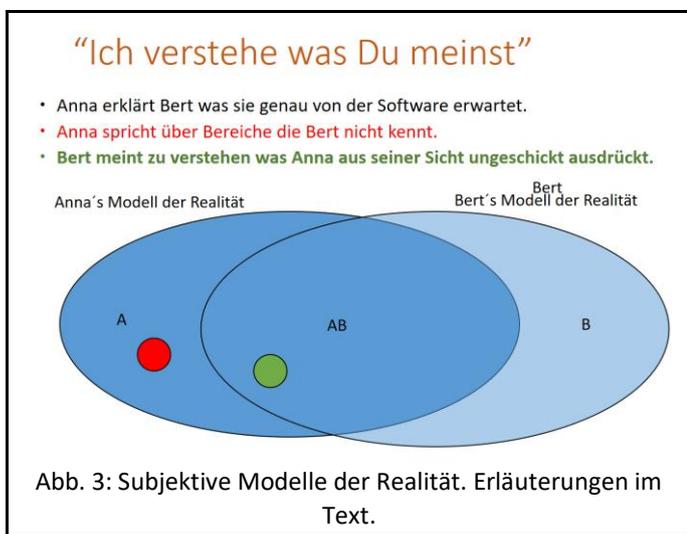
Die meisten Menschen, die sich mit dem einen oder anderen Softwareprodukt auseinandersetzen, beklagen fehlende Usability, „die aber so einfach zu erreichen wäre“. Interessanterweise ist die Vorstellung weit verbreitet, es wäre ganz einfach die Probleme die gerade jetzt mit der Bedienung auftreten, zu beseitigen. Wenn das so wäre, warum ist ganz offensichtlich die Usability von so vielen Anwendungen so schlecht? Das ist eine ernste Frage mit der wir uns an der Fachhochschule Technikum Wien¹ und in der UXPA-Austria² über viele Jahre hinweg beschäftigt haben.

¹ <https://www.technikum-wien.at>

² <https://www.uxpa-austria.at>

A. Die Realität ist ein subjektives Konstrukt des Gehirns

Realität ist aus Sicht des Menschen nicht objektiv, sondern radikal subjektiv. Aus Sicht eines beliebigen Individuums existiert nur eine Realität und dies ist die Realität, die von den Sinnesorganen wahrgenommen wird, vom Gehirn in überwiegend unbewussten Arealen, wie z.B. in der Sehrinde, aufbereitet und verarbeitet wird. Diese sodann als „Realität“ wahrgenommene Welt stellt keinesfalls eine Realität dar. Tatsächlich sind die Elemente dieser als „Realität“ wahrgenommenen Welt, lediglich Antworten auf die gerade vom Individuum auf die Welt gerichteten Fragen. Etwa 10% der Information wird dabei aus der Wahrnehmung bezogen, die restlichen 90 Prozent stammen aus dem Gedächtnis und sind damit von der Vorerfahrung des Individuums abhängig [1]. Besonders problematisch wird diese Tatsache, wenn zwei Menschen mit völlig unterschiedlicher Vorerfahrung über Anforderungen an eine Software sprechen.



Alle Erfahrungen, die Anna bisher in ihrem Leben mit der Fragestellung gemacht hat, sind im dunklen Oval angedeutet (A). Alle Erfahrungen die Bert mit derselben Fragestellung gemacht hat, sind im hellen Oval (B) angedeutet. Der Bereich, in dem beide ähnliche Erfahrungen gemacht haben, ist in der Überschneidung der Ovale (AB) angedeutet. Problematisch ist die Situation, weil sowohl Anna als auch Bert ausschließlich die Möglichkeit haben über die Teile der Realität nachzudenken, die sie kennen. Nehmen wir einmal an, Anna beschreibt einen Wunsch, der ausschließlich im Bereich liegt, den nur sie selbst kennt (A). Bert muss aber mit seinem Weltbild versuchen das Problem zu verstehen und wird aller Wahrscheinlichkeit nach auch eine Interpretation des Gehörten finden Diese kann jedoch kann nur in den von ihm überblickten Bereichen (B) und (AB) liegen, da er den Bereich A nicht kennt. Ein möglicherweise dramatisches Missverständnis kann hier seinen Ausgangspunkt nehmen.

B. Selektive Wahrnehmung

Menschen sind nicht in der Lage Informationen, die ihnen über die Sinnesorgane dargeboten werden, vollständig zu verarbeiten. In vielen Jahren der Evolution haben sich Verfahren entwickelt, die für das Individuum mit hoher Wahr-

scheinlichkeit relevante Informationen aus dem Datenstrom, der von den Sinnesorganen produziert wird, herausfiltern. Die dabei angewandten Kompression- und Dekodieralgorithmen erzeugen aus ca. 10^9 Bit/s Rohdaten ein subjektives bewusst wahrgenommenes Modell der Welt. Dieses bewusst wahrgenommene Modell der Realität wird nur mehr von etwa 10^2 Bit/s sehr komplex kodierter Information getragen. Abstrakte Zahlen sind beeindruckend, um jedoch ein tatsächliches Gefühl für die lückenhaften Prozesse der Wahrnehmung zu bekommen, lohnt es sich folgende auf YouTube angebotenen Videos in der dargestellten Reihenfolge anzusehen.

1. Die Wahrnehmung von Details

<https://www.youtube.com/watch?v=ubNF9ONEQLA>

2. Das Übersehen von wichtigen Informationen

https://www.youtube.com/watch?v=KB_ITKZm1Ts

3. Das Übersehen von wichtigen Informationen

https://www.youtube.com/watch?v=IGQmdoK_ZfY

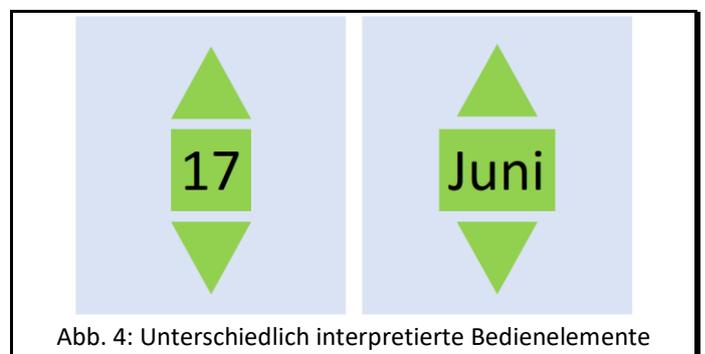
4. Bewusste Täuschung

<https://www.youtube.com/watch?v=GZGY0wPAnus>

Tabelle 1: Selektive Wahrnehmung – Beispiele

C. Mentale Modelle

Warum wissen Menschen, dass der Windows Button zu einem Menü führt, in dem ein Rechner der unter Windows läuft abgeschaltet werden kann? Diese und ähnliche Fragen werden selten gestellt. Subjektiv scheint es völlig klar zu sein, „dass dies so ist“. Hinter all diesem scheinbar so allgegenwärtigen klaren Wissen liegen sogenannte „Mentale Modelle“ [2]. Diese Mentalen Modelle beinhalten z.B. unser aus Erfahrung gewonnenes Wissen über die Funktionalität von Bedienelementen. Diese Mentalen Modelle sind überwiegend im unbewussten Teil unseres Gehirnes verankert, daher sind wir uns ihrer ständigen Unterstützung bei täglichen Tätigkeiten nahezu nicht bewusst.



In Abbildung 4 sind zwei sehr ähnliche Bedienelemente schematisch dargestellt. Die Interpretation der Bedeutung erfolgt aber nach zwei unterschiedlichen Mentalen Modellen. Stellen Sie sich die Frage, „Welche Zahl erscheint, wenn der

Pfeil nach oben, über der Zahl 17 gedrückt wird?“. Hier Antworten etwa 70% bis 80% der Menschen in der westlichen Welt mit „18“. Wenn Sie aber die dieselbe Frage bei der Anzeige des Monats Juni stellen, dann antworten ca. 50% mit „Mai“, die anderen „50%“ mit Juli. Für das Design von Interfaces bedeutet dieser zweite Fall, die Menschen wissen nicht was der Pfeil bewirken soll. Mentale Modelle sind höchst spezifisch und bereits kleine Änderungen des Nutzungskontextes, oder der Elemente kann völlig unerwartete Konsequenzen haben.

Ein weiteres wichtiges Phänomen ist die schnelle Akzeptanz von einmal verstandenen mentalen Modelle. Wenn eine Gruppe von Menschen ein Display entwickelt, dann ist bereits nach wenigen gemeinsamen Gesprächen ein Set an neuen mentalen Modellen aufgebaut. Diese Gruppe von Menschen hat in weiterer Folge große Schwierigkeiten zwischen Mentalen Modellen, die nur in ihrer Gruppe akzeptiert sind und anderen mentalen Modelle, die von Menschen auch außerhalb der Gruppe verstanden werden, zu unterscheiden. Oder anders ausgedrückt, ein Team, das Software entwickelt, verliert nach kurzer Zeit das Gefühl dafür, welche Teile von Benutzern verstanden werden und welche Teile nur Ratselraten auslösen.

III. ÜBERPRÜFUNG VON ANNAHMEN – DER USABILITY TEST

Bereits diese wenigen Beispiele zeigen, dass Annahmen über die Usability von Software niemals ohne Überprüfung als korrekt angesehen werden können. Die Usability muss während des Entwicklungsprozesses laufend überprüft werden. Personen, die in möglichst vielen Parametern den späteren Anwendern entsprechen, werden mit Aufgaben konfrontiert und es wird beobachtet, wie sie damit zurechtkommen. Usability Tests können sehr einfach organisiert und durchgeführt werden, erfordern aber spezifischen Know How in Bereich User Experience Design. Nur dann, wenn alle im Entwicklungsprozess eingebundenen Personen Verständnis für die Notwendigkeit von Tests, für die Verfahren zur Planung und Durchführung und für die korrekte Interpretation der Ergebnisse haben, können Usability Tests sinnvoll angewandt werden.

Grundsätzlich stehen drei völlig unterschiedliche Verfahren dazu zur Verfügung:

1. In House Test.
2. Remote Usability Tests.
3. Tests im User Experience Labor.

Alle Verfahren haben ihre eigenen spezifischen Vor- bzw. Nachteile und verursachen unterschiedliche Aufwände in der Planung, Durchführung und Interpretation [3].

Usability Tests müssen in jedem Fall von Personen mit einem tiefen Verständnis für Usability bzw. User Experience designed werden. Die Ergebnisse hängen direkt von der Qualität des Entwurfs der Tests ab. Mit geeigneten, oder besser gesagt ungeeigneten Entwürfen kann alles und nichts bewiesen werden.

Aber auch die Ergebnisse von ausgezeichnet entworfenen Usability Tests sind per se kaum aussagekräftig. Es ist unab-

dingbar, dass Personen, die den Blick auf die gesamte Software bewahren, die Ergebnisse in Hinblick darauf bewerten. Problematische Testergebnisse können, müssen aber keine Änderung des UX – Designs nach sich ziehen.

IV. UX DESIGN – EIN KREATIVER PROZESS

UX – Design ist ein kreativer Prozess. Das bedeutet aber in letzter Konsequenz, negative Testergebnisse an sich beinhalten keinesfalls einen Lösungsansatz für das gefundene Problem.

Diese einfache Tatsache verursacht in vielen Softwareprojekten Schwierigkeiten. Ein einfaches Beispiel dafür, ist in Abbildung 5 dargestellt.

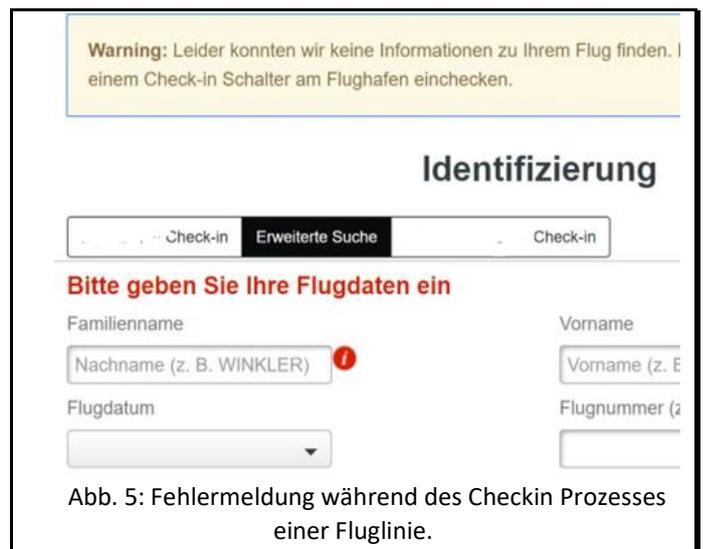


Abb. 5: Fehlermeldung während des Checkin Prozesses einer Fluglinie.

Eine Überprüfung der Verständlichkeit der Fehlermeldung, sowie der Abläufe während des Prozess ergab, die Mehrzahl der Benutzer sind zu diesem Zeitpunkt bereits verärgert über die schlechte Benutzerführung, die sie zu diesem Dialog geführt hat. Der Hauptgrund dafür ist, dass in diesem Dialog zum dritten Mal(!) die Buchungsnummer und der Familienname neu eingegeben werden muss. Eine zweite ernste Problematik ergibt sich aus dem Wortlaut der Fehlermeldung: „Warnung: Leider konnten wir keine Informationen zu Ihrem Flug finden.“ Diese Fehlermeldung wird häufig dahingehend interpretiert, dass mit der Buchung etwas nicht stimme. Dies führt unmittelbar zu überflüssigen Anrufen beim Helpdesk. Es ist für die Anwender nicht offensichtlich, dass die Buchung sehr wohl in Ordnung ist, dass lediglich der Zeitpunkt für den frühestens möglichen Check-in noch nicht erreicht ist.

Die Lösung des Problems könnte erfolgen, indem die Felder vorausgefüllt werden und die Fehlermeldung angepasst wird. Allerdings ist diese Lösung weit von einer optimalen User Experience entfernt. Es bleibt das Problem der drei überflüssigen Dialoge bestehen. Als Lösung könnte bereits zu Beginn der Suche, nach der für die Identifikation notwendigen Eingabe der Buchungsnummer und des Familiennamens, eine Information ausgegeben werden, die dem Anwender Aufschluss über diese Tatsachen gibt. Diese Information könnte zum Beispiel folgendermaßen lauten.

„Der Online Check-in öffnet in 3h 25 Minuten“, sowie „Ihre Buchungsdetails sind UX 789 von Wien nach Frankfurt am 27.11.2019 um 17:00“.

Selbstverständlich muss dieses Design wieder überprüft werden, da es sich ja lediglich um eine Annahme einer Verbesserung handelt. Üblicherweise wird bei einer derartigen neuen Überprüfung die Darstellung und das Wording nochmals verbessert.

Ein sehr gutes Beispiel für den kreativen Prozess des UX Designs stellen Wörterbücher dar. Ursprünglich waren Wörterbücher als kompakte Nachschlagwerke für Fragen rund um die Übersetzung von Worten gedacht. Mit der Verfügbarkeit von Smartphones wurden rasch auch Applikationen verfügbar, die kleine Wörterbücher sehr gut ersetzen konnten. Ein Beispiel für eine ausgezeichnete derartige Anwendung ist in Abbildung 6 dargestellt.

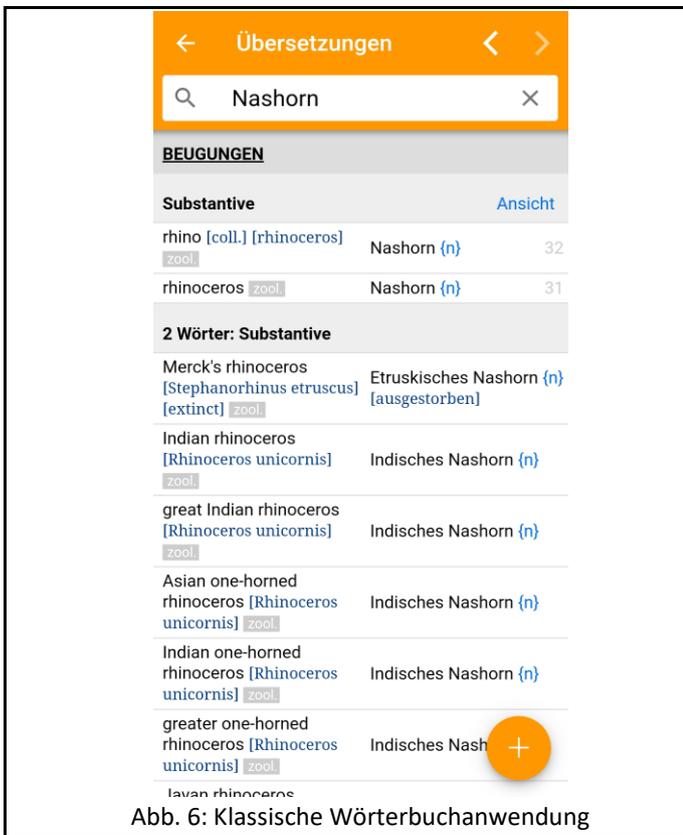


Abb. 6: Klassische Wörterbuchanwendung

Diese Anwendung imitiert das Verhalten eines Wörterbuches hervorragend und ersetzt in jedem Fall ein kleines Wörterbuch für die Reise.

Was geschieht nun, wenn aber die Frage untersucht wird, „Was benötigt der Anwender von einer Software, die übersetzt, und wie wird diese optimal eingesetzt“? Die Antwort führt sofort weit weg vom Wörterbuch. Es geht nämlich in den meisten Fällen nicht darum Worte zu übersetzen, sondern darum eine fremde Sprache zu verstehen, bzw. sich in dieser Sprache verständlich zu machen. Abbildung 7 zeigt eine Lösung, die diese Fragestellung gut beantwortet.

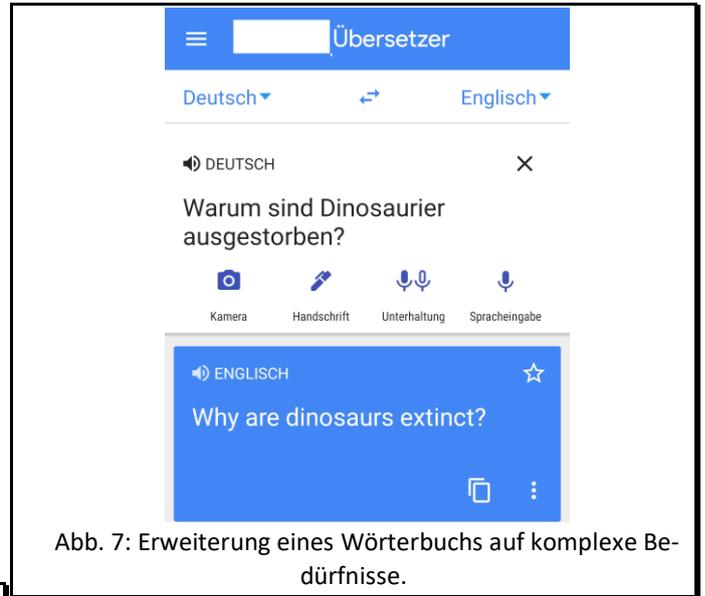


Abb. 7: Erweiterung eines Wörterbuchs auf komplexe Bedürfnisse.

In dieser Lösung können ganze Sätze recht passabel übersetzt werden. Besonders wichtig: der Satz kann auch in beiden Sprachen vorgelesen werden. Zusätzlich gibt es eine Funktion, mit der Text fotografiert und sofort übersetzt wird, sowie eine Funktion, die eine Art Konversation mit Übersetzung ermöglicht.

Der Weg von einem Wörterbuch zu der Anwendung in Abbildung 7 ist keinesfalls trivial und schon gar nicht offensichtlich. Dieser Weg kann nur mittels aufwändiger Analysen und in Teams kreativer Menschen gefunden werden. *Retrospektiv* betrachtet sind gute einfache Lösungen immer ein wenig mit dem Gefühl behaftet, „Sonnenklar, genau so stelle ich es mir vor“.

V. WIE KANN DIE USER EXPERIENCE GEMESSEN WERDEN?

Metriken für die User Experience sind ein unerschöpflicher Fundus an Anekdoten und aber auch für berechnete Fragen und Zweifeln. Wie kann etwas so radikal Subjektives wie das persönliche Erleben mit einer Software in objektive Metriken gebracht werden? Es existieren viele Metriken für UX und jede dieser Metriken hat ihre Stärken und ihre Schwächen. Ein Weg eine sinnvolle Metrik zu finden, ist die Orientierung an Lösungen führender Unternehmen im UX Design. Aus diesem Grund betrachten wir hier das Verfahren mit dem Google die User Experience seiner Anwender bewertet, das H.E.A.R.T Framework.

Dieses Framework basiert auf fünf einzelnen Metriken, die optimales User Experience beschreiben: happiness, engagement, adoption, retention, task success – abgekürzt H.E.A.R.T.

- **Happiness:** Wie zufrieden sind die Nutzer? Macht es ihnen Spaß mit der Site bzw. den Inhalten zu arbeiten?
- **Engagement:** Wie intensiv setzen sich die Nutzer mit den Inhalten auseinander? Wie lange bleiben sie auf der Site, wie viele Newsletter öffnen sie?
- **Adoption:** Wie viele neue Nutzer gibt es für eine Funktion oder Applikation? Oder, wie viele neue Abonnenten pro Zeiteinheit gibt es?

- **Retention:** Wie häufig kommen die Besucher wieder? Wie lange bleiben sie erhalten? Wann kündigen sie ihr Newsletter-Abo?
- **Task Success:** Können die Nutzer ihre Aufgaben abschließen? Gelingt ihnen das effizient? Machen sie dabei Fehler?

Ganz wichtig für H.E.A.R.T. ist, die Details müssen zwingend für die jeweilige Nutzung angepasst werden. H.E.A.R.T. ist daher keine Metrik, die in jedem Fall einen unmittelbaren direkten Vergleich zwischen zwei Applikationen erlaubt, aber HEART ist ausgesprochen gut geeignet, die Verbesserungen oder Verschlechterungen des Nutzererlebnisses aufzuzeigen [4], [5].

VI. UX DESIGN UND AGILE METHODEN (SCRUM)

SCRUM als die agile Methode zur Softwareentwicklung wird in zahlreichen Seminaren und an vielen Universitäten gelehrt und ist mittlerweile Basiswissen nahezu aller Software Ingenieure [10]. Allerdings sind in SCRUM Prozessen die Ziele Usability und User Experience keine explizit angepeilten Themen. Dies führt dazu, dass UX Designer häufig nicht vollständig in SCRUM Teams integriert werden. User Experience-Design und der SCRUM-Prozess existieren meist nur nebeneinander und haben wenige Berührungspunkte. Das UX-Design wird in vielen Fällen in einer Phase 0 erstellt und dem agilen Entwicklungsteam für die Umsetzung übergeben. Damit verursacht man allerdings ähnliche Probleme wie im klassischen Wasserfallmodell. Gelegentlich wird in so einem Fall von SCRUM-Waterfall gesprochen, ein Vorgehensmodell, das in jedem Fall zu vermeiden ist.

SCRUM ist ein inkrementelles Verfahren und optimiert die Umsetzung bei gleichzeitig möglichst geringem Aufwand.

In Abbildung 8 ist ein üblicher SCRUM Sprint dargestellt.

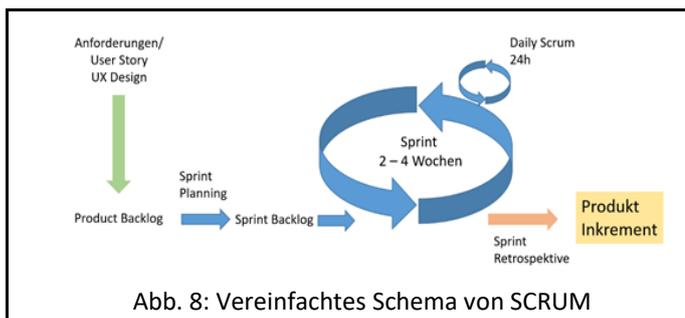


Abb. 8: Vereinfachtes Schema von SCRUM

Innerhalb eines SCRUM Sprints liegt das Augenmerk auf Umsetzung und nicht auf Design. Die meisten Sprints werden in relativ kurzen Zeiträumen von zwei bis maximal vier Wochen umgesetzt. Aus diese Gründen hat sich ein zeitlich versetztes Vorgehen bewährt [1], [6], [7].

Ein derartig verschachteltes Vorgehensmodell ist in Abbildung 9 dargestellt. Das UX Team entwickelt das grobe Design für den jeweils nächsten Sprint, gleichzeitig werden die Details des UX Designs für den gerade aktuellen Sprint *gemeinsam* mit den Softwareentwicklern erarbeitet.

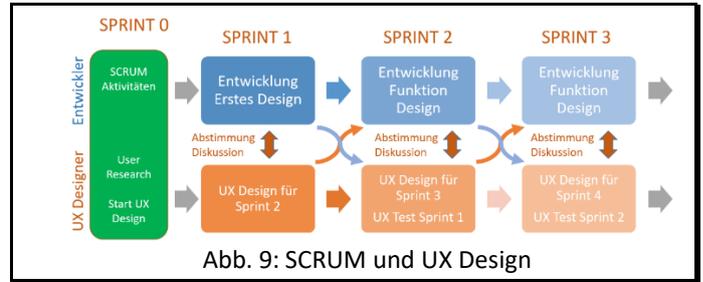


Abb. 9: SCRUM und UX Design

Ein häufiges Problem, das wir immer wieder bei der Zusammenarbeit von UX Teams und SCRUM Teams beobachten ist die unzureichende, oder völlig fehlende Integration der beiden Teams. Softwareentwickler und UX Designer arbeiten ohne ausreichende Abstimmung jeweils an ihren Aufgaben. Jedoch kommt es dadurch zu einem dramatischen Verlust an gemeinsam nutzbaren Wissen. Besonders wichtig ist daher, dass Details des UX Designs gemeinsam mit den Entwicklern innerhalb eines Sprints definiert werden. Eine aktive Teilnahme des UX Teams in allen SCRUM Meetings (Ceremonies) ist ebenso notwendig. Wenn dies nicht konsequent durchgeführt und auch beibehalten wird, besteht die Gefahr, dass u.a. folgende aus UX Sicht wichtigen Produktziele aus den Augen verloren werden:

- Das Gesamtbild, die Produktvision und die dadurch beabsichtigten Ergebnisse im gesehen Gesamtkontext.
- Durchführbarkeit und Praxistauglichkeit der UX Designs für das Entwicklungsteam.
- Tatsächliche Realisierung des Designs von Entwicklern in Code.
- Was bei den kommenden Sprints zu priorisieren ist.
- Die Chance, lediglich auf Meinungen basiertes Design durch benutzerzentrierten Untersuchungen zu optimieren.
- Gelegenheiten für die Entwickler die Ergebnisse der UX Tests wahrzunehmen, zu verstehen und darauf zu reagieren.

VII. GEMEINSAMES BASISWISSEN

Die beste Absicht zur Zusammenarbeit schlägt fehl, wenn das gemeinsam benötigte Basiswissen nicht vorhanden ist. Menschen, die zusammenarbeiten, müssen die gleiche Sprache sprechen, die gleichen Ausdrücke für dieselben Phänomene verwenden und die jeweils andere Gruppe in ihren Eigenheiten und taskbedingten Bedürfnissen verstehen.

Gerade UX Designer und Softwareentwickler leben häufig in verschiedenen Welten, umso wichtiger ist hier eine gemeinsame Sprache, die gegenseitiges Verstehen überhaupt erst ermöglicht [8], [9].

UX Teams müssen ein Grundwissen über Software Engineering besitzen und Softwareentwickler brauchen ein Grundwissen über User Experience Design, bzw. über Software Usability. Unserer Erfahrung nach kann das benötigte UX Wissen sehr gut über Zertifizierungsprogramme erreicht werden.

Im deutschsprachigen Raum weit verbreitet ist die Zertifi-

zierung zum *Certified Professional for Usability and User Experience – Foundation Level (CPUX - F)* des *International Usability and User Experience Qualification Boards (UXQB)*³.

Ein etwas umfangreicheres, stärker international ausgerichtetes Schema, das zusätzlich die Aspekte Wahrnehmungsphysiologie und den Softwareentwicklungsprozess beinhaltet, ist die Zertifizierung zum *Certified Professional for Usability Engineering and User Experience Design – Foundation Level (CPUE – FL)* des *User Experience Quality Certification Centers (UXQCC)*⁴.

Das benötigte Basiswissen im Bereich Software Engineering kann den UX Teams häufig im Unternehmen selbst vermittelt werden. Alternativ bieten sich hier auch kurze Ausbildungen an. Viele davon sind auf gängigen Plattformen (Udacity, Udemy, Coursera, edX, ..) einfach verfügbar.

VIII. CONCLUSION

Usability und User Experience (UX) haben sich von nice-to-have Faktoren zu entscheidenden Wettbewerbselementen entwickelt. Obwohl es scheinbar einfach ist, UX in Softwareprodukten zu optimieren, erweist sich die Umsetzung als komplex. UX Teams und Software Entwicklung Teams benötigen gegenseitiges Verständnis für die Probleme und Lösungswege der jeweils anderen, sowie ein gemeinsames Basiswissen. Werden der UX Design Prozess und agile Methoden des Software Engineerings konsequent zusammengeführt und ein gemeinsames Vorgehensmodell implementiert, dann gelingt die Entwicklung von Software, die von den Anwendern geschätzt wird.

REFERENCES

- [1] Verena Seibert-Giller und Robert Pucher, Usability und User Experience: Prüfungsvorbereitung zum Certified Professional for Usability Engineering and User Experience Design. UXKnowHow 2019, ISBN 978-3-9504772-0-7
- [2] Henrik Arndt, Mentale Modelle. In: Integrierte Informationsarchitektur. X.media.press. Springer 2019, Berlin, Heidelberg. Print ISBN978-3-540-24074-7
- [3] Nancy Davis (Herausgeber), Steve Krug (Autor), Mark Matcho (Illustrator), Rocket Surgery Made Easy: The Do-it-yourself Guide to Finding and Fixing Usability Problems. New Riders 2009, ISBN 0-321-65729-2
- [4] <https://www.youtube.com/watch?v=YugESrDVHuo> (aufgerufen 15.12.2019)
- [5] <https://library.gv.com/how-to-choose-the-right-ux-metrics-for-your-product-5f46359ab5be> (aufgerufen 15.12.2019)
- [6] <https://www.nngroup.com/articles/ux-scrum/> (aufgerufen 15.12.2019)
- [7] Hoa Loranger, Page Laubheimer, Jakob Nielsen. Effective Agile UX Product Development. 3rd Edition, Eigenverlag NNG 2019.
- [8] Cooper Alan, The Inmates Are Running the Asylum: Why High-tech Products Drive Us Crazy and How to Restore the Sanity, SAMS 2004
- [9] Robier Hannes, UX Redefined: Winning and Keeping Customers with Enhanced Usability and User Experience, Springer, 2015
- [10] Leyton Ryland, The Agile Business Analyst: Moving from Waterfall to Agile, Eigenverlag, 2015

³ <https://uxqb.org/de/zertifizierung/basiszertifizierung-cpux-f/>

⁴ <https://uxqcc.com/certificates/personal-certificates/foundation-level/>