

Usability-Workshop 2

Dipl.-Inf. Holger Röder

roeder@informatik.uni-stuttgart.de

Universität Stuttgart
Institut für Softwaretechnologie
Abteilung Software Engineering



Agenda



- Kurze Wiederholung: Usability Patterns
- Auswahl von Usability Patterns für das Sopra
- Spezifikation von Usability Features
 - Prinzip und Prozess
 - Erweiterung der Use-Case-Spezifikation
 - Spezifikationsschablonen
- Spezifikation mit Tulip
- Usability Patterns im Sopra – Teil 2

Usability Patterns

«*Mit welchen Features können wir eigentlich die Usability verbessern?*»

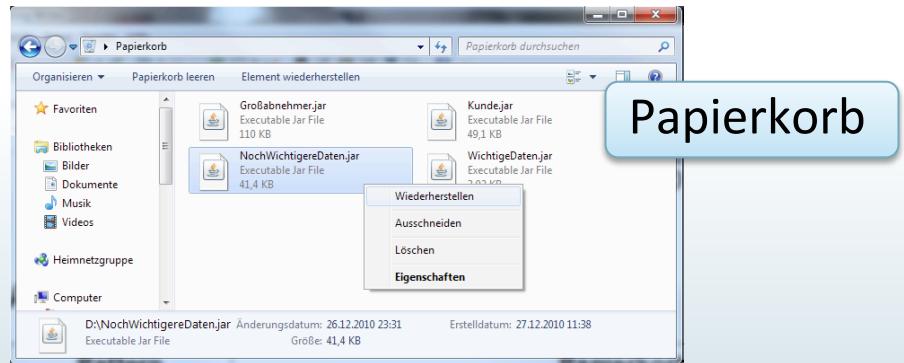
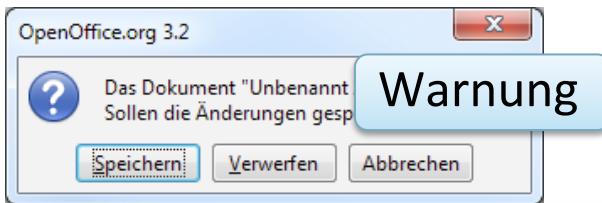
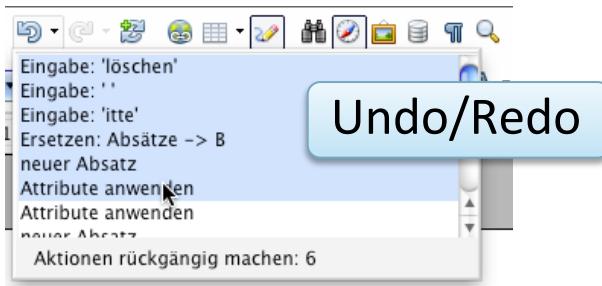


Usability Patterns

- Funktionale Lösungsmuster für typische Interaktionsprobleme in einem bestimmten Kontext
- Beschreibung wiederverwendbarer Usability Features in strukturierter, natürlichsprachlicher Form:
Problem – Lösung – Kontext – ...
- Usability Patterns dienen als **Gedächtnisstütze**,
Diskussionsgrundlage und **Anleitung** für die Projektbeteiligten!

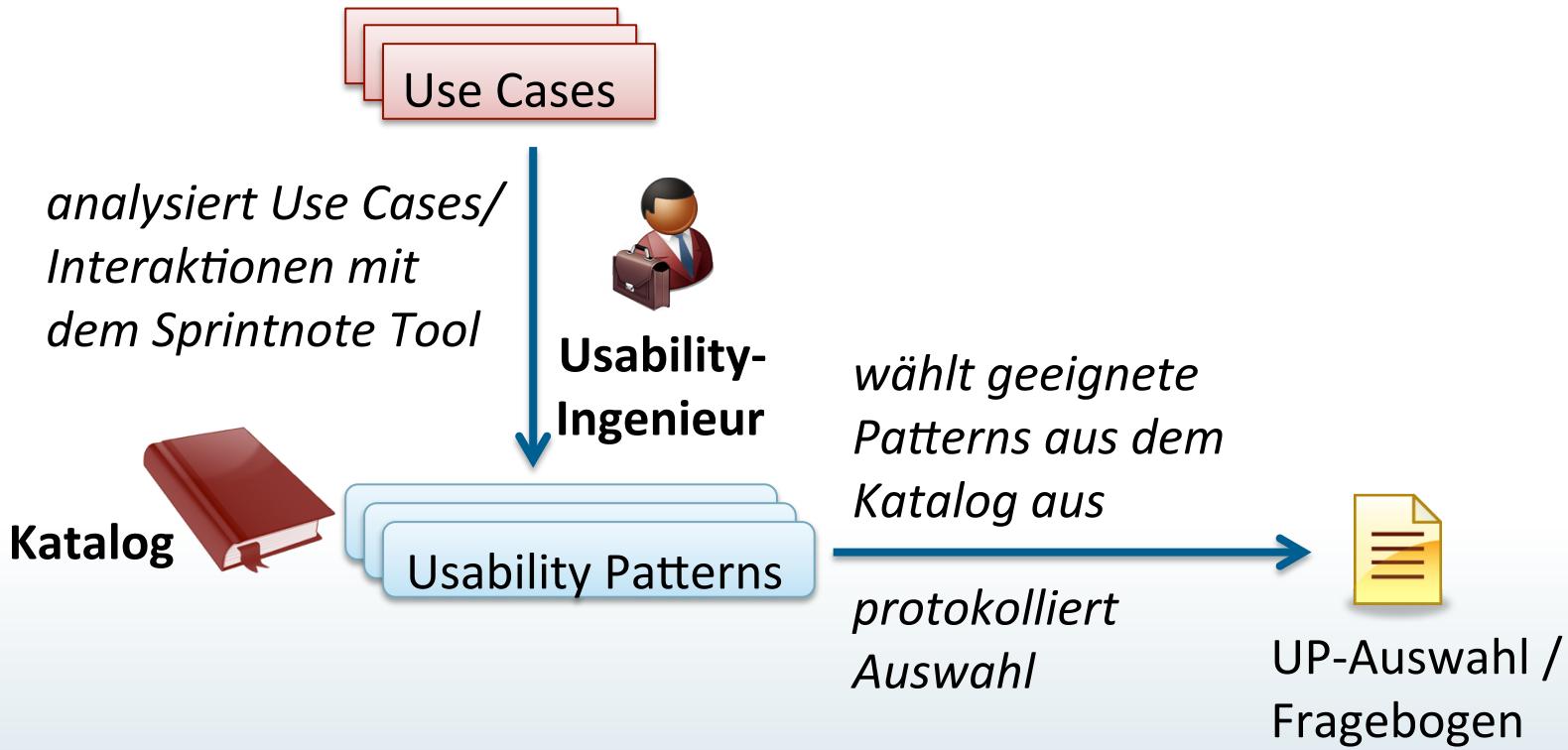
Katalog von Usability Patterns

20 Usability Patterns für interaktive Informationssysteme

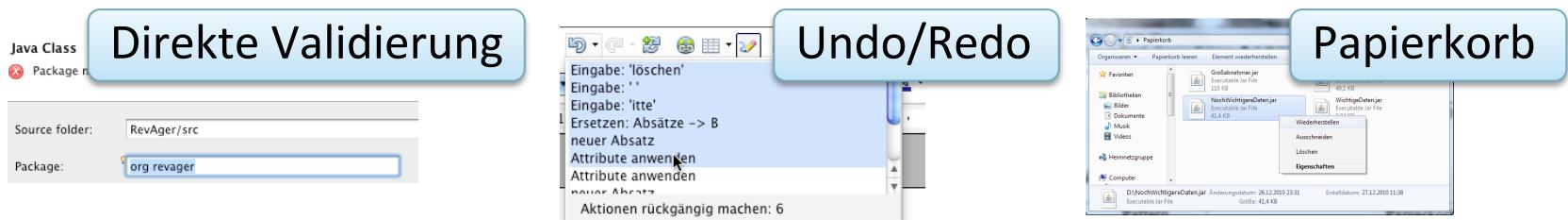


Usability Patterns im Sopra (1)

■ Schritt 1: Auswahl geeigneter Usability Patterns



Aufgabe: Usability Patterns für das Sprint Note Tool



- Überlegen Sie nochmals kurz, ob diese drei Usability Patterns in Ihrem neuen Spring Note Tool angewendet werden sollten.
- Welche Argumente sprechen für, welche gegen die Anwendung?



Zeit fürs Nachdenken: 3 Minuten

Anschließend Diskussion

Spezifikation von Usability Features

«Wie können wir die Anwendung von
Usability Patterns spezifizieren?»



Motivation

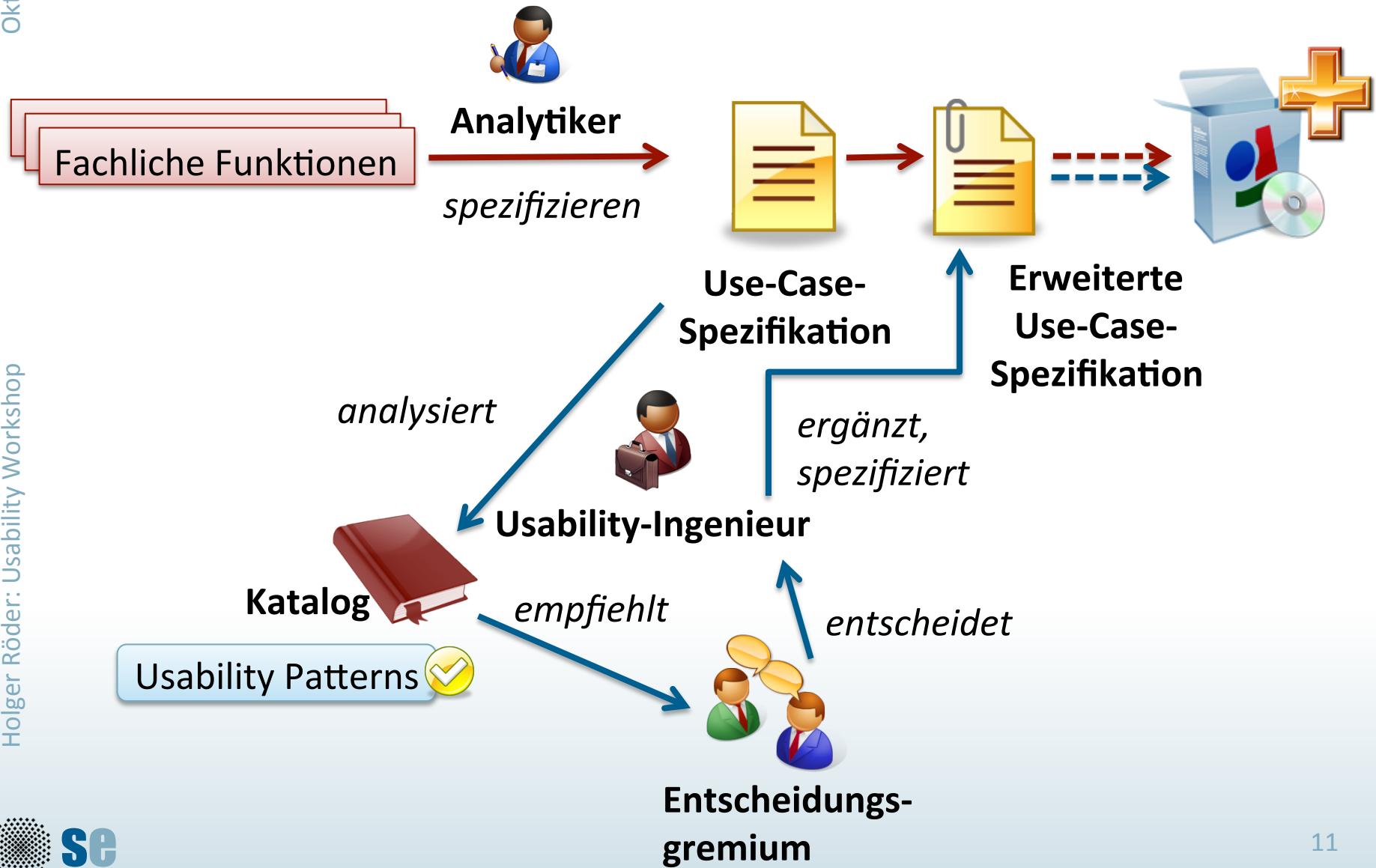
- Usability Patterns (Lösungsmuster) sind Ausgangspunkt für konkrete Usability Features des zu entwickelnden Systems.
- Die geforderten Usability Features sollten als «**vollwertige Anforderungen** spezifiziert werden. So kann die **Berücksichtigung bei Architekturentwurf, Implementierung und Test** sichergestellt werden!

Prinzip

- Usability Features beziehen sich stets auf fachliche Funktionen, deren Benutzung sie verbessern sollen.
- Die «**Verwebung**» von fachlicher Funktionalität und Usability Features sollte in der Spezifikation abgebildet werden.
- Idee: Usability Features werden spezifiziert, in dem eine vorhandene **Use-Case-Spezifikation** um spezielle Elemente **erweitert** wird.



Prozess



«Regeln» für die Spezifikation

Jedes Usability Feature wird **zweiteilig** spezifiziert:

Globale Vorgaben



Lokale Vorgaben

- Gelten für *alle* Vorkommen dieses Usability Features im System
- Werden an zentraler Stelle in der Use-Case-Spezifikation beschrieben

→ **Globale Parameter**

→ **Globale Funktionen**

- Gelten individuell für *ein* Vorkommen dieses Usability Features

- Werden in den einzelnen Use Cases spezifiziert

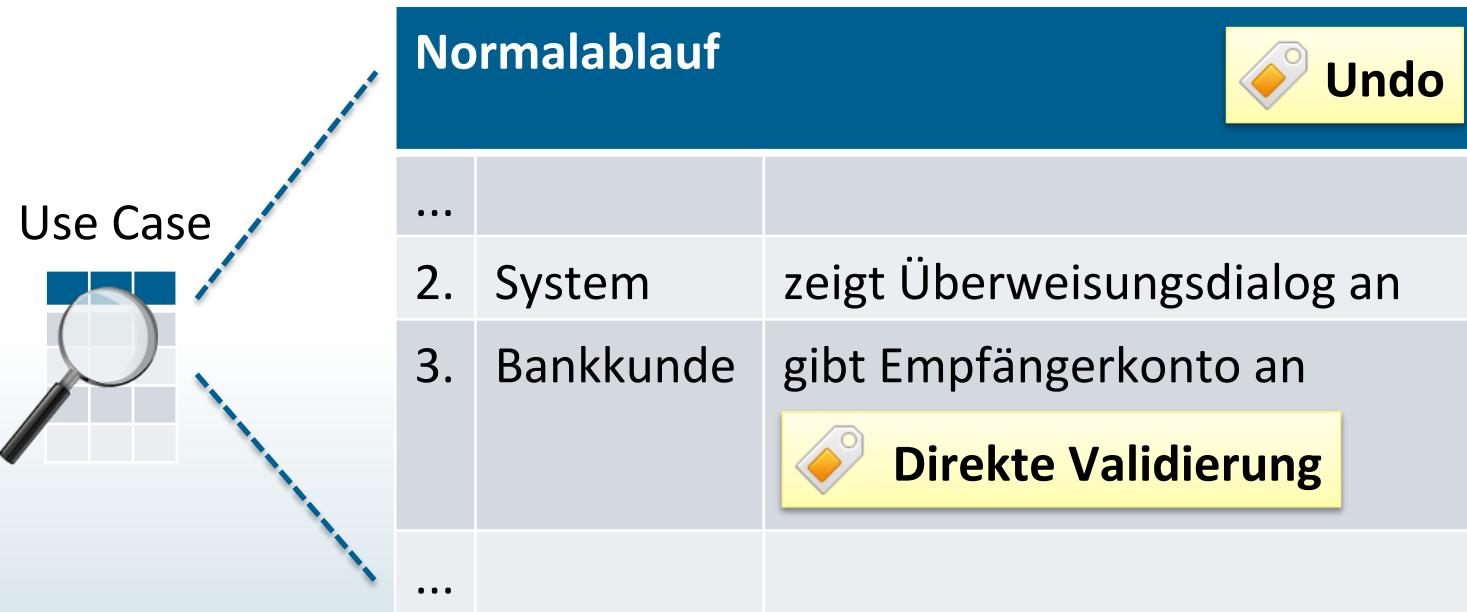
→ **Annotationen**

→ **Lokale Parameter**

→ **Use-Case-Erweiterungen**

Annotationen

- Annotationen kennzeichnen, in welchen Interaktionen ein Usability Feature eingesetzt bzw. angeboten werden soll.
- Je nach Feature können Annotationen an Abläufe oder Schritte im Use Case angehängt werden.



Lokale Parameter

- Lokale Parameter spezifizieren, *wie* ein bestimmtes Usability Feature in einer bestimmten Interaktion umgesetzt werden soll.
- Lokale Parameter **ergänzen** stets Annotationen.

2.	System	zeigt Überweisungsdialog an
	3. Bankkunde	<p>gibt Empfängerkonto an</p> <p> Direkte Validierung</p> <p> Gültige Werte: Kontonummer nach IBAN-Standard</p>

Use-Case-Erweiterungen

- Durch die Erweiterung eines Use Case um **zusätzliche Schritte oder Abläufe** wird der **veränderte Interaktionsablauf** spezifiziert, der sich aus dem Einsatz eines Usability Features ergibt.
- Die zusätzlichen Schritte/Abläufe werden von der Annotation **referenziert**.

Use-Case-Erweiterungen

Normalablauf		
...		
5.	Benutzer	schließt Dokument
		<div style="background-color: #ffffcc; padding: 10px;"> Warnung  Alternativen: Speichern & Schließen  Ablauf bei Alternative</div>
...		
Alternativer Ablauf 5a: Speichern & Schließen		
5a1.	System	speichert Dokument
...		

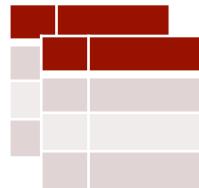
Referenz

Globale Parameter und Funktionen

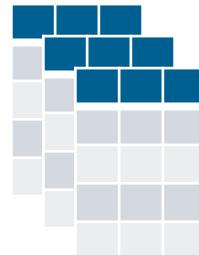
- **Globale Parameter** spezifizieren systemweite Vorgaben für Umsetzungsaspekte eines Usability Features.
- **Globale Funktionen** sind Referenzen auf Use Cases, die eigenständige Funktionen beschreiben, die sich aus Usability Features ergeben.
- Globale Parameter und Funktionen eines Usability Features werden in einem **separaten Kapitel** der Use-Case-Spezifikation beschrieben.

Erweiterte Use-Case-Spezifikation

Erweiterte Use-Case-Spezifikation



Usability Features:
Globale Parameter
Globale Funktionen



Use Cases
+ Annotationen ...

Globale Parameter und Funktionen

Usability
Feature



Usability Feature: Direkte Validierung

Darstellung

Fehler-Symbol direkt neben
Eingabefeld bei ungültigen
Eingaben ...

Validierungs-zeitpunkt

Unmittelbar während der
Eingabe ...

...

Usability Feature: Papierkorb

Papierkorb leeren

→ Use-Case UC-12

...

Spezifikationsschablonen

- Spezifikationsschablonen sind **Vorlagen**, die festlegen, mit welchen zusätzlichen Elementen ein Usability Feature in der Use-Case-Spezifikation spezifiziert wird.
- Jede Spezifikationsschablone legt **Art, Name und Inhalt** eines zu ergänzenden Elements (bzw. einer Menge von Elementen) fest.
- Eine Spezifikationsschablone kann als *benötigt* oder als *optional* gekennzeichnet sein. Entsprechend *muss* oder *kann* das beschriebene Element spezifiziert werden.
- Für jedes Usability Pattern sind **1..*** Spezifikationsschablonen als Teil der Beschreibung festgelegt.

Spezifikationsschablonen für «Ausführung im Hintergrund»

Spezifizierte **globale Vorgaben** für den Einsatz des Usability Patterns „Ausführung im Hintergrund“:

Vorgabe **Darstellung** Benötigt

Globale Parameterschablone



Vorgabe für die einheitliche Darstellung der im Hintergrund ausgeführten Aktionen.
Beispiel: „Anzeige der im Hintergrund ausgeführten Aktionen in der Statuszeile.“

Spezifizierte, für welche Interaktionen das Usability Pattern „Ausführung im Hintergrund“ eingesetzt wird. Annotiere und ergänze dazu vorhandene **Use Cases**:

Annotation @ Schritt **Hintergrundausführung**

Der Verarbeitungsschritt kann auf Wunsch des B werden.

Parameter **Standardmäßig im Hintergrund** Optional

Angabe, ob der Schritt standardmäßig automatisch im Hintergrund ausgeführt wird oder ob Benutzer die Hintergru

Annotationsschablone



Lokale Parameterschablone



«*Spezifizierte systemweite Vorgabe zur UI-Darstellung ...!»*

«*Annotiere alle Use-Case-Schritte, die im Hintergrund ausgeführt werden können ...!»*

«*Spezifizierte jeweils das Standardverhalten ...!»*

Aufgabe: Spezifikation «Direkte Validierung»

- Sie erhalten einen Auszug aus der (fiktiven) Use-Case-Spezifikation des Sprint Note Tools.
- Erweitern Sie die Use-Case-Spezifikation um das Usability Feature «Direkte Validierung»!

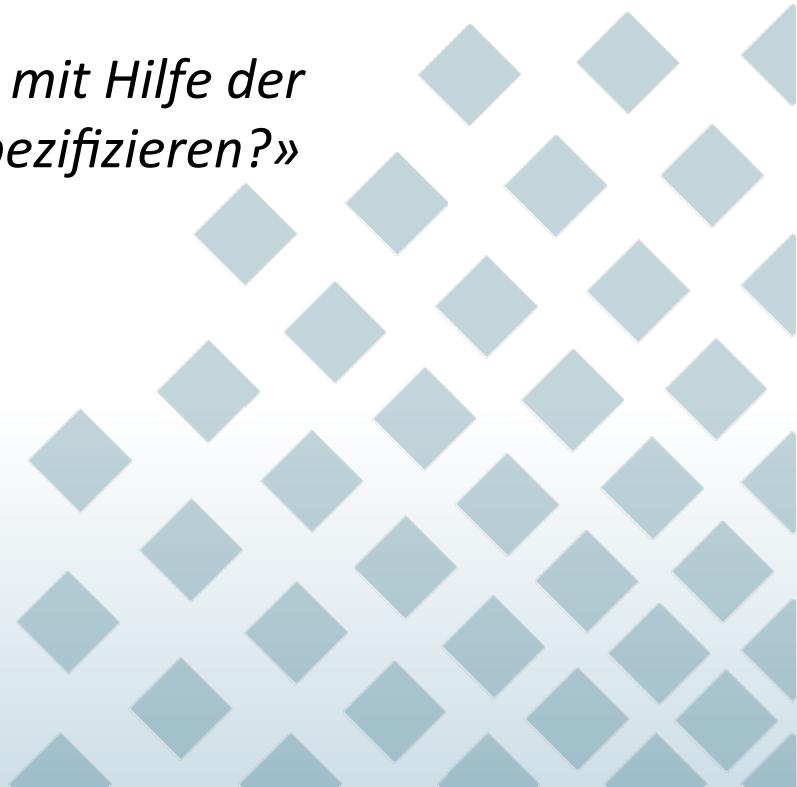


Arbeit in der Sopra-Gruppe
Zeit: 5 Minuten

Anschließend Vorstellung und Diskussion

Spezifikation mit Tulip

«Wie können wir Usability Features mit Hilfe der Spezifikationsschablonen einfach spezifizieren?»



Unterstützung von Usability Patterns/Features in Tulip



- Tulip unterstützt direkt die Spezifikation von Usability Features durch Erweiterung von Use Cases.
- Der Usability-Pattern-Katalog ist in Tulip integriert.
- Tulip berücksichtigt die für jedes Usability Pattern definierten Spezifikationsschablonen.
- **Support** gibt es im Tulip-Forum in ILIAS!



Vorgehen

1. Usability Pattern in Projekt (= Spezifikation) übernehmen
2. Globale Parameter und Funktionen spezifizieren
(benötigte und ggf. optionale)
3. Alle Abläufe oder Schritte Schritte annotieren, in denen das Usability Feature eingesetzt werden soll
4. Für jede Annotation jeweils Lokale Parameter und Use-Case-Erweiterungen spezifizieren (benötigte und ggf. optionale)
5. «Vollständigkeit» der Usability-Feature-Spezifikation prüfen

Schritte 3 – 5 bei neuen Use Cases ggf. wiederholen.

Tulip - Online Banking System (OnlineBankingSystem.tulip)

Neues Projekt Projekt laden Speichern Speichern unter ... Bericht Validieren Pattern Browser Info

Online Banking System Überweisung tätigen Kontoauszug anfordern Übersichtsdiagramm für Bankkunden Pattern Browser

Online Banking System Akteure Bankkunde Usability Patterns Abbruch Direkte Validierung Verarbeitungsanzeige Use Cases Anmeldung und Verwaltung 100: Am System anmelden Bankgeschäfte 200: Überweisung tätigen Kontoauszug anfordern Use-Case-Diagramme Übersichtsdiagramm für Bankkunden

Globales Undo Objektbezogenes Undo Abbruch Wiederholung Warnung Fortschrittsanzeige Verarbeitungsanzeige Ausführung im Hintergrund Gute Standardwerte Direkte Validierung Auto-Vervollständigung Nachsichtiges Format Filter Vorschau Papierkorb Dokumentwiederherstellung Automatisches Speichern Sicherheitskopie Assistent Systemstatus

Papierkorb

Beschreibung Beispiele Anwendung

1. Usability Pattern in Projekt übernehmen

- Welche Arten von Datenobjekten sollen beim Löschen zunächst vorläufig in den Papierkorb verschoben werden?
- Bestehen Gründe gegen die Verwendung eines Papierkorbs, z.B. Anforderungen des gesetzlichen Datenschutzes oder die Vertraulichkeit der zu löschen Daten?

Spezifikation: Globale Vorgaben

Papierkorb leeren Benötigt

Use Case, mit dem Benutzer alle oder einzelne Daten im Papierkorb endgültig löschen können. Akteur kann auch ein Zeitgeber/Timer sein, der nach einem gewissen Aufbewahrungszeitraum Daten automatisch (ggf. auch ohne Rückfrage an den Benutzer) aus dem Papierkorb entfernt.

Daten wiederherstellen Benötigt

Use Case, mit dem Benutzer einzelne oder alle im Papierkorb enthaltenen Daten wiederherstellen können.

Strategie Benötigt

Vorgabe der Papierkorb-Strategie: Festlegung, wie der Papierkorb beim Löschen von Daten einbezogen wird.

manuellen Löschen von Daten durch Benutzer fragt das System Benutzer, ob diese in den Papierkorb endgültig gelöscht werden sollen. Benutzer können eine der beiden Alternativen als Standardverhalten automatischen Löschen von Daten verschiebt das System diese stets in den Papierkorb.

Papierkorb-Inhalt Benötigt

Vorgabe der Arten von Datenobjekten, die in den Papierkorb verschoben werden können.

Beispiel: Fachliche Datensätze der Bestellverwaltung: Kunden, Artikel, Bestellungen, Rechnungen

Spezifikation: Individuelle Ergänzungen in Use Cases

Annotation @Schritt Papierkorb Benötigt

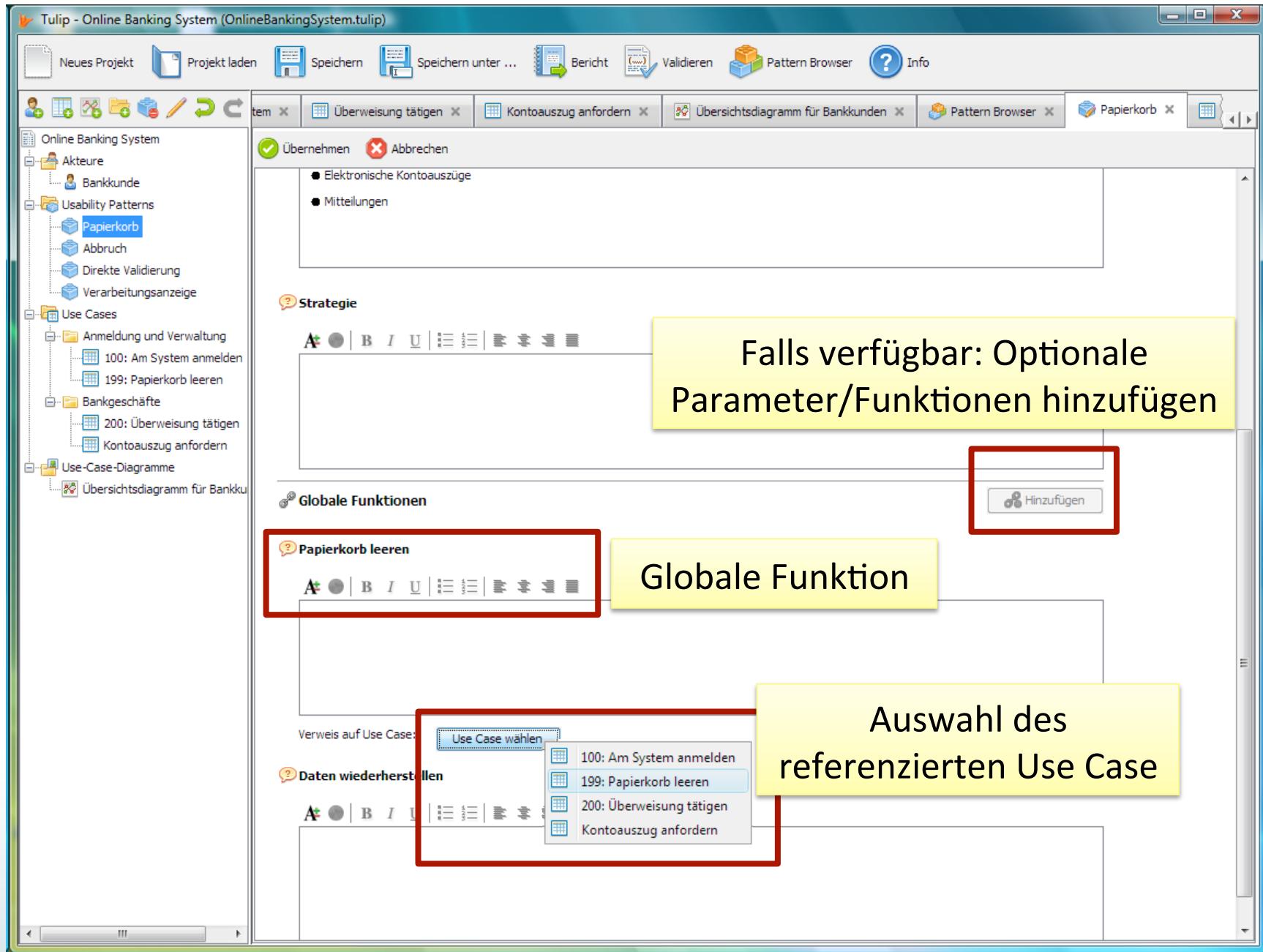
Das System soll Daten in diesem Schritt nicht endgültig löschen, sondern in den Papierkorb verschieben.

III

Usability Features im Projekt

The screenshot shows the Tulip interface with the following sections highlighted:

- Usability Patterns:** A red box highlights the 'Papierkorb' item under 'Usability Patterns' in the left sidebar.
- Papierkorb:** A yellow box contains the following text: "Allgemeine Beschreibung des Usability Features". Below it is a section titled "2. Globale Parameter und Funktionen spezifizieren".
- Globale Vorgaben:** A yellow box contains the text: "Eingabebereich für globale Parameter".
- Strategie:** A yellow box contains the text: "Tooltips als Hilfestellung".



Falls verfügbar: Optionale Parameter/Funktionen hinzufügen

Globale Funktion

Auswahl des referenzierten Use Case

Tulip - Online Banking System (OnlineBankingSystem.tulip)

The screenshot shows the Tulip application window with the following elements:

- Toolbar:** Neues Projekt, Projekt laden, Speichern, Speichern unter ..., Bericht, Validieren, Pattern Browser, Info.
- Left Sidebar:** Online Banking System, Akteure (Bankkunde), Usability Patterns (Papierkorb, Abbruch, Direkte Validierung, Verarbeitungsanzeige), Use Cases (Anmeldung und Verwaltung, 100: Am System anmelden, 100: Beendigung loggen).
- Central Panel:** A process step titled "Übernehmen" with options "Übernehmen" (checked) and "Abbrechen". It includes a rich text editor and a description: "Der Kunde gibt die *notwendigen* Informationen (Konto, Betrag, Empfänger und Verwendungszweck) an. Das System führt die Überweisung durch." Below it is an "Akteure:" section with "Bankkunde" and a "Bankkunde" icon.
- Bottom Panel:** An "Annotation hinzufügen" dialog titled "Element: Schritt 3" containing three sections: "Papierkorb" (with a "Hinzufügen" button), "Direkte Validierung" (with a "Hinzufügen" button), and "Verarbeitungsanzeige" (with a "Hinzufügen" button). The "Verarbeitungsanzeige" section is highlighted with a red border.
- Annotations:**
 - A yellow callout box labeled "3. Abläufe und Schritte annotieren" points to the "Übernehmen" step in the process.
 - A yellow callout box labeled "Auswahldialog für verfügbare Annotationen" points to the "Annotation hinzufügen" dialog.
 - A yellow callout box labeled "Annotation hinzufügen" points to the "Hinzufügen" button in the "Verarbeitungsanzeige" section of the dialog.

4. Lokale Parameter und Use-Case-Erweiterungen spezifizieren

The screenshot shows the Tulip interface for specifying use-case extensions. On the left, the project tree includes nodes like 'Online Banking System', 'Akteure', 'Usability Patterns', and 'Use Cases'. The 'Use Cases' node is expanded, showing 'Anm.' (Annotations) with items 100 and 199, and 'Bankgeschäfte' with items 200 and 210. The main workspace displays the 'Übernehmen' extension for use case 200. It has fields for 'Beschreibung' (Description) containing the note 'Der Kunde gibt die *notwendigen* Informationen (Konto, Betrag, Empfänger und Verwendungszweck) an.', 'Priorität' (Priority) set to 'Mittel', and 'Ebene' (Level) set to 'Benutzersicht'. Below this is the 'Normalablauf' (Normal Flow) section with steps 1, 2, and 3. Step 1: 'Bankkunde wählt Aktion "Überweisung"', Step 2: 'System zeigt Überweisungsauswahl an', Step 3: 'Bankkunde gibt Überweisungsdaten ein'. A red box highlights the 'Direkte Validierung' step, which specifies 'Gültige Werte: IBAN-Kontonummer nach Norm ISO-13616'. A yellow box labeled 'Annotation mit lokalem Parameter' points to this validation step.

Zusätzliche optionale Parameter oder Erweiterungen hinzufügen

Annotation mit lokalem Parameter

Tulip - Online Banking System (OnlineBankingSystem.tulip)

Prüfung starten

5. Vollständigkeit der Usability-Feature-Spezifikation prüfen

Beschreibung: Der Bankkunde ruft das System auf und meldet sich mit Kontonummer und PIN am System an.

Akteure: Validierung

Priorität:

Ebene:

Vorbed:

Folgende Validierungsfehler wurden gefunden:

- 100: Am System anmelden
- Direkte Validierung (Schritt 3): Fehlender lokaler Parameter 'Gültige Werte'

Einfache Prüfung auf fehlende/leere benötigte Parameter/Erweiterungen

Nachbedingung: Der Bankkunde ist am System angemeldet.

Alternativablauf4a

Vorbedingung: Anmeldedaten sind ungültig

4a1 System zeigt Fehlermeldung

Nachbedingung: Weiter in Schritt 2

The screenshot shows the Tulip tool interface for modeling a banking system. A validation dialog is open, listing errors found during the validation process. One specific error is highlighted with a red box: "100: Am System anmelden" under the heading "Direkte Validierung (Schritt 3): Fehlender lokaler Parameter 'Gültige Werte'". A yellow callout box to the right of the dialog states: "Einfache Prüfung auf fehlende/leere benötigte Parameter/Erweiterungen" (Simple check for missing/redundant required parameters/extensions). The main window displays the use case "Am System anmelden" with its steps and associated actors. The toolbar at the top includes a 'Validieren' button, which is also highlighted with a red box.

Online Banking System.pdf - Adobe Reader

Datei Bearbeiten Anzeige Fenster Hilfe

Werkzeuge Kommentar

2.3. Direkte Validierung

Eingaben des Benutzers soll das System möglichst unmittelbar auf Korrektheit hinweisen. Dies kann z.B. durch die Verwendung von Globale Vorgabe.

Annotation	@Schritt	Direkte Validierung
Globale Vorgabe	Darstellung	Bei ungültigen Eingaben soll ein Fehlersymbol (rotes X) neben dem Eingabefeld angezeigt werden.

Globale Parameter und Funktionen in der PDF-Spezifikation

2.4. Verarbeitungsanzeige

Eine animierte Darstellung der Ablaufschritte, noch damit

Annotationen und Lokale Parameter in der PDF-Spezifikation

Online Banking System.pdf - Adobe Reader

Datei Bearbeiten Anzeige Fenster Hilfe

Werkzeuge Kommentar

Ebene:		Benutzersicht
Priorität		Mittel
Normalablauf		
Vorbedingung:		Der Bankkunde ist am System angemeldet
1	Bankkunde	wählt Aktion "Überweisung tätigen"
2	System	zeigt Überweisungsdialog
3	Bankkunde	gibt Überweisungsdaten ein
Direkte Validierung		
Gültige Werte: IBAN-Kontonummer nach Norm ISO-13616		
prüft Überweisungsdaten		
Verarbeitungsanzeige		
5	System	führt Überweisung durch
6	System	zeigt Bestätigung
Nachbedingung:		Das System hat die Überweisung durchgeführt.
Abbruch		

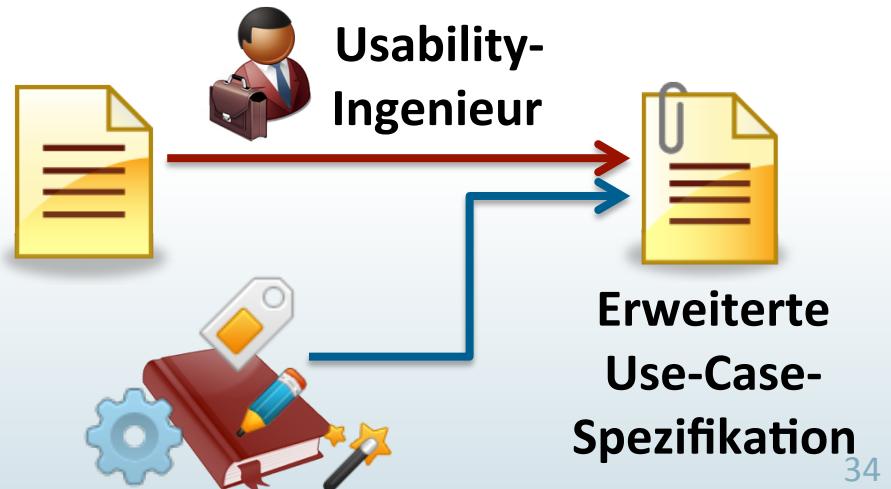
Usability Patterns im Sopra

«Was müssen wir als Nächstes tun?»



Usability Patterns im Sopra (2)

- **Schritt 2: Spezifizierung der konkreten Usability Features Ihrer Software**
- Entscheiden Sie (als Team), welche konkreten Usability Features Ihre Software enthalten soll!
- Erweitern Sie (als Team) Ihre Use-Case-Spezifikation mit Tulip oder «von Hand», in jedem Fall unter Verwendung der Spezifikationsschablonen!



Ausblick

- Bei Fragen zu Auswahl, Spezifikation, Umsetzung etc.
→ roeder@informatik.uni-stuttgart.de
- Prüfung der Feature-Spezifikationen (und vieler anderer Aspekte) in den **Spezifikationsreviews**
- Vor Beginn der Implementierung:
 - Befragung zur Spezifikation der Usability Features und zur Berücksichtigung in der Software-Architektur
 - Auswertung der Spezifikationen (durch Abt. SE)
- Im Systemtest: Prüfung der Feature-Umsetzung

Ende des Workshops

«*Haben Sie noch Fragen?*»

Nächster Meilenstein:
M3 – Abgabe Spezifikation + UI-Prototyp
15. November 2011, 12 Uhr

