

# mConcAppt Methode

## UX und Interaktionsdesign für mobile Business Apps



### Steffen Hess

Fraunhofer IESE  
Fraunhofer-Platz 1  
67663 Kaiserslautern  
steffen.hess@iese.fraunhofer.de

### Felix Kiefer

Fraunhofer IESE  
Fraunhofer-Platz 1  
67663 Kaiserslautern  
felix.kiefer@iese.fraunhofer.de

### Abstract

Unternehmen haben längst das Potential von mobilen Anwendungen (Apps) erkannt. Gerade mobile Mitarbeiter können ihre Effizienz und den Komfort ihrer Workflows durch die Nutzung von Apps steigern. Die Entwicklung solcher Business Apps ist mit speziellen Herausforderungen verbunden. Beispielsweise hohe Usability und User Experience, kurze Produkteinführungszeit, klar definierter Funktionsumfang, Integration in die bestehende IT Infrastruktur und die Berücksichtigung verschiedener Geräte und Geräteplattformen. Diese Herausforderungen müssen schon während der Konzeption von Business Apps durch Usability Professionals adressiert werden.

Das Tutorial vermittelt anhand eines durchgängigen Beispiels die Anwendung der mConcAppt Methode unter der Benutzung von praktisch erprobten Templates. Die Methode wurde bereits mehrfach bei der Erstellung von konkreten Business Apps angewendet und hat als Resultat ein fertiges Interaktionskonzept. Teilnehmer des Tutorials erlernen unter anderem die Erstellung von Personas, Interaction Cases, Rapid Prototypes und Screen Flows für mobile Business Apps.

### Keywords:

/// Mobile Business Apps  
/// User Experience  
/// Interaktionsdesign  
/// Requirements Engineering

## 1. Einleitung

Ohne Zweifel haben mobile Apps den Konsumentenmarkt erobert und Einzug ins tägliche Leben der Menschen erhalten. Aber auch im geschäftlichen Umfeld wird das Potential von mobilen Geschäftsapplikationen (sog. Mobile Business Apps) erkannt. Mehr und mehr Unternehmen versuchen ihr eigenes mobiles Potential zu entdecken und dieses für sich zu nutzen. Ziele der Unternehmen sind dabei einerseits die Steigerung von Arbeitseffizienz und Arbeitskomfort der Angestellten (interne Nutzung von Apps), aber auch die Erweiterung des bestehenden Produktportfolios durch Apps (Produktion von Apps). Mobile Business Apps bieten darüber hinaus ein hohes Potential, bestehende Geschäftsprozesse nicht nur zu unterstützen, sondern vielmehr diese zu erweitern bzw. signifikant zu vereinfachen.

Bei der Entwicklung von Mobile Business Apps sehen sich Unternehmen mit neuen Herausforderungen konfrontiert, denn die Konzeption und Entwicklung von mobilen Applikationen unterscheidet sich von der herkömmlicher Informationssysteme. So müssen bei der Konzeption von Mobile Business Apps nachstehende mobile spezifische Herausforderungen von Usability Professionals berücksichtigt werden:

– **Hohe Usability und User Experience:** Usability und User Experience sind sehr wichtige Qualitätsattribute für Mobile Business Apps. Für die Akzeptanz einer Applikation erwarten Nutzer diese direkt produktiv einsetzen zu können. Typischerweise lesen sie keine Anleitungen oder benutzen Hilfesysteme. Viel mehr werden Anwendungen, die die Erwartung der Anwender nicht erfüllen abgelehnt und nicht weiter eingesetzt. Es wird eine einzigartige User Experience erwartet, die insbesondere auf die zugrundeliegende Plattform

zugeschnitten ist. Zur Erhöhung der User Experience sollten unter anderem besondere Features der Geräte (z. B. Sensordaten, Multitouch, Kamera, Mikrophon) verwendet werden. Insbesondere die bei Mobile Business Apps in der Regel benötigte Kommunikation mit einem Backend ist für Usability und User Experience relevant.

– **Klarer und limitierter Funktionsumfang:** Ein weiterer kritischer Erfolgsfaktor für Mobile Business Apps, den wir in der Praxis beobachten konnten, ist ein klarer und limitierter Funktionsumfang. Optimale Produktivität kann in der Regel erreicht werden, wenn die App auf eine bestimmte Geschäftsaktivität für eine bestimmte Rolle zugeschnitten ist und keine allesumfassende Lösung darstellt.

– **Erweiterung existierender Geschäftsprozesse:** Zunächst müssen Organisationseinheiten und existierende Geschäftsprozesse

identifiziert werden, die sinnvoll durch Mobile Business Apps unterstützt werden [Yuan, 2005] können. Diese Identifikation muss dann direkt mit der Konzeption verbunden sein. Dabei sollten kreative Lösungen gefunden werden, wie der Geschäftsprozess nicht nur unterstützt sondern ggf. auch vereinfacht werden kann.

– **Nutzungskontext und Nutzungsumgebung:**

Herausforderungen bezüglich des Nutzungskontexts sind schon bei der traditionellen Anwendungs-konzeption sehr vielfältig. Neben Faktoren, wie Temperatur, Lichtverhältnisse, Geräusche, Ablenkung, Mobilität des Nutzers, kognitive Belastung und physische Einschränkungen [Looije, 2007] muss für Mobile Business Apps der konkrete Nutzungskontext erhoben werden. Da Mobile Business Apps auf einem konkreten Unternehmenstask basieren, ist der Nutzungskontext besser bestimmbar als vergleichsweise bei Konsumenten-Apps.

– **Durchführung von frühen Usability Tests im Nutzungskontext:**

Bezüglich Usability Testing besteht eine der Hauptherausforderungen in der Durchführung von frühen Nutzertests im aktuellen Nutzungskontext auf dem Endgerät. Mitunter sind Nutzer von Mobil Business Apps nur schwer verfügbar (z. B. Piloten), da diese in geschäftskritische Aktivitäten eingebunden sind und eine Freistellung oft mit hohen Kosten für das Unternehmen verbunden ist. Außerdem involviert der Nutzungskontext oft Kunden des Endnutzers (z. B. Passagiere), die aus verschiedenen Gründen nicht in die Konzeption der App involviert werden können.

– **Konsistentes Look & Feel:** Aus dem klaren und limitierten Funktionsumfang einer Mobile Business App ergibt sich für Unternehmen die Konsequenz ein sog. App-Portfolio mit mehreren maßgeschneiderten Anwendungen aufzubauen. Für dieses Gesamtportfolio an Mobile Business Apps, ist es wichtig, ein konsistentes

Look & Feel zu gewährleisten. Durch ein konsistentes Look & Feel können erlernte Konzepte vom Anwender über das gesamte App-Portfolio angewendet werden. Eine weitere Herausforderung für das Look & Feel von Mobile Business Apps besteht darin, dass bereits existierende sonstige Software im Unternehmen beachtet werden muss, um somit ein konsistentes Look & Feel über die gesamte Unternehmenssoftware zu erhalten.

– **Limitierte Nutzeraufmerksamkeit:**

Da die Nutzung von Mobil Business Apps oft in einem ständig wechselnden, mobilen Kontext stattfindet haben Nutzer nur eine limitierte Aufmerksamkeit für die Interaktion. Ein Großteil der Aufmerksamkeit muss von den Anwendern auf den wechselnden Kontext der Benutzung oder weiteren ‚Tasks‘ (bspw. gehen) aufgewendet werden. Der Umstand, dass die App in einem primären oder sekundären Task benutzt wird muss bei der Konzeption der Mobile Business Apps beachtet werden.

Neben diesen spezifischen Herausforderungen gibt es aber auch noch generelle Herausforderungen, die einen Einfluss auf die Konzeption von Mobil Business Apps haben:

– **Kurze Markteinführungszeit und limitiertes Budget:** Entwicklungen von Mobile Business Apps müssen schnell sein und haben in der Regel ein sehr limitiertes Budget. Dennoch müssen sie ein hohes Maß an Softwarequalität bereitstellen. Daher ist eine leichtgewichtige Vorgehensweise in der Praxis notwendig, die es ermöglicht schnell hochwertige Apps zu erstellen.

– **Integration in eine existierende IT-Infrastruktur:** Grundlage für Mobile Business Apps stellt in der Regel die Kommunikation mit einem Backend System dar. Häufig werden benötigte Informationen aus einer bereits im Unternehmen existierenden Infrastruktur an die Mobile Business

App übertragen. Die Integration von Mobile Business Apps hat oft Modifikationen im Backend zur Folge. Hieraus ergibt sich ein besonderer Abstimmungsbedarf zwischen den Bereichen des Interaktionsdesigns der Applikation und der Architektur des Gesamtsystems. Bereits in frühen Konzeptionsphase müssen sich beide Bereiche intensiv miteinander auseinandersetzen, da sich hier entscheidet, welche Funktionalität in der eigentlichen App und welche auf dem Backend realisiert wird und welche Auswirkungen diese Entscheidungen auf die User Experience haben.

– **Unterstützung verschiedener Geräte und Geräteplattformen:**

Schon bei der Konzeption von Mobile Business Apps muss berücksichtigt werden, dass die spätere Business App auf verschiedenen Geräten und Geräteplattformen eingesetzt werden kann. Dies beinhaltet aktuell vor allem die Unterscheidung zwischen Android und iOS-Systemen und zwischen Smartphone und Tablet.

Um den Herausforderungen an Mobile Business Apps gerecht zu werden und diese in ihrer Gesamtheit adressierbar zu machen, müssen sie mit einem methodischen Engineering Ansatz erstellt werden. Dieser muss auf der einen Seite sehr nutzerzentriert sein, um die Ziele der Benutzer zu erfüllen, auf der anderen Seite leichtgewichtig und integriert sein, um die Ziele der Unternehmen und der übrigen im Softwareentwicklungsprozess beteiligten Rollen zu erfüllen.

Dieser Artikel stellt die mConcAppt Methode vor – eine Methode zur Konzeption von Mobil Business Apps. Konzeption umfasst in diesem Fall sowohl Requirements Engineering Aktivitäten, als auch Interaktionsdesign und dessen Evaluation. Diese integrierte Sichtweise stammt aus dem TORE Ansatz zum task-orientierten Requirements Engineering [Adam, 2009]. Die mConcAppt Methode verwendet weitestgehend traditionelle Konzepte des Requirements und Usability Engineering.

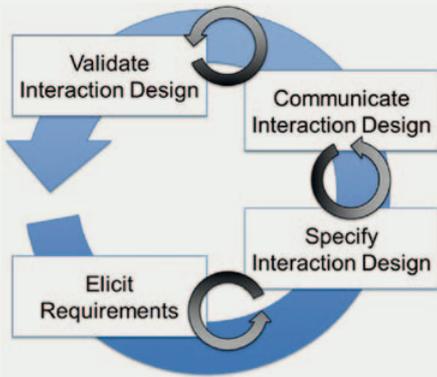


Abb. 1.  
Überblick der mConcAppt Methode

Diese sind jedoch auf die besonderen Herausforderungen und Bedürfnisse von Mobile Business Apps angepasst. Die Methode ist nutzerzentriert, um hohe Usability und User Experience, sowie einen klaren und limitierten Funktionsumfang zu gewährleisten. Sie ist leichtgewichtig, um die Anforderungen von Organisationen an effiziente App Entwicklung und kurze Markteinführungszeiten zu realisieren. Außerdem bietet mConcAppt Schnittstellen zu sämtliche Aktivitäten des Software Engineering Prozesses und ist somit in diesen vollständig integriert.

Das folgende Kapitel gibt einen Überblick über die mConcAppt Methode und die darin verwendeten Templates und Ansätze.

## 2. mConcAppt Methode

Im nachstehenden Kapitel wird zunächst ein genereller Überblick über die mConcAppt Methode aufgezeigt. Darauf folgend werden die einzelnen Phasen der Methode im Detail beschrieben.

### 2.1. Überblick

Die mConcAppt Methode besteht aus vier Phasen (vgl. Abbildung 1): **Elicit Requirements, Specify Interaction Design,**

**Communicate Interaction Design und Validate Interaction Design.** Bevor ein App Konzept erstellt wird, gibt es vorgelagerte Aktivitäten. Als Ausgangspunkt wird hierbei der Bedarf nach mobiler Unterstützung oder eine App-Idee innerhalb eines Unternehmens angenommen. Basierend darauf muss zunächst ein initiales Projektbudget allokiert und ein Projektteam zusammengestellt werden.

Danach beginnt die eigentliche Durchführung der mConcAppt Methode – entscheidend dabei ist die enge Kommunikation mit den umliegenden Entwicklungsaktivitäten. Vor allem die Schnittstelle zu Business Analyse und zu Architektur sind wichtige Quellen für Anforderungen an das Interaktionsdesign. Aber auch frühe Kommunikation zu Entwicklung, Testen und Visual Design sind notwendig, um z. B. Testfälle zu planen, technische Konzepte zu erproben und visuelle Designs vorzubereiten.

Nachgelagerte Aktivitäten der Methode umfassen vor allem die eigentliche Entwicklung von App und Backend, sowie die in-house-Distribution bzw. die Distribution in den App-Stores. [Abb. 1]

Zunächst werden in der **Elicit Requirements Phase** die Requirements für das Interaktionskonzept erhoben. Darauf aufbauend wird das eigentliche Interaktionskonzept spezifiziert und anschließend zu den übrigen Stakeholdern kommuniziert. Es ist sehr wichtig, dass bereits frühe Versionen des Interaktionskonzeptes zu sämtlichen Mitgliedern des Projektteams und übrigen Stakeholdern kommuniziert werden, da deren Feedback bzgl. Machbarkeit und die Auswirkungen auf deren Bereiche Einfluss auf das Interaktionsdesign haben. Eine genaue Struktur der Informationsschnittstellen zwischen der mConcAppt Methode und den anderen SE-Rollen ist ausführlicher in [Hess, 2012] beschrieben. Eine Validation des Konzeptes in Form von Endnutzertests erfolgt nur zu bestimmten Zeitpunkten in Absprache mit dem Projektmanagement. Gerade bei Business Apps ist es notwendig, dass die Produktverantwortlichen entscheiden können, an welchem Punkt im Entwicklungsprozess Nutzer involviert werden

können, da frühe Entwicklungsstadien von Produkten oft kritisch sind bzgl. der öffentlichen Bekanntmachung. Wichtig für die Durchführung der Methode ist, dass sie keinen sukzessiver Ablauf, sondern ein iteratives Vorgehen beschreibt. Input und Output aus den verschiedenen Phasen gehen jeweils in die übrigen Phasen ein. Die folgenden Unterkapitel zeigen, wie die Phasen im Detail aussehen, unter der Annahme, dass diese einmal komplett durchlaufen werden. Für jede Aktivität der Phasen existieren Templates, die die Anwendung der Methode operationalisieren.

### 2.3. Elicit Requirements

Die **Elicit Requirements Phase** fasst die initiale Erhebung von Anforderungen für das Interaktionsdesign zusammen. Sämtliche Aktivitäten dieser Phase und deren geschätzte Dauer sind in Abbildung 2 dargestellt. Zentraler Baustein hierbei ist ein Workshop, der sowohl Endnutzer als auch übrige relevante Stakeholder inkludiert. [Abb. 2]

**Workshop Preparation** beinhaltet hierbei alle Planungstätigkeiten, die notwendig sind um den Workshop durchzuführen. Hierzu zählt vor allem die Auswahl der Teilnehmer des Workshops. Wir haben beste Erfahrungen damit gemacht, die Gruppe möglichst klein zu halten und die Rollen der Teilnehmer klar zu spezifizieren (vgl. Tabelle 2). [Tab. 1]

Die eigentliche Durchführung und Dokumentation des Workshops ist in Abbildung 3 dargestellt. Zunächst werden die Stakeholder und Nutzerrollen der künftigen App erhoben. Darauf aufbauend wird für die Hauptnutzergruppe gemeinsam eine User Persona [Cooper, 1999] erstellt. Die gewählte Persona sollte hierbei eine Instanziierung der Nutzergruppe sein, die dem 80%-Nutzer am nächsten kommt. Sie beschreibt die verschiedenen Ziele und Verhaltensweisen des Nutzers, sowie persönliche Eigenschaften und Charakteristika. Zusätzlich kapselt und erklärt sie kritische Verhaltensweisen und stellt diese personifiziert dar. Dies ermöglicht es



Abb. 2. Elicit Requirements Phase

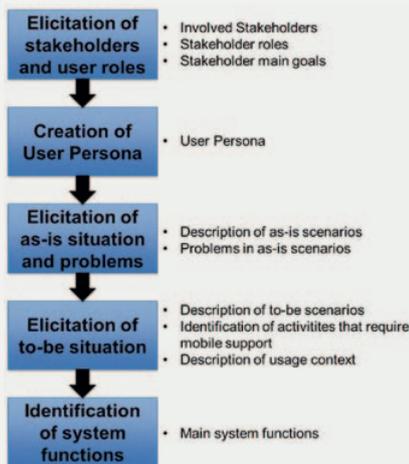


Abb. 3. Workshop Durchführung

anderen Projekt-Stakeholder diese Eigenschaften zu verstehen, sich an sie erinnern und sich bei Designentscheidungen darauf beziehen zu können. Die Persona wird auch genutzt, um die aktuelle Situation des Nutzers in Form von Szenarien zu beschreiben. Die Szenarien beinhalten die vorab identifizierte App-Idee oder den identifizierten Bedarf nach mobiler Unterstützung. Ein wichtiger Fokus bei der Beschreibung liegt hierbei auf der Identifikation von Problemen in der aktuellen Situation (as-is Situation), die durch den Einsatz einer App gelöst oder gemindert werden können. [Abb. 3]

Die Beschreibung der künftigen Situation (to-be Situation) enthält die zuvor definierten Szenarien der as-is Situation inklusive der Unterstützung durch die künftige App. Hierbei wird insbesondere darauf eingegangen, wie existierende Geschäftsprozesse durch die App verbessert bzw. erweitert werden können. Sowohl die as-is Situation als auch die to-be Situation werden mit einem standardisierten Template beschrieben (siehe Tabelle 3). Es enthält in einer leichtgewichtigen Form die wichtigsten Schritte einer Szenariobeschreibung. Abschließend werden im Workshop bereits gemeinsam die wichtigsten Systemfunktionen (Main System Functions) skizziert und priorisiert. Diese werden aus den beschriebenen to-be Szenarien abgeleitet und bilden den Kern der App. Durch die frühe Skizzierung der Kernfunktionalitäten wird bereits ein wichtiger Grundstein für einen kleinen Umfang von Features gelegt. Diese Features fokussieren dabei genau auf die zuvor definierten Lösungen und verhindern einen zu breitgefächerten Funktionsumfang einer einzelnen Applikation. Während des Workshops können die erhobenen

Artefakte oft nur skizzenhaft dokumentiert werden. Um auch die Dokumentationsarbeiten, die im Nachgang durchgeführt werden müssen, leichtgewichtig zu halten, bietet die mConcAppt Methode auch für diesen Schritt geeignete Templates und Vorlagen. [Tab. 2]

Praktische Erfahrungen zeigen, dass in der Tat nur ein Workshop benötigt wird, um die initialen Anforderungen zu erheben. Dennoch sollten im Nachgang kurze Kommunikationswege (bspw. E-Mail oder Telefon) eingesetzt werden um durch Gespräche mit vorher bestimmten Ansprechpartnern etwaige Unklarheiten beseitigen zu können. Dennoch ist es wichtig, so viele Informationen wie möglich bereits im Voraus zu erheben. Um eine möglichst vollständigen Informationsfluss zu erhalten ist es notwendig, dass alle zuvor definierten Rollen am Workshop teilnehmen. Unsere Erfahrung zeigt auch, dass die Qualität der Ergebnisse des Workshops erheblich durch die Qualitäten und Erfahrungen des Moderators bestimmt werden.

Teilnehmer	Rolle im Workshop
Interaktionsdesigner/ Requirements Experte (1-2)	Konzeption des Workshops und Vorbereitung. Moderator des Workshops
(Lead) User (2-4)	Stellen die benötigten Informationen aus Nutzersicht zur Verfügung. Der Lead User Ansatz sollte gewählt werden, falls keine tatsächlichen Benutzer verfügbar sind.
Projektmanagement (1)	Stellt Informationen aus Sicht des Projektmanagements zur Verfügung. Entscheidungsträger des Projektes.
Business Analyst (1)	Stellt domänenrelevante Informationen und Informationen bzgl. Machbarkeit zur Verfügung.
Kunde (1)	Stellt Informationen und Limitierungen aus Sicht des Kunden zur Verfügung. Diese Rolle repräsentiert den Entscheidungsträger beim eigentlichen Kunden.

Tab. 1. Workshop Teilnehmer und deren Rolle im Workshop

Item	Beschreibung
Nutzungskontext	Kontext der Nutzung, der zur eigentlichen Task-Ausführung führt.
Vorbedingung	Vorbedingung, die erfüllt sein muss.
Schritt1-N	Schritte, die zur Erfüllung des Szenarios durchgeführt werden.
Nachbedingung	Status, der nach Durchführung des Szenarios erreicht ist.

Tab. 2. Template zur Beschreibung von as-is und to-be Situation

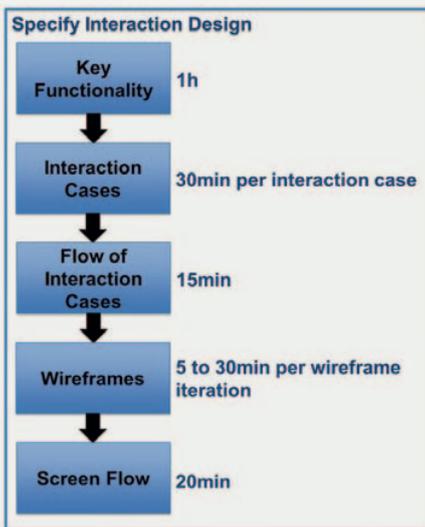


Abb. 4. Interaction Design Specification Phase

### 2.3. Specify Interaction Design

Input für die eigentliche Spezifikation des Interaktionskonzeptes (siehe Abbildung 4) sind sämtliche Informationen, die während des Workshops erhoben und nachfolgend dokumentiert wurden. Für die Spezifikation des Interaktionskonzeptes wird zunächst die Key Funktionalität bestimmt. Sie wird mit Hilfe der to-be Szenarien und den im Workshop bereits definierten Systemfunktionen abgeleitet. Im Gegensatz zu den Systemfunktionen, die mit einem höheren Abstraktionsgrad beschrieben werden, legt die **Key Funktionalität** konkrete Schritte zur Erfüllung der angestrebten Systemfunktion fest. Es wird explizit festgehalten, welche Schritte im Szenario tatsächliche mobile Unterstützung benötigen. Durch die Beschreibung der **Key Funktionalität** wird auch der letztendliche Umfang der Business App bestimmt. [Abb. 4]

Zur Beschreibung der **Key Funktionalität** werden Interaction Cases spezifiziert. Interaction Cases sind leichtgewichtige Use Cases, die einen sehr starken Fokus auf die Benutzerinteraktion haben. Für die Dokumentation der Interaction Cases stellt

die mConcAppt Methode ein Template (siehe Tabelle 3) zur Verfügung. Diese Template beinhaltet wenige Items um dem leichtgewichtigen Anspruch der Methode gerecht zu werden. Allerdings wird bereits hier explizit zwischen Benutzeraktion (Human Action) und Systemaktion (System Action) unterschieden um einen möglichst hohen Grad von Nutzerzentriertheit wiederzugeben. Die Human Action beschreibt die konkrete Form der Interaktion (z. B. die konkrete Geste, mit der der Benutzer die Eingabe tätigt) – die System Action beschreibt die explizite Systemreaktion auf die Eingabe des Benutzers. Bei der Beschreibung der System Action gibt das Template vor, explizit zwischen Eingabefeedback und eigentlicher Systemausgabe zu unterscheiden. [Tab. 3]

In der Regel ist der Interaction Case so geschnitten, dass jeweils eine Human Action und eine System Action als User Story abgebildet werden können.

Nachdem sämtliche **Interaction Cases** spezifiziert wurden, werden diese in einem **Interaction Case Flow** arrangiert. Der Interaction Case Flow zeigt Abhängigkeiten und Navigationsmöglichkeiten zwischen den verschiedenen Interaktionen auf. Mit Hilfe des **Interaction Case Flow** wird der momentane Skopus der Applikation in einer übersichtlichen Form dargestellt. Diese Darstellung dient als Grundlage der

finale Entscheidung über den Umfang der App. Beinhaltet sie zu viele Bereiche oder zu viele Interaction Cases, ist der Skopus womöglich zu groß gewählt und es sollte untersucht werden, ob sich dieser sinnvoll in mehrere kleinere Apps unterteilen lässt.

Anschließend werden für die **Interaction Cases Wireframes** erstellt. Hierbei wird in der Regel jedes Paar aus Human Action und System Action durch ein **Wireframe** dargestellt (vgl. Abbildung 5). Für das Interaktionskonzept ist es ausreichend, diese mit Papier und Stift zu erstellen – oder ein leichtgewichtiges Tool wie z. B. Balsamiq (<http://www.balsamiq.com>) zu verwenden. Da die **Wireframes** untereinander semantisch verknüpft sind, wird parallel zu ihnen auch der Screen Flow (siehe Abbildung 6) erstellt. In ihm wird dargestellt, wie der Benutzer durch die **Wireframes** navigieren kann. Neben den Navigationswegen wird auch angegeben, welche Screenübergänge (Transitionen) zwischen den Screens eingesetzt werden. [Abb. 5], [Abb. 6]

Praktische Erfahrungen zeigen, dass es von Vorteil ist, wenn der Interaktionsdesigner selbst extensiver Nutzer der Zielplattform ist. Somit ist garantiert, dass plattformspezifische Interaktionsformen bekannt sind und auch im Interaktionskonzept der Applikation berücksichtigt werden. Neben direkten Entscheidungen

Item	Beschreibung
ID	Eindeutige ID des Interaction Case.
Benutzungskontext	Plausible Beschreibung der aktuellen Umgebung und Situation basierend auf Annahmen, wie der Interaction Case ausgeführt wird.
Human Action 1	Eine Human Action wird vom Benutzer auf dem Eingabegerät ausgeführt und wird durch einen konkreten Usage Type (z.B. double tap) beschrieben.
System Action 1	Beschreibung der zugehörigen Systemausgabe. Hierbei ist explizit zwischen Eingabefeedback und eigentlicher Ausgabe zu unterscheiden.
...	...
Nachbedingung	Die Nachbedingung zeigt explizit, wo sich der Benutzer aktuell im System befindet.

Tab. 3. Beschreibungstemplate für Interaction Cases



Abb. 5. Abbildung verschiedener Evolutionsstadien von Wireframes

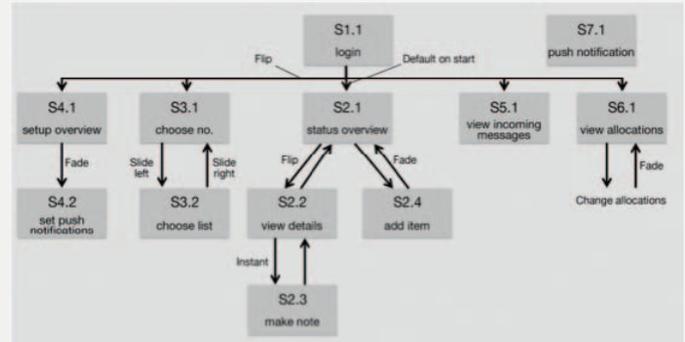


Abb. 6. Abbildung eines Screen Flows einer Beispielanwendung

des Interaktionsdesigners haben auch Entscheidungen die gemeinsam mit anderen SE-Rollen getroffen werden starke Bedeutung für das App-Verhalten. So haben gerade die Entscheidungen, die bestimmen, welche Funktionalität auf dem Backend und welche auf dem Frontend (der eigentlichen App) realisiert wird großen Einfluss auf die Usability und User Experience und somit auch auf den Erfolg einer Applikation haben. Werden bspw. zu rechenintensive Funktionen in der Applikation ausgeführt oder stehen benötigte Informationen nicht zum gewünschten Zeitpunkt in der Applikation zur Verfügung leidet die User Experience und somit auch die Akzeptanz der Applikation beim Endnutzer. Um falsche Entscheidungen in diesem, aber auch in anderen Bereichen der App-Entwicklung vorzubeugen ist eine frühe Kommunikation von Ideen und Sketches zu den entsprechenden Stakeholdern notwendig. Durch eine permanente Kommunikation wird signifikant die Qualität des Resultates verbessert. Unsere Erfahrung hat gezeigt, dass gerade das Interaktionsdesign und die Software-Architektur sich erheblich beeinflussen. Die angesprochene Problematik der Backend/Frontend-Beziehung wird maßgeblich von der Architektur beeinflusst. Das App-Frontend hängt dabei sehr von der Performanz im Backend ab.

## 2.4. Communicate Interaction Design

In der **Communicate Interaction Design** Phase wird das Interaktionsdesign zu den übrigen Projekt Stakeholdern kommuniziert (Vergleiche Abb.9). Unsere Erfahrungen zeigen, dass die Erstellung eines Videos, in dem das Interaktionskonzept beschrieben wird eine gute Lösung ist, um das Verständnis der Konzepte und Ideen des Interaktionsdesigners zu kommunizieren. Dies ist insbesondere der Fall, wenn es sich um ein lokal verteiltes Projektteam handelt. Sollte das Projektteam nicht lokal verteilt sein, ist die Erstellung des Videos als optionaler Schritt zu sehen. Auch wenn unsere Erfahrung gezeigt hat, dass es gerade bei nicht technischen Stakeholdern ein sehr gutes Kommunikationsmedium darstellt. Der Fokus des Videos sollte darauf liegen, die Informationen zwischen allen beteiligten Stakeholdern auszutauschen und Feedback einzuholen. Dieses kann dann bei der Gestaltung des Interaktionskonzeptes Berücksichtigung werden.

Um das Video zu erstellen, sollte insbesondere der unerfahrene Interaktionsdesigner zunächst ein Drehbuch (**Presentation Script**) schreiben. Basierend auf diesem Drehbuch werden Clips aufgenommen, die sich an den zuvor erstellten **Interaction**

**Cases** orientieren. Bei der Erstellung der Clips ist es wichtig jeden einzelnen Schritt im Video nicht nur zu zeigen sondern auch mit kurzen und prägnanten Worten zu erklären. Aus diesen Clips wird nachfolgend ein komplettes Video erstellt, das im gesamten Projektteam verteilt werden kann. Dieses Video ermöglicht es insbesondere fachfremden Personen, das Interaktionskonzept schnell und in einfacher Weise zu verstehen, ohne ein umfassende Dokumentation lesen zu müssen. [Abb. 7]

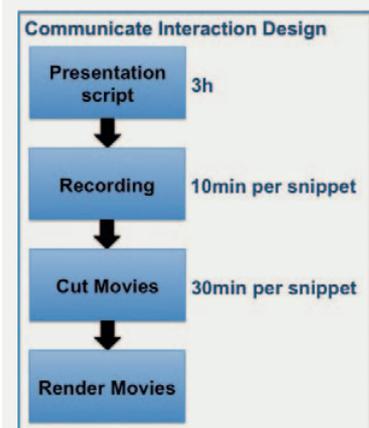


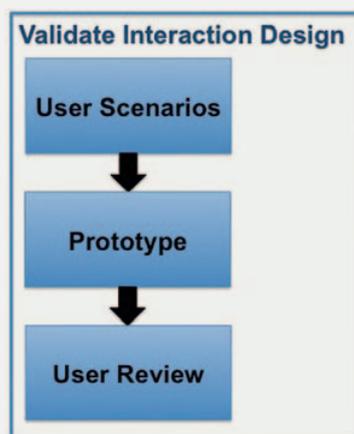
Abb. 7. Communicate Interaction Design Phase



Im **Presentation Script** wird die Interaktion mit der Mobile Business App anhand der erstellten Wireframes in Form von Szenarien erfasst. Dabei ist es wichtig, vor allem Hintergrundinformationen bzgl. Designentscheidungen, die man nicht auf den ersten Blick sehen kann, zu geben (z. B. Screen Transitionen, Gesten, Eingabefeedback und besonders innovatives oder nicht standardisiertes Verhalten der Applikation). Erfahrene Interaktionsdesigner können auf die Erstellung des **Presentation Script** verzichten und direkt das Video drehen. Die Erstellung des **Presentation Script** ist dennoch ratsam, da es eine wichtige Dokumentationsquelle auch für die weiterführenden Arbeiten darstellt.

## 2.5. Validate Interaction Design

Wie in Abbildung 1 zu sehen ist, wird die Validierungsphase erstmals am Ende einer kompletten Iteration der Methode durchgeführt. Abbildung 8 zeigt einen Überblick der durchzuführenden Aktivitäten. Basierend auf den **Interaction Cases** werden konkrete Nutzungsszenarien [Carrol, 2000] erstellt, die die Basis für die Durchführung der Nutzertests darstellen. Zusätzlich wird ein interaktiver Prototyp erzeugt. Hierzu können einfach Tools wie



**Abb. 8.**  
Validate Interaction Design Phase

Präsentationssoftware für Mobilgeräte verwendet werden. Wichtig ist dabei, dass die Präsentationen nicht-linear ablaufen müssen (z. B. Presentation Link – <http://www.presentation-link.com>). Dieser Ansatz erfordert nur geringfügig mehr Aufwand als beispielsweise das Wizard-Of-Oz Testing [Kelley, 1983] mit echten Papierprototypen. Zudem ermöglicht es der Einsatz von mobilen Präsentationstools, den Prototyp in der tatsächlichen Nutzungsumgebung auf dem Endgerät zu testen. Da sowohl der Nutzungskontext wie auch das Endgerät großen Einfluss auf die Mobile Business Apps haben, ist ein Test unter möglichst realen Bedingungen unbedingt erforderlich. [Abb. 8]

## 3. Zusammenfassung und Ausblick

Die mConcAppt Methode unterstützt ihren Anwender bei der systematischen Durchführung von Requirements Engineering und Interaktionsdesign für Mobile Business Apps. Sie hilft dabei vor allem, die für Mobile Business Apps genannten spezifischen Herausforderungen zu adressieren. So wird beispielsweise eine hohe Usability und User Experience durch den nutzerzentrierten Ansatz und die Verwendung existierender anerkannter Konzepte garantiert. Ein klarer und limitierter Funktionsumfang wird explizit während der Spezifikation des Interaktionsdesigns berücksichtigt. Erweiterung existierender Geschäftsprozesse, explizite Beachtung des Nutzungskontextes sowie die Sicherstellung eines konsistenten Look & Feels sind durch die systematische Erhebung von as-is und to-be Situation in Kombination mit den künftigen Endbenutzern ein zentraler Teil der Methode. Frühes Usability Testing im Nutzungskontext ist durch die Durchführung von Nutzertest in jeder Iteration der Methode ermöglicht.

Die limitierte Aufmerksamkeit der Benutzer ist bisher nicht explizit adressiert und ist eine der künftigen Arbeiten zur Weiterentwicklung der Methode. Die generellen Herausforderungen, die im ersten Kapitel beschrieben wurden, werden implizit durch die

komplette mConcAppt Methode adressiert.

## Literatur

1. Adam, S.; Doerr, J.; Eisenbarth, M.; Groß, A. Using Task-oriented Requirements Engineering. In Proc. RE 2009, (2009) 267-272.
2. J.M. Carroll, "Making use: Scenario-based design of human-computer interactions" Cambridge, MA: The MIT Press, 2000.
3. A. Cooper, "The Inmates Are Running the Asylum : Why High-Tech Products Drive Us Crazy and How to Restore the Sanity" Indianapolis, USA, 1999.
4. S. Hess, and F. Kiefer, "Quality by Construction through mConcAppt – Towards Using UI-Construction as Driver for High Quality Mobile App Engineering" in Proc. of QUATIC 2012.
5. J.F. Kelley, "An empirical methodology for writing user-friendly natural language computer applications" Proc. of ACM SIG-CHI '83 Human Factors in Computing systems, Boston, 1983, New York: ACM, pp. 193-196.
6. R. Looije, and G. te Brake, "Neerincx, M.; Usability engineering for mobile maps" in Proc. of Mobility '07., ACM, New York, NY, USA, 2007, pp. 532-539.
7. Y. Yuan, and W. Zheng, "From Stationary Work Support to Mobile Work Support: A Theoretical Framework" Proc. of the International Conference on Mobile Business (ICMB ,05). IEEE Computer Society, Washington, DC, USA, 2005, pp. 315-321.

