

# Vorgehensmodell zum Einsatz von Storyboarding als Basistechnik für die kontext- und modellbasierte Ableitung von Interaction-Cases für ubiquitäre Systeme

Christine Keller, Romina Kühn, Thomas Schlegel

{Christine.Keller, Romina.Kuehn, Thomas.Schlegel}@tu-dresden.de

**Abstract:** Die Technik des Storyboardings etabliert sich als erfolgreiche Methode für „User-Centered Design“. Die bildliche Darstellung von Storyboards erleichtert die Erarbeitung von Szenarien und Anforderungen an ein System, da insbesondere in Gruppen von Stakeholdern mit technischem und nicht technischem Hintergrund oft eine gemeinsame Sprache fehlt. In ubiquitären Umgebungen finden oftmals hochkomplexe Interaktionen statt, zusätzlich spielt Kontext eine wichtige Rolle. In Storyboards können diese komplexen Elemente visuell erfasst werden. Modellgestützte Storyboards bieten eine Möglichkeit, ubiquitäre interaktive Systeme mit allen beteiligten Stakeholdern zu modellieren. Mit Hilfe von Interaction-Cases, die interaktive Prozesse modellieren helfen und sich leicht inkrementell verfeinern lassen, können Interaktionsabläufe von Storyboards abgeleitet werden. Wir stellen in dieser Arbeit eine Methode vor, die dazu geeignet ist, in interdisziplinären Teams mit Hilfe von Storyboards Kontexte für kontextsensitive ubiquitäre Systeme zu identifizieren und davon ausgehend kontextadaptive Interaction-Cases zu erarbeiten.

## 1 Einleitung

Die Interaktion zwischen Benutzer und System ist eine zentrale Komponente, auch und gerade für ubiquitäre Systeme. Solche Systeme bestehen häufig aus verschiedenen Einzelsystemen, die multimodale Interaktion ermöglichen. Oft werden hierzu innovative Interaktionsmechanismen eingesetzt, wie beispielsweise die Gestenerkennung oder Spracherkennung und -synthese. Hinzu kommt, dass ubiquitäre Systeme meist kontextsensitiv angelegt sind, da sie sich an Umgebungsbedingungen und den Benutzer anpassen und flexibel reagieren müssen, um wirklich allgegenwärtig zu sein und als System in den Hintergrund zu treten. Beim Entwurf innovativer, ubiquitärer und interaktiver Systeme ist darauf zu achten, dass die Vorgehensweise flexibel bleibt und die Möglichkeit bietet, nach dem Prinzip des „User-Centered Design“ den Benutzer und die Benutzerperspektive an verschiedenen Stellen im Entwicklungsprozess einzubinden, um die Benutzerfreundlichkeit des entstehenden Systems zu erhöhen. In der Entwicklung interaktiver Systeme kommen in frühen Phasen verschiedene Personen mit unterschiedlichem technischem Hintergrundwissen zusammen, um das System zu entwerfen. Hier müssen Sprach- und Wissensbarrieren überwunden werden, technische Details können und sollen in diesen Phasen noch keine große Rolle spielen. Daher wird mit informalen Methoden wie der Erstellung von

Personas und Szenarien versucht, eine gemeinsame Basis für die Anforderungserhebung zu finden. Personas und Szenarien werden in sehr vielen Softwareentwicklungsprozessen für verschiedenste Arten von Systemen als erfolgreiche Analyseinstrumente eingesetzt [BFJZ93]. Aus Personas und Szenarien können dann Anwendungsfälle (Use Cases) abgeleitet werden, die in strukturierter Art und Weise die typische Verwendung des Systems abbilden. Weiter formalisierte Methoden umfassen Klassendiagramme und formale Spezifikationen, die allerdings von beteiligten Personen ohne technischen Hintergrund schwer zu verstehen sind.

Seit einiger Zeit werden auch vermehrt Storyboards zum Erfassen und Abbilden von Interaktionssituationen zwischen Benutzern und Systemen angewandt. Mit Hilfe von Storyboards können einzelne Szenarien illustriert und die Umstände der Verwendung des Systems detailliert beschrieben werden [LM96]. Ein Storyboard kann dabei in verständlicher Weise nicht nur die direkte Interaktion eines Users mit einem System beschreiben, sondern auch die Umgebung und Umwelt des Benutzers sowie des Systems miteinbeziehen [THA06]. Storyboards als nichttechnisches Ausdrucksmittel eignen sich besonders gut für die Arbeit in interdisziplinären Teams oder auch für Evaluationen mit Benutzern, denn durch die grafische Darstellung der Szenerie werden technische Details auch für Laien verständlich. Storyboards sind sehr gut geeignet, um die komplexen Interaktionen mit ubiquitären Systemen abzubilden. Da ein Storyboard immer auch implizit die Gegebenheiten des Systems und der jeweiligen Situation abbildet, können daraus Beschreibungen kontextsensitiver Systeme abgeleitet werden. Um interaktive ubiquitäre und kontextsensitive Systeme zu entwerfen und umzusetzen, ist es nötig, sowohl mit informellen Methoden Anforderungen erheben und Designs evaluieren zu können, als auch diese schrittweise zu strukturieren und letztlich in eine Systemarchitektur umsetzen zu können. Wir schlagen daher vor, aufbauend auf Personas, Szenarien und Storyboards Use Cases und als Verfeinerung Interaction-Cases zum Entwurf und Design ubiquitärer Systeme einzusetzen. Interaction-Cases basieren auf Use Cases und dienen zur detaillierten Modellierung von Interaktionsabläufen [SR10]. Die Verwendung von Interaction-Cases unterstützt den iterativen Modellierungsprozess, indem die Interaction-Cases bereits auf relativ allgemeinem Niveau definiert und darauf aufbauend schrittweise konkretisiert werden können. Interaction-Cases können kontextadaptiv modelliert werden [SK11]. In ubiquitären Umgebungen ist die Interaktion zwischen Benutzer und System häufig kontextsensitiv, da solche Systeme meist multimodale Interaktion unterstützen und kontextbasiert verschiedene Interaktionskanäle anbieten können. Mit Hilfe von Interaction-Cases kann nun die Interaktion abhängig von verschiedenen Kontexten modelliert werden.

In dieser Arbeit beschreiben wir einen Ansatz mit dem Storyboards gestaltet und genutzt werden können, um Kontext und kontextsensitive Interaktion zunächst zu erkennen und zu identifizieren. Insbesondere für Personen mit nicht-technischem Hintergrund kann die Identifikation und Zuordnung von Kontexten für kontextsensitive Systeme schwierig sein, da das Konzept von Kontext sehr komplex ist. Wir schlagen daher vor, Storyboards als visuelles Hilfsmittel einzusetzen. Weiterhin stellen wir vor, wie darauf aufbauend kontextadaptive Interaction-Cases abgeleitet werden können. In Abschnitt 2 gehen wir auf aktuelle und wichtige Arbeiten auf den relevanten Gebieten ein. Wir stellen Personas, Szenarien, Storyboards und Use Cases vor und beschreiben kurz die Methodik der Inter-

aktionsmodellierung mit Hilfe von Interaction-Cases. Weiterhin beschreiben wir in Abschnitt 3 einen Entwicklungsprozess für interaktive kontextsensitive ubiquitäre Systeme, der von Personas über Szenarien und Storyboards sowohl die Erstellung von Use Cases als auch von Interaction-Cases unterstützt. In Abschnitt 4 folgt dann eine Beschreibung der Darstellung und Identifikation von Kontext in Storyboards und die Herleitung von kontextsensitiven Interaction-Cases. In Abschnitt 5 folgt die Zusammenfassung unserer Überlegungen sowie ein Ausblick auf die momentan laufenden Arbeiten. Das Thema und die bisherigen Arbeiten bieten aus unserer Sicht ein hohes Potenzial für Weiterentwicklungen, führen zu neuen Ideen und werfen neue Fragen auf, die ebenfalls in diesem Abschnitt vorgestellt werden.

## **2 Stand der Technik**

Ubiquitäre Systeme sind bereits seit einigen Jahren Forschungsgegenstand [Wei93]. Sie werden und wurden für verschiedenste Zwecke entwickelt, vom „Smart Office“ und „Smart Home“ über Systeme im öffentlichen Raum hin zu persönlichen ubiquitären Systemen, beispielsweise zum „Ambient Assisted Living“ (AAL) für ältere Menschen [BSP<sup>+</sup>06, DS08, FjCS11, JZS11]. Im Mittelpunkt solcher Systeme steht ihr Benutzer, für den die Nutzung des Systems möglichst intuitiv gestaltet werden soll, so dass das System selbst in den Hintergrund tritt und wirklich ubiquitär erscheint. Daher ist es sinnvoll, den Ansatz des „User-Centered Design“ für das Design ubiquitärer Systeme einzusetzen [Bød00, VMSC02, Aoy07]. Wichtige Analyseinstrumente im „User-Centered Design“ sind Personas, Szenarien und Use Cases. Es lassen sich mit ihrer Hilfe schrittweise Anforderungen an das System aus Benutzersicht erarbeiten und konkretisieren. In letzter Zeit wurden vermehrt Storyboards zur visuellen Darstellung der Interaktion zwischen Benutzer und System eingesetzt. Diese Methoden und dazu existierende Ansätze werden daher im Folgenden genauer beleuchtet. Ebenfalls beschrieben wird die Technik der Interaction-Cases, die aufbauend auf Use Cases entwickelt und weiterentwickelt wurden, um interaktive Systeme detaillierter und kontextsensitiv modellieren zu können. In Unterabschnitt 2.7 werden daher Interaction-Cases näher betrachtet.

### **2.1 Personas**

Der Begriff Persona wurde von Cooper [Coo99] eingeführt und umfasst fiktive Benutzer, die mit verschiedenen Eigenschaften, Zielen und Aufgaben belegt werden [BA02]. Während des Design-Prozesses wird versucht, sich an diesen Zielen und Aufgaben zu orientieren. Der Zweck dieses Konzepts liegt darin, dass Designer oder Entwickler den Nutzer während der Entwurfs- und Entwicklungsphase plastisch vor Augen haben und seine Anforderungen nicht aus den Augen verlieren. Die Entwicklung von Personas geschieht meist in Textform, sowohl als reiner Fließtext als auch in Form von Stichpunkten oder Karteikarten. Häufig werden auch Bilder der entsprechenden Personas gezeichnet, um einen direkten Bezug herbeizuführen [VdGvdV06].



Abbildung 1: Systemdesign mit Personas, Szenarien, Storyboards und Use Cases

## 2.2 Szenarien

Für die Entwicklung von Szenarien eignet es sich, Personas heranzuziehen, deren Handlungsweisen dann in konkreten Situationen beschrieben werden. Eine solche Situation kann beispielsweise den Einsatz eines Produktes zur Erfüllung einer Aufgabe umfassen [JA07]. Dabei stehen die einzelnen Ziele, die der Benutzer mit einem System erreichen möchte, im Vordergrund [Car00]. Um möglichst viele Aspekte zu berücksichtigen, werden Szenarien meist sehr ausführlich beschrieben. Dabei muss jedoch darauf geachtet werden, dass Szenarien trotz ihrer Informationsfülle übersichtlich und eindeutig bleiben.

## 2.3 Storyboards

Storyboards werden nicht mehr nur in der Film- und Werbeindustrie verwendet, sondern mittlerweile auch häufig für den Software-Entwicklungsprozess [vdL06]. Ein Storyboard besteht dabei aus einzelnen Bildern, Panels genannt, die bildlich eine Szene darstellen. Innerhalb eines Storyboards kann auf diese Weise eine Situation dargestellt und beschrieben werden. Textuelle Beschreibungen wie Personas und Szenarien haben den Nachteil einer Sprachbarriere: Entwickler, Kunden, Benutzer und andere Stakeholder haben häufig verschiedene Hintergründe und benutzen verschiedene Begriffe. Die visuelle Darstellung in Storyboards eignet sich dagegen, über Sprachbarrieren und Fachsprachen hinweg wichtige Anforderungen zu beschreiben. Sie dienen als Visualisierung von Szenarien oder auch als low-fidelity Prototyp zur ersten Evaluation des Systemdesigns mit Benutzern. In ubiquitären Systemen nimmt diese Möglichkeit einen besonders wichtigen Stellenwert ein, da komplexe Interaktionen und wichtige Umgebungsvariablen, die in einer ubiquitären Umgebung eine Rolle spielen können, visuell einfacher zu erfassen sind [DSL06, Bas08]. Da Storyboards einen großen Interpretationsumfang haben, eignen sie sich nicht nur für Entwickler sondern auch für die Arbeit mit dem potenziellen Benutzer, da dieser beispielsweise solche Skizzen leicht selbst anfertigen kann.

## 2.4 Modellbasierte Storyboards

Mit der Anwendung von Personas, Szenarien und Storyboards ergeben sich allerdings Medienbrüche bei der Überführung der erhobenen Anwendungen in ein Systemmodell. Die

informalen Artefakte werden nicht mit den entstehenden Modellierungen verknüpft und werden bei späteren Änderungen im Modell nicht angepasst. So geht ihr Nutzen über die Zeit der Entwicklung verloren. Daher gibt es Ansätze, diese informalen Artefakte und dabei insbesondere Storyboards, mit Hilfe von Annotationen direkt mit den entstehenden Modellen zu verknüpfen [HMLC10]. Haesen et al. haben beispielsweise Tools vorgestellt, die zur Erstellung und Annotation von Personas, Szenarien und Storyboards genutzt werden können [HdBM<sup>+</sup>11]. Die informalen Artefakte können dabei miteinander, aber auch mit Modellen verknüpft werden. Storyboards werden dabei gezielt zur Entwicklung von formalen Artefakten, in diesem Fall UsiXML, aus informalen Artefakten eingesetzt [LVM<sup>+</sup>04, LHO<sup>+</sup>10].

## 2.5 Use Cases

Bei Personas, Szenarien und Storyboards handelt es sich um informale Möglichkeiten zur Bestimmung von Anforderungen. Die Vielzahl der Informationen muss für eine Systementwicklung jedoch in eine strukturiertere Sprache transformiert werden, wofür sich Use Cases (Anwendungsfälle) eignen. Solche Anwendungsfälle sind in UML als Use Case Diagramme vorgesehen. Sie beschreiben das Verhalten eines Systems. Dabei werden die einzelnen Schritte, die zur Durchführung einer Aufgabe benötigt werden, betrachtet und in eine Reihenfolge gesetzt. Meist werden die Use Cases tabellarisch oder textuell beschrieben.

## 2.6 Kontextsensitive ubiquitäre Systeme

Ubiquitäre Systeme sind meist kontextsensitiv, das heißt sie reagieren auf bestimmte Faktoren in einer Situation, die sie als Kontext erkennen [PB03, CFJ03]. Erst diese Eigenschaft erlaubt es ubiquitären Systemen, den Alltag der Benutzer zu durchdringen und als System in den Hintergrund zu treten. Durch Anpassung an den Kontext des Benutzers kann dieser besser in seinen Aufgaben unterstützt werden. Insbesondere in ubiquitären Systemen im öffentlichen Raum ist die Miteinbeziehung von Kontext unerlässlich, da sich die Situationen, in denen die verschiedenen Benutzer das System verwenden, stark unterscheiden können. Es existieren viele verschiedene Definitionen und Beschreibungen von Kontext für kontextsensitive ubiquitäre Systeme [BDR07]. In unseren Arbeiten stützen wir uns auf die Definition von Dey und Abowd [DA00]:

*Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.*

Je nach System und Anwendung verändert sich der Kontext, der relevant ist und auf den reagiert werden muss. Die Kriterien, die für die Erfassung und Ausprägung von Kontext in einem System relevant sind, sollten bereits in der Analyse- und Entwurfsphase erarbeitet werden, da für die Modellierung eines kontextsensitiven Systems nicht auf eine zu

Grunde liegende Kontextmodellierung verzichtet werden kann. Für kontextsensitive ubiquitäre Systeme im öffentlichen Raum haben wir beispielsweise eine Basismodellierung für relevante Kontextkriterien entwickelt [KKS11]. In Abbildung 2 ist die erste Ebene dieser Basistaxonomie dargestellt. Unter *spatial context* fällt hier beispielsweise der Ort des Benutzers (*location*), der *interaction context* bezieht sich auf die Möglichkeiten des Benutzers, Eingaben zu tätigen oder Ausgaben wahrzunehmen. Hier kann beispielsweise modelliert werden, dass ein Benutzer blind ist. Diese Taxonomie von Kontextkriterien lässt sich während der Analyse- und Entwurfsphase für das konkret zu entwickelnde System konkretisieren und anpassen. So liegt frühzeitig eine erste Kontextmodellierung vor. Aufbauend auf dieser Kontextmodellierung kann dann das ubiquitäre interaktive System modelliert werden. Die Struktur der Kontextmodellierung erlaubt dabei die parallele schrittweise Verfeinerung der Kontext- sowie der Systemmodellierung.

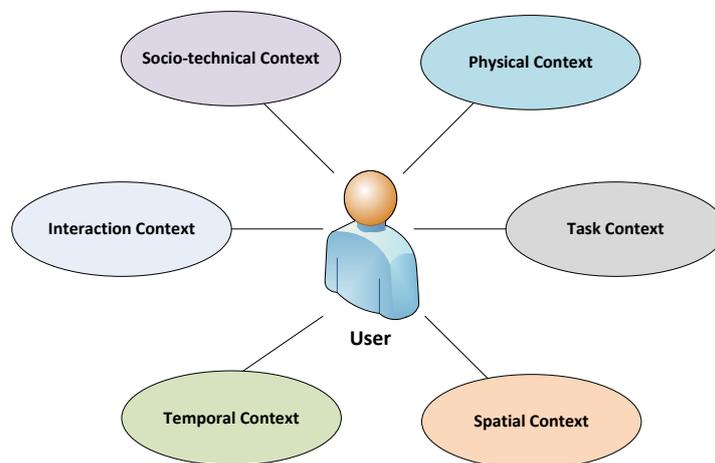


Abbildung 2: Eine Kontext-Taxonomie für ubiquitäre Systeme [KKS11]

## 2.7 Interaction-Cases zur Modellierung von Interaktion in ubiquitären kontextsensitiven Systemen

Use Cases bieten bereits eine Möglichkeit zur semi-formalen Modellierung des Systemverhaltens. Die konkreten Interaktionsabläufe zwischen System und Benutzer werden in Use Cases allerdings mit nur wenig Detailtiefe textuell beschrieben. Die einzelnen Interaktionsschritte zwischen Benutzer und System werden so nicht sehr genau abgebildet. Für die Modellierung von ubiquitären Umgebungen, die häufig multimodal und kontextsensitiv sind und damit hochkomplexe Interaktionsmöglichkeiten bieten, reicht die Abbildungsmöglichkeit für Interaktionen mit Use Cases nicht aus.

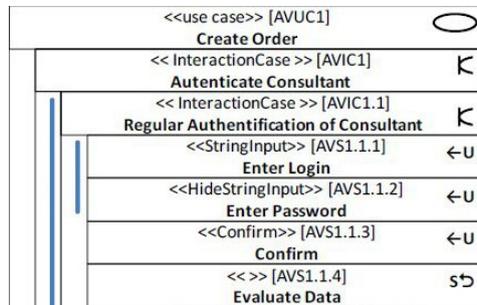


Abbildung 3: Teil eines Interaction-Cases

Zur strukturierteren Modellierung von Interaktionsabläufen wurden daher die Interaction-Cases entwickelt [SR10]. Interaction-Cases sind modular gehalten. So kann ein Interaction-Case aus mehreren Interaction-Cases bestehen, während das atomare Bestandteil eines Interaction-Cases ein Interaction-Step ist. In Abbildung 3 ist ein Teil eines Interaction-Cases zu sehen. Wie erkennbar ist, wird ein Interaction-Case direkt in einen Use Case integriert. Durch ihren modularen Aufbau lassen sich Interaction-Cases schrittweise konkretisieren und daher gut auch in iterativen Vorgehensmodellen einsetzen. In frühen Iterationen können zunächst einzelne große Interaction-Cases definiert werden, die in späteren Schritten dann bis hin zu einzelnen Interaction-Steps verfeinert werden können. Durch die Nutzung von Interaction-Case Bibliotheken können Interaction-Steps und Interaction-Cases, die häufig vorkommen, wiederverwendet werden, was die Erstellung von Interaction-Cases deutlich vereinfacht. Den Interaction-Cases liegt dabei ein objektorientiertes Typkonzept zu Grunde, das es ermöglicht, Interaction-Case Typen zu definieren und davon auch verfeinerte Typen von Interaction-Cases abzuleiten.

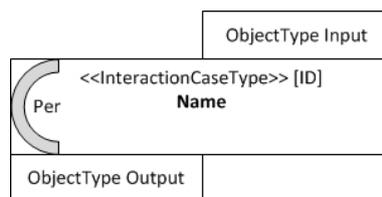


Abbildung 4: Kontextadaptiver Interaction-Case

Als Weiterentwicklung können Interaction-Cases auch kontextadaptiv gestaltet werden [SK11]. Hierzu kann an einem Interaction-Case ein Kontextmodifikator angebracht werden, der anzeigt, dass dieser Interaction-Case kontextsensitiv ist. In Abbildung 4 ist ein kontextadaptiver Interaction-Case dargestellt. Der Kontextmodifikator ist ein links angebrachter Halbkreis, in dessen Mitte der Bezeichner des Kontexts notiert wird, für dessen Ausprägungen der Interaction-Case adaptiert wird. Ein mögliches Beispiel hierfür ist das Anpassen von Aus- und Eingabe auf einen Audiokanal, für den Fall, dass der Benutzer als blind erkannt wird.

### 3 Von Storyboards zu Interaction-Cases im Entwicklungsprozess für ubiquitäre Systeme

In ubiquitären Systemen tritt das System selbst in den Hintergrund, die Aufgaben, die der Benutzer mit Hilfe des Systems erledigen möchte, treten dafür in den Vordergrund. In ubiquitären Systemen steht keine einheitliche Benutzungsoberfläche mit definierten und restriktiven Interaktionskonzepten zur Verfügung, vielmehr wird die Umgebung des Benutzers selbst auf vielfältige Weise interaktiv. Es existiert keine singuläre Benutzungsschnittstelle, es sind stattdessen viele verschiedene Schnittstellen zum System vorhanden. Daher können für ubiquitäre Systeme keine einfachen Prototypen der Benutzungsschnittstellen erstellt werden, wie es für grafische Benutzungsschnittstellen bereits mit vielfältiger Toolunterstützung möglich ist. Ubiquitäre Systeme reagieren auf verschiedene Eingaben des Benutzers, die in verschiedenen Modi geschehen können, so sind beispielsweise die Integration von Gestenerkennung und Touchinterfaces denkbar. Ebenso reagieren ubiquitäre Systeme auch auf die Umgebung und die konkrete Situation des Benutzers, sind also kontextsensitiv. Da textuelle Beschreibungen immer abstrahieren und linearisieren, lassen sich diese Elemente ubiquitärer Systeme nur schwer in ausreichender Genauigkeit durch beispielsweise Szenarien abbilden. Durch die visuelle Darstellung der Interaktion mit dem System sind ubiquitäre Interaktionsmechanismen einfacher zu erfassen als in einer textuellen Beschreibung. Zusätzlich ermöglichen Bilder die Darstellung der Umgebung des Benutzers und seiner Reaktionen auf das System, sowie der vielfältigen Reaktionen des Systems auf Benutzer und Umgebung.

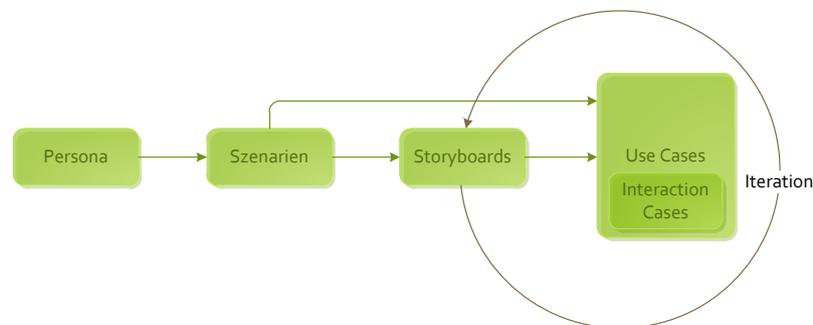


Abbildung 5: iterative Vorgehensweise

Storyboards eignen sich daher gut zur Integration in einen Entwicklungsprozess für ubiquitäre Systeme. Da in Storyboards die wichtigen Details der Interaktion zwischen Benutzer und System gut dargestellt werden können, eignen Storyboards sich auch gut dazu, daraus Interaction-Cases abzuleiten. Storyboards können als low-fidelity Prototyp des ubiquitären Systems gesehen werden und eignen sich so auch zur Evaluation mit Benutzern. Hierdurch fügen sie sich gut in einen iterativen Entwicklungsprozess ein, ebenso wie Interaction-Cases iterativ entwickelt werden können.

Stilmittel	Anhaltspunkte zum Einsatz
Einsatz von Text	nicht zu viel $\Rightarrow$ bis zu zwei Zeilen pro Panel zur Erläuterung von komplexen Vorgängen
Detailgrad	einfachere Storyboards zum Fokus auf das Wesentliche
Anzahl der Panels	optimal sind zwischen 3 und 6 pro Szene
Ablauf von Zeit	nur wenn nötig explizit darstellen
Dargestellte Szene	möglichst atomare, kurze Szenen

Tabelle 1: Anhaltspunkte für die Erstellung von Storyboards

Für diesen Entwicklungsprozess wurde eine Vorgehensweise entwickelt, die auch in Abbildung 6 dargestellt ist. Für jedes Projekt muss identifiziert werden, welche Stakeholder in der Analysephase einzubinden sind. Dies können je nach Art des Projektes Kunden, Designer, Entwickler, Benutzer oder andere Personenkreise sein. In dieser Gruppe müssen zunächst wichtige Szenarien für das zu entwickelnde System identifiziert werden. Dies kann, wie im „User-Centered-Design“ üblich, auf Basis von zuvor entwickelten Personas geschehen. Als ersten Schritt werden dann einzelne Szenarien in der Gruppe bearbeitet und daraus Storyboards für diese Szenarien erstellt. Für die Erstellung von nützlichen Storyboards können gewisse Anhaltspunkte gegeben werden, aufbauend auf der Evaluation von Truong et al. [THA06]. Diese Anhaltspunkte sind in Tabelle 1 angegeben. In weiteren Schritten werden diese Storyboards dann analysiert. Hierzu eignet sich die Arbeit in der Gruppe ebenso wie die Analyse der Storyboards von einzelnen Beteiligten, wobei dann eine Konsolidierung und Zusammenstellung der identifizierten Entitäten im Team folgen sollte. Aus den Storyboards identifiziert werden sollten:

1. Benutzer und Rollen
2. interaktive und passive Objekte
3. Aufgaben des Benutzers
4. Interaktionen

In diesem Schritt können die identifizierten Elemente durch unterstützende Werkzeuge in den Storyboards mit Semantik hinterlegt werden, so dass die einzelnen Elemente formalisiert und mit dem Storyboard verknüpft werden. Ein Werkzeug, das die Annotation von Storyboards für ubiquitäre Systeme unterstützt, befindet sich momentan in Entwicklung. Bei der Identifikation von Benutzern und Rollen ist die Verknüpfung mit eventuell zuvor entwickelten Personas vorstellbar, wie auch Haesen et al. dies umsetzen [HMLC10]. Die Identifikation der Benutzer und Rollen erfolgt, indem die Protagonisten des Storyboards analysiert werden. Welche dieser Protagonisten nehmen welche Rollen in Bezug auf das System ein und wie handeln sie darauf aufbauend? In einem weiteren Schritt werden die Objekte im Storyboard analysiert. Welche Objekte sind dargestellt und welche dieser Objekte werden interaktiv verwendet, d.h. sind Teil eines Interaktionsschrittes? Die Aufgaben des Benutzers können zunächst relativ grobgranular festgestellt werden. Hier wird das Storyboard nach den Zielen analysiert, die die Protagonisten verfolgen. Da ein einzelnes

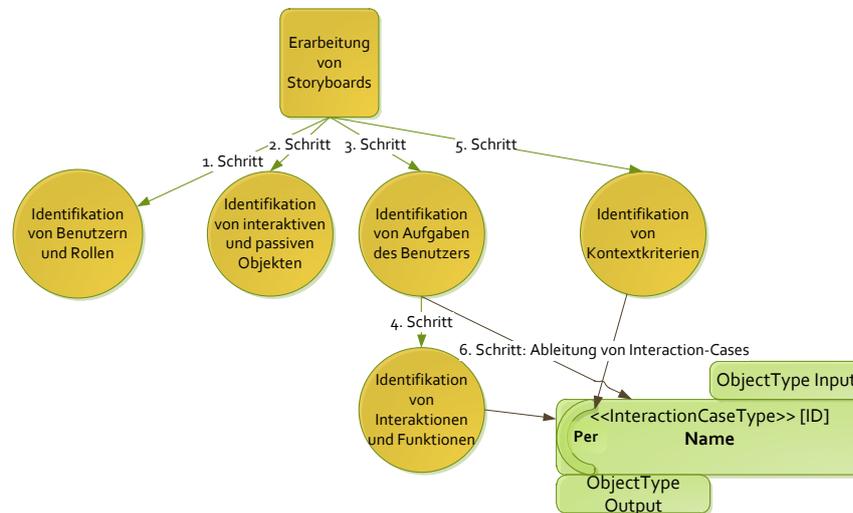


Abbildung 6: Vorgehensweise zur Ableitung von kontextsensitiven Interaction-Cases aus Storyboards

Storyboard eine atomare Szene beschreiben soll (vgl. Tabelle 1), hat jeder Protagonist im Storyboard idealerweise ein Ziel oder eine Aufgabe. Im nächsten Schritt sollen diese Aufgaben dann auf einzelne Funktionen verfeinert werden, die der Benutzer verwenden kann und es werden die Interaktionen identifiziert, derer er sich bedient. Hierzu werden die einzelnen Interaktionsschritte beleuchtet, die der Benutzer geht, um sein Ziel zu erreichen. Ebenso werden die Interaktionen von Seiten des Systems beleuchtet. Diese Schritte sind meist auf die einzelnen Panels des Storyboards aufgeteilt, da in jedem Panel nur eine einzelne Situation beschrieben werden kann. Für jedes einzelne Panel kann daher in diesem Schritt analysiert werden, wie der Benutzer agiert oder reagiert und wie das System agiert oder reagiert. In einem weiteren Schritt werden dann die Gegenstände, die für diese Interaktionen verwendet werden, mit den Funktionen verknüpft, für die sie benutzt werden. Rollen, Aufgaben und Interaktionen können nun in Use Cases und Interaction-Cases umgesetzt werden. Dabei sind einzelne Interaktionsschritte bereits direkt auf Interaction-Cases abbildbar. Die Verfeinerung der Aufgaben auf feingranulare Funktionen und Interaktionsschritte des Benutzers kann iterativ erfolgen, ebenso wie die Erarbeitung und der Abgleich der Use Cases und Interaction-Cases iterativ ablaufen kann, wie in Abbildung 5 dargestellt.

#### 4 Kontext in Storyboards

Zur Analyse der grundlegenden Vorgehensweise von Stakeholdern und Nutzern bei der freien Erstellung von Storyboards wurden Studenten beauftragt, ein vorgegebenes Szenario

in ein Storyboard zu überführen. Als Anhaltspunkt dienten den Studenten die in Tabelle 1 dargestellten Regeln zur Erstellung von Storyboards, die auf den Vorschlägen von Truong et al. basieren [THA06].

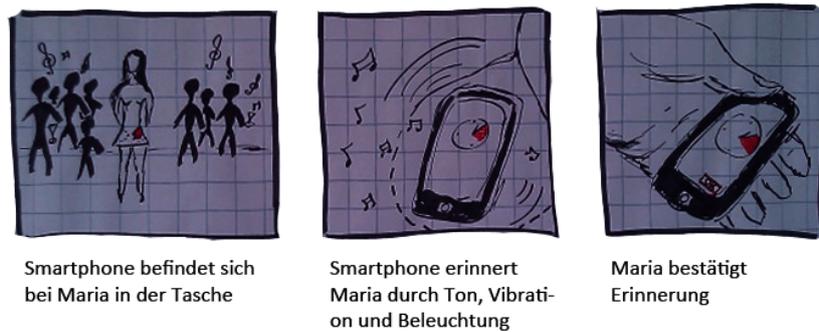
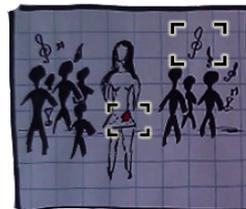


Abbildung 7: Beispiel für ein Storyboard

Das Szenario betrifft eine Applikation für Smartphones, die für den öffentlichen Personennahverkehr eingesetzt werden soll. Die handelnde Person ist eine Studentin, die abends eine Bar besucht. Um den letzten Bus nach Hause nicht zu verpassen, stellt sie ihre Smartphone-Applikation so ein, dass sie rechtzeitig daran erinnert wird, die Bar zu verlassen und zur Bushaltestelle zu gehen. In Abbildung 7 wird das Ergebnis einer Studentengruppe dargestellt.



Smartphone befindet sich bei Maria in der Tasche

Abbildung 8: Markierung von Kontext innerhalb eines Storyboards

Wie man erkennen kann, wird im ersten Panel die laute Umgebung der Bar bildlich dargestellt. Ebenso ist erkennbar, dass sich das Smartphone in der Tasche der Akteurin befindet. Im zweiten Panel wird durch die Darstellung und den angegebenen Text beschrieben, dass das Smartphone sowohl mit Ton, Beleuchtung als auch Vibration auf sich aufmerksam macht. Daraufhin wird dargestellt, wie die Studentin die Erinnerung bestätigt und dabei angezeigt bekommt, wie viel Zeit ihr zum Erreichen des Busses noch bleibt. Bereits in

diesem Stadium eines Storyboards wird deutlich, dass Kontext implizit miteinbezogen und abgebildet wurde, ohne dass dies in der Aufgabenstellung gefordert worden war.

Diese Darstellung von Kontextausprägungen kann nun dazu benutzt werden, Kontextkriterien zu identifizieren, die das zu gestaltende System in diesen Situationen beeinflussen. Hierzu wird das oben beschriebene Vorgehen erweitert. Nach der Identifikation der Rollen, Objekten, Aufgaben und Interaktionen bzw. Funktionen, die im Storyboard vorkommen, werden in einem weiteren Schritt Elemente im Storyboard identifiziert, die Kontextkriterien anzeigen oder auf für das System relevante Kontextkriterien hinweisen. Im vorliegenden Beispiel sind dies beispielsweise die Notenschlüssel und Noten, die eine laute Umgebung andeuten. Sind diese Kontextkriterien identifiziert, können sie im Storyboard markiert werden. Abbildung 8 verdeutlicht, wie dies in einer entsprechenden Werkzeugumgebung möglich sein könnte. Der nächste Schritt umfasst dann die Ermittlung derjenigen Kontextkriterien, die die Funktionen und Interaktionen zwischen System und Benutzer beeinflussen. Im Falle des Beispiels ist dies die laute Umgebung, die bewirkt, dass das Smartphone mit Vibrationsalarm und Beleuchtung reagiert, nicht nur mit akustischen Hinweisen. Die entsprechenden Kontextkriterien werden nun mit Annotationen versehen. Hierbei wird identifiziert, um welche Art Kontext es sich handelt. Um diesen Schritt zu unterstützen, steht eine vormodellierte Kontext-Taxonomie zur Verfügung, wie in Abschnitt 2.6 beschrieben. Um die Identifikation von relevanten Kontextkriterien zu vereinfachen, schlagen wir vor, für die Kontextkriterien in der hinterlegten Taxonomie einfach erkennbare Icons zu entwickeln, die dann wiederum zur Annotation des Storyboards benutzt werden können, wie in Abbildung 9 gezeigt.



Abbildung 9: Identifizierung von Kontext innerhalb eines Storyboards

## 5 Zusammenfassung und Ausblick

Storyboarding eignet sich gut zur Unterstützung der Entwicklung von ubiquitären Systemen. Storyboards können eingesetzt werden, um erste low-fidelity Prototypen eines Systems zu erstellen und um Anforderungen und erste Interaktionsabläufe für das zu entwickelnde System zu erarbeiten. Die bildliche Darstellung von Storyboards eignet sich speziell zur Arbeit in interdisziplinären Teams, da keine Sprachbarriere besteht. Auch können multimodale und kontextsensitive Interaktionsabläufe visuell einfacher beschrieben werden als textuell. Um aus den informalen Storyboards den ersten Schritt in Richtung Systemmodellierung zu unterstützen, stellten wir in dieser Arbeit eine Vorgehensweise vor, wie aus Storyboards Use Cases und Interaction-Cases abgeleitet werden können, die die Interaktion zwischen Benutzer und System modellieren. Die Ableitung der Use Cases und

Interaction-Cases aus Storyboards kann iterativ erfolgen und so einfach zum Beispiel in agile Prozessmodelle zur Entwicklung ubiquitärer Systeme eingebunden werden. Wir entwickeln derzeit eine Werkzeugunterstützung für diese Vorgehensweise, die die Erstellung und Annotation von Storyboards erleichtern soll. Hierbei sollen die Identifikationsschritte in der beschriebenen Vorgehensweise durch die Möglichkeit der Annotation der Storyboards unterstützt werden. Sind die in den Storyboards identifizierten Elemente entsprechend in einem Werkzeug annotiert worden, ist dann mit Hilfe der Bibliothek von Interaction-Case Typen die semi-automatische Ableitung von Interaction-Cases denkbar. Die Integration einer Kontext-Taxonomie und der Möglichkeit der Annotation von Storyboards mit vordefinierten oder abgeleiteten Kontextkriterien ist ebenfalls vorgesehen und wird die Erarbeitung kontextadaptiver Interaction-Cases unterstützen, insbesondere da Kontext für Beteiligte mit nicht-technischem Hintergrund und häufig auch für Beteiligte mit technischem Hintergrund ein schwerer zu fassendes Konzept darstellt. Werkzeugunterstützung, Modellbasierung und Standardisierung ermöglicht auch technisch und analytisch nicht versierten Benutzern die Teilnahme an der komplexen Entwicklung ubiquitärer Systeme und vereinfacht gerade auch die Erarbeitung von Kontexten und kontextsensitiver Interaktion. Ein Werkzeug, das die Ableitung von Use Cases und Interaction-Cases aus Storyboards mit Hilfe von Annotationen unterstützt, stellt auch sicher, dass spätere Änderungen sowohl an den Storyboards als auch den Interaction-Cases nachgezogen werden können und so die Artefakte stets auf dem neuesten Stand bleiben. Auch ist es hierdurch möglich, neue Interaktionen einzuführen, indem neue Storyboards erstellt und mit bestehenden Interaction-Cases verknüpft und nach Bedarf weitere Interaction-Cases abgeleitet werden. Hierzu ist momentan eine Benutzerstudie in Durchführung, bei der die Basiselemente für Annotationen in Storyboards, sowie die Vorgehensweise von Gruppen zur Erstellung von Storyboards untersucht werden. Ein entsprechendes Werkzeug wird darauf aufbauend entwickelt. Hiermit soll es zukünftig möglich werden, Interaktionen und Kontexte bereits frühzeitig mit allen Beteiligten auf gut verständliche Art und Weise zu erstellen und in die modellbasierte und inkrementelle Entwicklung zu integrieren.

## Danksagung

Wir danken Mandy Korzetz, Susann Struwe und Bianca Zimmer, sowie den Studenten der Vorlesung „Software Engineering ubiquitärer Systeme“ im Wintersemester 2010/2011 für ihre Unterstützung bei der Vorgehensanalyse zum Storyboarding.

## Literatur

- [Aoy07] Mikio Aoyama. Persona-Scenario-Goal Methodology for User-Centered Requirements Engineering. *Requirements Engineering, IEEE International Conference on*, 0:185–194, 2007.
- [BA02] Åsa Blomquist and Mattias Arvola. Personas in action: ethnography in an interaction design team. In *Proceedings of the second Nordic conference on Human-computer*

*interaction*, NordiCHI '02, pages 197–200, New York, NY, USA, 2002. ACM.

- [Bas08] Mark Baskinger. Cover Story - Pencils before pixels: a primer in hand-generated sketching. *Interactions*, 15(2):28–36, 2008.
- [BDR07] Matthias Baldauf, Schahram Dustdar, and Florian Rosenberg. A Survey on Context-Aware Systems. *International Journal of Ad Hoc and Ubiquitous Computing*, 2(4):263–277, 2007.
- [BFJZ93] Kevin Benner, Martin S. Feather, W. Lewis Johnson, and Lorna A. Zorman. Utilizing Scenarios in the Software Development Process. 1993.
- [Bød00] S. Bødker. Scenarios in user-centred design—setting the stage for reflection and action. *Interacting with Computers*, 13(1):61 – 75, 2000.
- [BSP<sup>+</sup>06] F. Bagci, H. Schick, J. Petzold, W. Trumler, and T. Ungerer. The reflective mobile agent paradigm implemented in a smart office environment. *Personal Ubiquitous Comput.*, 11:11–19, October 2006.
- [Car00] J. M. Carroll. Five reasons for scenario-based design. *Interacting with Computers*, 13(1):43 – 60, 2000.
- [CFJ03] Harry Chen, Tim Finin, and Anupam Joshi. An ontology for context-aware pervasive computing environments. *Knowl. Eng. Rev.*, 18(3):197–207, 2003.
- [Coo99] Alan Cooper. *The Inmates Are Running the Asylum: Why High-Tech Products Drive Us Crazy and How to Restore the Sanity*. Sams Publishing, 1999.
- [DA00] Anind K. Dey and G. D. Abowd. Towards a better understanding of context and context-awareness. In *Computer Human Interaction 2000 Workshop on the What, Who, Where, When, Why and How of Context-Awareness*, 2000.
- [DS08] Maria Danninger and Rainer Stiefelwagen. A context-aware virtual secretary in a smart office environment. In *Proceeding of the 16th ACM international conference on Multimedia*, MM '08, pages 529–538, New York, NY, USA, 2008. ACM.
- [DSLL06] Steven Dow, T. Scott Saponas, Yang Li, and James A. Landay. External representations in ubiquitous computing design and the implications for design tools. In *DIS '06: Proceedings of the 6th conference on Designing Interactive systems*, pages 241–250, New York, NY, USA, 2006. ACM.
- [FjCS11] Marcus Foth, Jaz Hee jeong Choi, and Christine Satchell. Urban informatics. In *Proceedings of the ACM 2011 conference on Computer supported cooperative work*, CSCW '11, pages 1–8, New York, NY, USA, 2011. ACM.
- [HdBm<sup>+</sup>11] Mieke Haesen, Jan Van den Bergh, Jan Meskens, Kris Luyten, Sylvain Degrandart, Serge Demeyer, and Karin Coninx. Using Storyboards to Integrate Models and Informal Design Knowledge. In Heinrich Hussmann, Gerrit Meixner, and Detlef Zuehlke, editors, *Model-Driven Development of Advanced User Interfaces*, volume 340 of *Studies in Computational Intelligence*, pages 87–106. Springer, Berlin / Heidelberg, 2011.
- [HMLC10] Mieke Haesen, Jan Meskens, Kris Luyten, and Karin Coninx. Draw Me a Storyboard: Incorporating Principles and Techniques of Comics to Ease Communication and Artefact Creation in User-Centred Design. In *24th BCS Proceedings of the Conference on Human Computer Interaction (HCI 2010)*, 2010.

- [JA07] Maria Johansson and Mattias Arvola. A case study of how user interface sketches, scenarios and computer prototypes structure stakeholder meetings. In *Proceedings of the 21st British HCI Group Annual Conference on People and Computers: HCI...but not as we know it - Volume 1*, BCS-HCI '07, pages 177–184, Swinton, UK, UK, 2007. British Computer Society.
- [JZS11] Antonio J. Jara, Miguel A. Zamora, and Antonio F. Skarmeta. An internet of things—based personal device for diabetes therapy management in ambient assisted living (AAL). *Personal Ubiquitous Comput.*, 15:431–440, April 2011.
- [KKS11] Romina Kühn, Christine Keller, and Thomas Schlegel. A Context Taxonomy Supporting Public System Design. In *Proceedings of the 1st International Workshop on Model-based Interactive Ubiquitous Systems, EICS, to appear*, 2011.
- [LHO<sup>+</sup>10] Kris Luyten, Mieke Haesen, Dawid Ostrowski, Karin Coninx, Sylvain Degrandart, and Serge Demeyer. On stories, models and notations: Storyboard creation as an entry point for model-based interface development with UsiXML. In *Proceedings of the User Interface eXtensible Markup Language workshop (UsiXML-EICS 2010)*, 2010.
- [LM96] James A. Landay and Brad A. Myers. Sketching storyboards to illustrate interface behaviors. In *Conference companion on Human factors in computing systems: common ground*, CHI '96, pages 193–194, New York, NY, USA, 1996. ACM.
- [LVM<sup>+</sup>04] Quentin Limbourg, Jean Vanderdonckt, Benjamin Michotte, Laurent Bouillon, and Víctor López-Jaquero. USIXML: A Language Supporting Multi-path Development of User Interfaces. In Rémi Bastide, Philippe A. Palanque, and Jörg Roth, editors, *EHCI/DS-VIS*, volume 3425 of *Lecture Notes in Computer Science*, pages 200–220. Springer, 2004.
- [PB03] Paul Prekop and Mark Burnett. Activities, context and ubiquitous computing. *Computer Communications*, 26(11):1168–1176, 2003.
- [SK11] Thomas Schlegel and Christine Keller. Model-based Ubiquitous Interaction Concepts and Contexts in Public Systems. In *Proceedings of the 14th International Conference on Human-Computer Interaction, to appear*, 2011.
- [SR10] Thomas Schlegel and Michael Raschke. Interaction-Cases: Model-Based Description of Complex Interactions in Use Cases. In *Proceedings of IADIS International Conference Interfaces and Human Computer Interaction*, Freiburg, Germany, 2010.
- [THA06] Khai N. Truong, Gillian R. Hayes, and Gregory D. Abowd. Storyboarding: an empirical determination of best practices and effective guidelines. In *Proceedings of the 6th conference on Designing Interactive systems*, pages 12–21. ACM, 2006.
- [VdGvdV06] Dhaval Vyas, Spencer de Groot, and Gerrit C. van der Veer. Understanding the academic environments: developing personas from field-studies. In *Proceedings of the 13th European conference on Cognitive ergonomics: trust and control in complex socio-technical systems*, ECCE '06, pages 119–120, New York, NY, USA, 2006. ACM.
- [vdL06] Corrie van der Lelie. The value of storyboards in the product design process. *Personal and Ubiquitous Computing*, 10(2-3):159–162, 2006.
- [VMSC02] Karel Vredenburg, Ji-Ye Mao, Paul W. Smith, and Tom Carey. A survey of user-centered design practice. In *Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves*, CHI '02, pages 471–478, New York, NY, USA, 2002. ACM.

[Wei93] Mark Weiser. Some Computer Science Issues in Ubiquitous Computing. *Communications of the ACM*, 36(7):75–84, July 1993.