

Usability-Workshop 1

Dipl.-Inf. Holger Röder

roeder@informatik.uni-stuttgart.de

Universität Stuttgart

Institut für Softwaretechnologie

Abteilung Software Engineering



Warum? Darum!

Report Selection

Fac	Order#	New	Print	View	Fax	Email	OCB	SSF View	Dupe Load	View Invent	Routing Sheet	Print Bill	Call Log	Cancelled
0	99004234	99031927												
0														

Mode From SC To SC

Air	ADT	ADT
Tariff	CAXR9-00-01	
Service	2D	D194
From	YVR	AF
To	YYZ	AE
Deliver By	06-12-02 17:00	
Clock Stop		

Find CAX#

CAX#:	100670861
Shpr Ref:	
B/L:	
PO#:	
GBL Num:	
Cons Ref:	
Billing Ref:	
Ref 5:	
MasterID:	0
MAWB:	
Statement:	0
Hold P/U:	
Non-Freight:	
Manifest Hold:	<input checked="" type="checkbox"/>
Print Hold:	<input checked="" type="checkbox"/>
Rate	

Charges: 761.50
Discount: 0%
SubTotal: 761.50
Accessorial: 40.00
DV: 0 0.00

FSC CAX 2.50% 38.08
Total: 839.58
Balance: 839.58

Addend
 Closed
 Post

Broker / Customs Agent

Broker: CANADIAN HARDWARE & H
Value: 0.00 **In:** USD
Notified:
Verbal Pod:
Notify on POD:
Hazmat:

SAVED

Units Type Description

	Units	Type	Description	Stated	ActWT	Dimensions	ChgWt	Rate	Charges
►	1	CRATE	CRATE	91	94	97 25x25x30	97	50.00	48.50
►	1	MAN	2 MAN P&D					40.00	40.00
►	2	CRATE	CRATE	500		1,426 60x48x48	1,426	50.00	713.00
*	0							0.00	0.00

3 | Accs: \$40.00 | DV: 0 | \$0.00 | 591 | 94# | 1523 | 1,523# | 761.50

Agenda

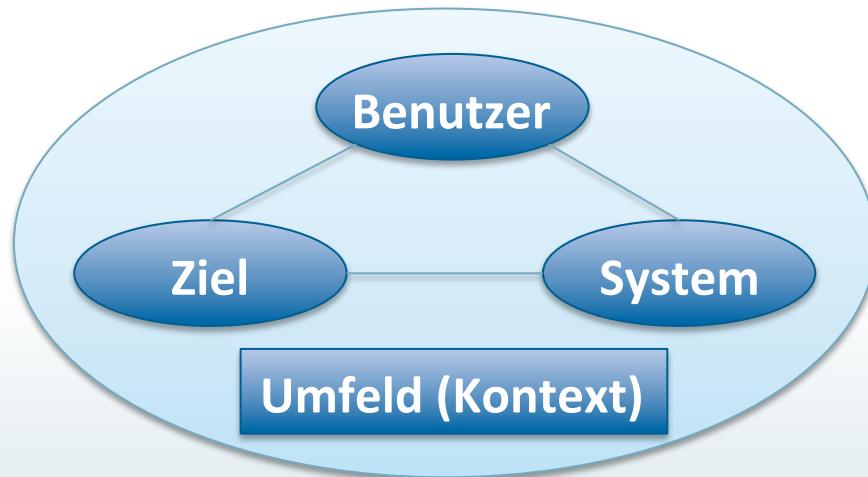
- Einführung: Usability?
- Personas und Szenarien
- UI-Prototypen
- Usability Patterns, Einsatz im Sopra
- Use-Case-Editor Tulip



Usability

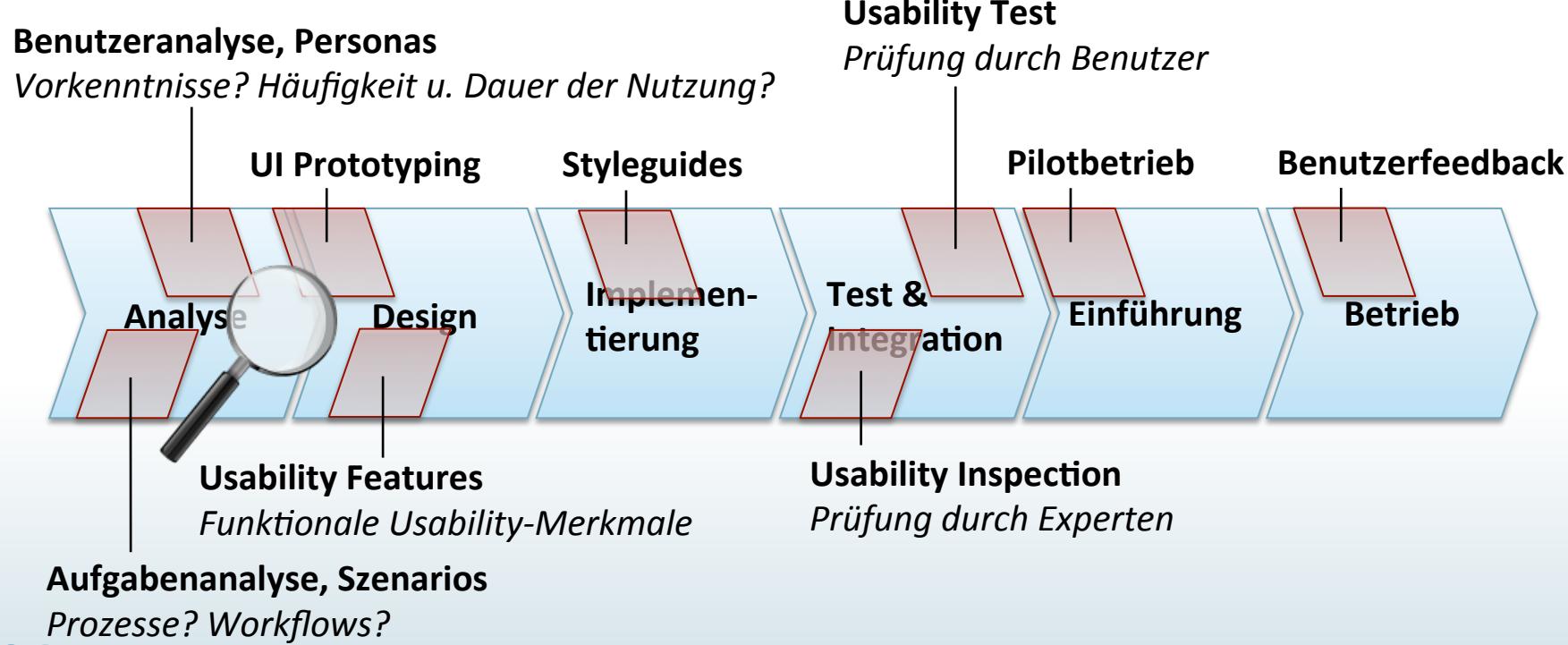
Usability bezeichnet das Ausmaß, in dem ein System von einem bestimmten Benutzer verwendet werden kann, um festgelegte Ziele in einem bestimmten Umfeld *effektiv, effizient und zufriedenstellend* zu erreichen.

(angelehnt an DIN EN ISO 9241)

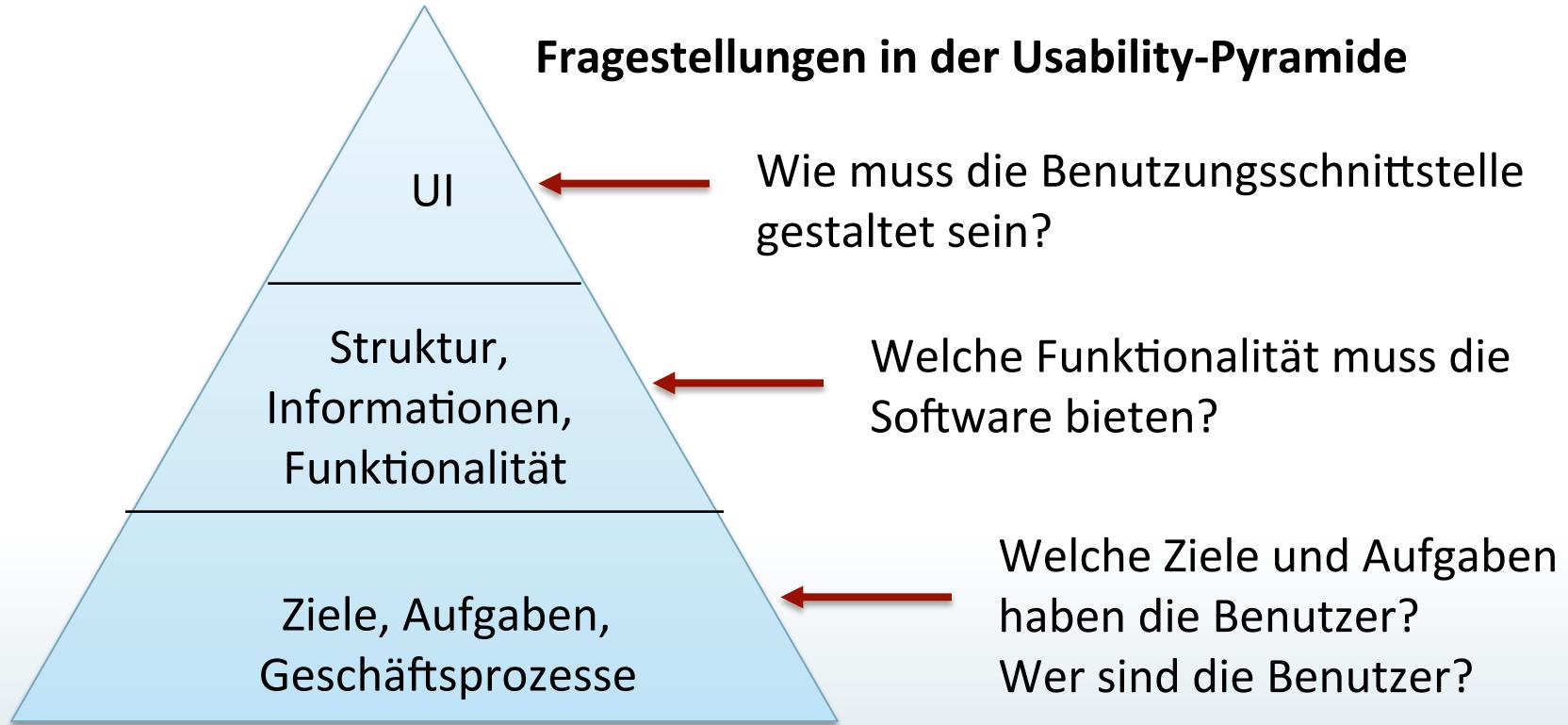


Usability Engineering (Software-Ergonomie)

- Techniken und Methoden zur Gestaltung gebrauchstauglicher Software – also Software mit guter Usability.



Usability ist mehr als eine schickes User Interface!



Usability ist überall ... nötig!



Personas und Szenarien

«Wer arbeitet eigentlich mit dem System
– und wie?»



Personas

- **Personas** sind Beschreibungen fiktiver Personen, die prototypisch Benutzer des Systems repräsentieren.
- Eine Persona steht stellvertretend für eine **Benutzergruppe**, die das System einsetzen wird.
- Wortherkunft: *Persona* = Maske der Schauspieler im griechischen Theater, die ihre jeweilige Rolle typisierte.



Beispiel einer Persona

- Hintergrund: Entwicklung eines neuen Systems zur Übermittlung von Werkstatt-Kostenvoranschlägen an Versicherungen



Peter Brunner

„Es gibt viel zu tun, packen wir's an!“

- ✓ Kfz-Mechanikermeister
- ✓ 53 Jahre alt, männlich
- ✓ verheiratet, 2 erwachsene Töchter
- ✓ seit 17 Jahren im Betrieb
- ✓ zuständig für die Weiterleitung von Kostenvoranschlägen an die Kfz-Versicherungen (bisher: per Brief/Fax)
- ✓ lediglich Computer-Grundkenntnisse
- ✓ erhofft sich Verringerung des anfallenden Papierkram

MethodicalMatt



"Cut scenes are excellent, I love being in the story."

"I like to check out water effects to see if it ripples as I walk through it, but it doesn't and I'm sad."

Matt:

- Likes understanding the storyline
- Methodically tries all weapons and special features
- Discovers the in-depth gaming tactics
- Watches all trailers and cut scenes
- Enjoys games that feel like an epic movie

Methodical Matt is a 28 year-old IT specialist at General Electric. Matt has a girlfriend, and they have two cats, Tinker and Bell. He likes hanging out with his friends at pubs, and checking out new bands at local theaters. Matt enjoys gaming on the weekend when his girlfriend is at work. It's just him and the cats.

Matt's a good gamer, been gaming since he was six years old. He downloads a few demos every month. Methodical Matt is interested in the whole package. He wants to know the storyline, why he is shooting, and wants to be immersed into another world. He reads everything, takes his time, and always inverts his controls. While gaming, he reads all the tutorials, and practices after every tip. To make sure he is doing everything properly, he reviews the controller layout frequently on the menu screen. He likes to explore every level before moving on, looking at all the details.

RunAndGunGary



"That was lame. They gave me the mini-gun right at the end...amazing gun and then there's nothing."

"I like the destructable environments. I'm noticing that the pillars they chip away which is nice."

Gary:

- Likes jumping right into the game
- Thinks game-play experience is most important
- Expects controls for shooters to be consistent
- Is happy when demos have multiplayer mode
- Wants big guns
- Is impressed with destructible environments

Gary is a freshman at Michigan State. He is studying engineering, but is thinking about switching to business. He lives in the dorms and can't wait to have an apartment off-campus next year with his buddies. His favorite food is pizza, and he collects pizza magnets on his fridge. He tries to keep up on his studies, but finds it hard to turn down some halo multi-player action with his friends.

Gary owns a lot of games and he downloads demos based on the latest internet reviews. Once downloaded, he jumps right in the game. Gary likes when the demo gives him the option to skip past all the trailers and cut scenes—that's secondary to him. Gary wants to check out the big guns! He is sensitive to the artificial intelligence, he doesn't like running out of ammo, and thinks good graphics are a bonus. He likes to go for high scores and kills, so he can compare with his friends. If the demo has a multiplayer option, Gary is always quick to grab a friend and test it out. They are always on the lookout for any new multiplayer games.

CasualCarl



"I feel dumb at this point and I can't ask the guy because I killed him."

"Tutorial is helpful. It shows you how to play as you need it. I didn't need to know how to crouch, but now I need do and it's nice to know."

Carl:

- Enjoys when tutorials point out all the features
- Likes when demos have difficulty options
- Doesn't play demos all the way through
- Does not like to die frequently
- Likes having maps & objectives

Casual Carl is 30 years old and just started at Initech in the finance department. He is a busy guy, always on the go with helping friends move, or building a deck with his dad. Carl used to play games on a PC, but recently bought a Xbox 360.

Carl downloads demos sporadically. Usually, if a friend recommends a game, he will try it out. If he isn't enjoying the game in the first 10 minutes, he will lose interest and quit. He has average gaming skills, and wants to have fun, but doesn't try to get the highest score. He watches some trailers and cut scenes, but not all. He reads the tutorials closely, and looks down at the controllers frequently to find the appropriate buttons. Since Carl is new to console games, he tends to die a lot; he doesn't like it, and feels good when he gets through a level without dying.

Eigenschaften von Personas

- Eine Persona beschreibt **Eigenschaften** von Benutzern, die für die Gestaltung des Systems relevant sein können:
 - Beruf, Tätigkeiten und Verantwortlichkeiten, Arbeitsumfeld
 - Ziele und Aufgaben
 - Ausbildung, Kenntnisse und Fähigkeiten
 - Wünsche, Werte, Vorlieben und Ängste
- Personas erhalten eine (fiktive) **Persönlichkeit**, um die Benutzer des Systems realistisch abzubilden:
 - Name, Alter, Geschlecht, Familiärer Hintergrund
 - Foto oder Skizze, Motto, Zitate

Einsatz und Nutzen von Personas

- Personas werden bei der **Anforderungsanalyse** identifiziert und anschließend in der Anforderungsspezifikation beschrieben.
- **Ziele und Bedürfnisse** der Benutzer rücken in den Vordergrund.
- Personas erleichtern es Entwicklern, die **Perspektive der Benutzer** einzunehmen.
- Fragen nach der tatsächlich benötigten Funktionalität oder der **Priorisierung von Anforderungen** können aus Benutzersicht diskutiert werden.
 - Wer benötigt diese Funktionalität?
 - Würde *Peter* diese Funktion nutzen?

Aufgabe 1: Persona

- Entwickeln Sie eine Persona, die einen typischen Benutzer des Sprintnote Tools beschreibt!
 - Sopra-Gruppe mod 3 = 0 → **Administrator**
 - Sopra-Gruppe mod 3 = 1 → **Entwicklerin**
 - Sopra-Gruppe mod 3 = 2 → **Übersetzer**
- Schreiben Sie deutlich und lesbar! ☺



Arbeit in der Sopra-Gruppe
Bearbeitungszeit: 5 Minuten

Anschließend Vorstellung und Diskussion

Szenarien

- **Szenarien** beschreiben beispielhaft, wie ein Benutzer das System benutzt, um ein Ziel zu erreichen.
- Der Ablauf eines Szenarios wird in einzelnen Schritten geschildert.
- Kern des Szenarios ist die konkrete **Interaktion des Benutzers mit dem Systems**. Daneben kann das Szenario auch Schritte umfassen, die ohne Benutzung des Systems ablaufen, aber für das Verständnis der Handlung notwendig sind.
- Szenarien modellieren Aspekte der Systemnutzung, die für die Gestaltung der Lösung relevant sind. Sie können sowohl **normale Abläufe** als auch **Ausnahmesituationen** beschreiben.

Beispiel eines Szenarios

Szenario 12: Übermittlung eines Kostenvoranschlags durch die Kfz-Werkstatt

Es ist 14 Uhr. Ein Kunde bringt ein Auto mit Unfallschaden in die Werkstatt. Peter Brunner begutachtet den Schaden und notiert Kontakt- und Versicherungsdaten des Kunden. Peter erstellt mit dem (*vorhandenen*) Werkstattsystem einen Kostenvoranschlag für die Reparatur. Zusätzlich fotografiert er den Schaden und überträgt die digitalen Fotos auf den Rechner.

Peter öffnet Peter das APS-System und meldet sich an. Er trägt die Daten des Kunden ein und übermittelt den Kostenvoranschlag und die Schadenfotos.

Da es bis zu 24 Stunden dauert, bis das APS-System per E-Mail die Reparaturfreigabe erteilt, meldet er sich vom System ab und wendet sich wieder anderen Aufgaben zu.

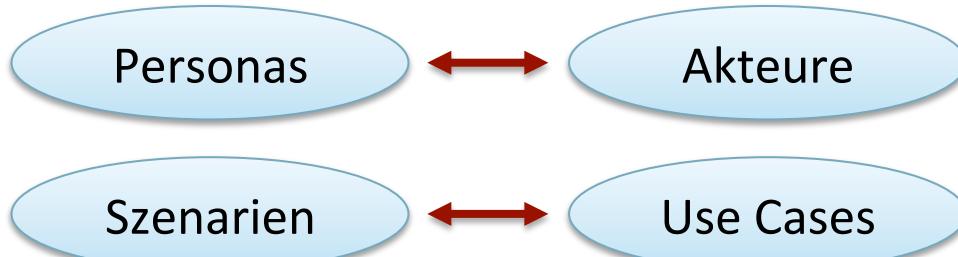
Einsatz von Szenarien

- Szenarien werden – wie Personas – in der **Anforderungsanalyse** entwickelt und modelliert.
- Szenarien können gut in **Kombination mit Personas** eingesetzt werden. Die handelnden Figuren in den Szenarien werden durch Personas dargestellt.
- Szenarien basieren auf den identifizierten **Anforderungen** an das System und veranschaulichen diese.
- Szenarien werden **iterativ entwickelt**, geprüft, korrigiert und verfeinert. Benutzer sollten bei der Entwicklung der Szenarien einbezogen werden, z. B. im Rahmen gemeinsamer **Szenario-Workshops**.

Nutzen von Szenarien

- Szenarien dienen Software-Leuten und Klienten, ein **einheitliches Verständnis** der Anforderungen zu entwickeln, die das System umsetzen muss.
- Szenarien sind **leicht verständlich** und weniger abstrakt als Use Cases. Häufig umfasst ein Szenario auch mehrere Use Cases.
- Klienten – insbesondere Benutzer – können Szenarien prüfen und ergänzen und auf diese Weise Anforderungen an das System stellen.

Einordnung der Techniken



- Personas und Szenarien sind **benutzerorientierte Techniken**. Sie beschreiben konkret typische Benutzer und die typische Nutzung des Systems.
- Akteure und Use Cases sind **systemorientierte Techniken**. Sie beschreiben abstrakte Benutzer (Rollen) und abstrakte Abläufe.
- Die Techniken konkurrieren nicht, sondern sollten kombiniert werden!

Aufgabe 2: Szenario

- Geben Sie Ihre Persona weiter an die Gruppe mit der nächsthöheren Gruppennummer ($1 \rightarrow 2, \dots, 23 \rightarrow 1$)!

- Formulieren für die erhaltene Persona ein Szenario, dass beschreibt, wie die Persona das Sprintnote Tool für eine für sie typische Aufgabe einsetzt.
- Berücksichtigen Sie die beschriebenen Charakteristika der Persona!



Arbeit in der Sopra-Gruppe
Bearbeitungszeit: 8 Minuten

Anschließend Vorstellung und Diskussion

UI-Prototypen

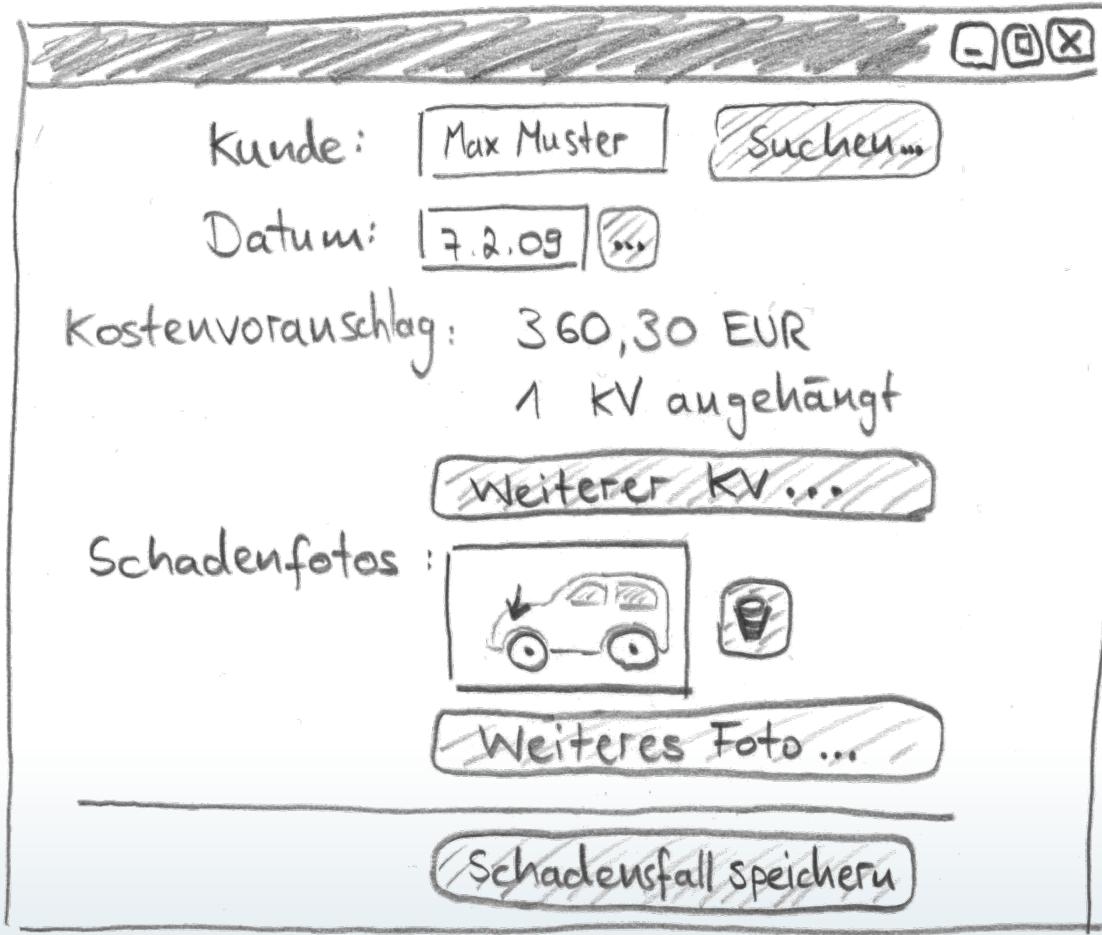
«Wie soll das System eigentlich aussehen?»



UI-Prototypen

- User-Interface-Prototypen (Mockups) sind **Modelle der Benutzungsschnittstelle** des späteren Systems.
- UI-Prototypen können für die Gestaltung, Prüfung und Verbesserung von Aspekten der Benutzungsschnittstelle eingesetzt werden.

Papierprototyp («low-fidelity»)



Wireframe (digitales Drahtmodell)

Schadensfall erfassen

Kunde: Max Muster (18390) [Kunde suchen ...](#)

Schadensdatum: 13.01.2009

Kostenvoranschlag: 744,50 EUR
1 Kostenvoranschlag angehängt (14.01.2009)
[Weiteren KV anhängen ...](#)

Schadenfotos: 1 Schadensfoto angehängt

[Weiteres Foto anhängen ...](#)

Schadensfall speichern

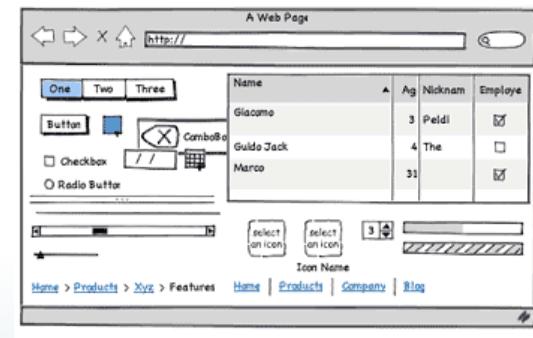
Ausführbarer Prototyp (``high-fidelity``)



Einsatz von UI-Prototypen

- Klärung der Anforderungen, Diskussionsgrundlage
- Konzeption und Spezifikation der Benutzungsschnittstelle:
Dialogaufbau, Navigation, Informationsdarstellung
- Ergonomische und ästhetische Feingestaltung der
Benutzungsschnittstelle

- Viele Werkzeuge verfügbar, zum Beispiel
 - Pencil: <http://pencil.evolus.vn/>
 - Balsamiq: <http://balsamiq.com/>



Plädoyer für Papierprototypen!

- Papierprototypen bieten sich als Technik für frühe Entwürfe der Benutzungsschnittstelle an:
 - Sie sind einfach und schnell erstellbar.
 - Sie schränken die Kreativität nicht ein.
 - Die Hemmschwelle ist niedrig, alle Klienten können mitarbeiten.
 - Die offensichtliche Distanz zur späteren Software erleichtert Diskussion von Grundsatzfragen.
 - Sie können leicht weggeworfen und neu erstellt werden!

Aufgabe 3: Papierprototyp

- Zeichnen Sie einen Papierprototypen für den Dialog, in dem der Administrator die möglichen Werte für die projektabhängigen Attribute (SB-Ver., NE-Ver., Komponente) und für die projektunabhängigen Attribute (Eintragstyp) festlegen kann!



Arbeit in der Sopra-Gruppe
Bearbeitungszeit: 5 Minuten

Anschließend Vorstellung und Diskussion

Usability Patterns

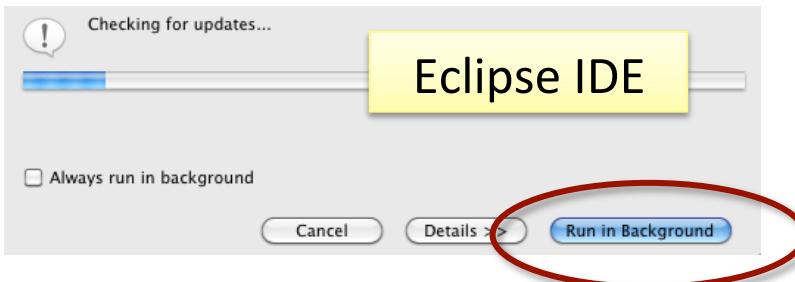
«*Mit welchen Features können wir eigentlich die Usability verbessern?*»



Usability Patterns

- Forschungsprojekt der Abteilung SE seit 2009
- **Idee:** Funktionale Merkmale von Software, die die Usability verbessern, systematisch und früh im Entwicklungsprozess berücksichtigen!

Usability Features



Ausführung der Update-Prüfung im Hintergrund

- Funktionalität, die primär die Usability verbessert
 - In vielen Fällen Querschnittsfunktionalität
 - Häufig große Auswirkungen auf Software-Architektur und Implementierung
-
- Abstraktion von konkreten Lösungen erlaubt Identifizierung von **Lösungsmustern**

Usability Patterns

- Funktionale Lösungsmuster für typische Interaktionsprobleme in einem bestimmten Kontext
- Beschreibung wiederverwendbarer Usability Features in strukturierter, natürlichsprachlicher Form:
Problem – Lösung – Kontext – ...
- Abbildung der Beziehungen zwischen verschiedenen Patterns aus ergonomischer Sicht (Mustersprache)
- Usability Patterns dienen als **Gedächtnisstütze**,
Diskussionsgrundlage und **Anleitung** für die Projektbeteiligten!

«Ausführung im Hintergrund»

Name

Ausführung im Hintergrund

Ausführung im Hintergrund

Background Task

Benutzer möchten nicht auf die Ausführung einer langdauernden Aktion warten, sondern währenddessen weiter mit dem System arbeiten.

Problem

Problem Benutzer möchten nicht auf die Ausführung einer langdauernden Aktion warten, sondern währenddessen weiter mit dem System arbeiten.

Lösung

Biete Benutzern an, langdauernde Aktionen im Hintergrund auszuführen.

Download kann, abhängig von der Dateigröße, längere Zeit dauern und wird deshalb im Hintergrund ausgeführt. Benutzer können mit dem Web-Browser weiter im Web surfen und müssen nicht den Abschluss des Downloads abwarten. Der Fortschritt des Downloads wird in der Statuszeile des Browser-Fensters dargestellt; ein eingebetteter Hinweis informiert Benutzer über den erfolgreichen Abschluss des Downloads.

Lösung

Beispiel

Google Chrome 8.0 (Mac OS X): Download von Dateien

Der Web-Browser Chrome führt Downloads im Hintergrund aus. Aktive Downloads werden in der Statuszeile am unteren Rand des Browser-Fensters angezeigt; Benutzer können dort u. a. den Fortschritt und die verbleibende Dauer des Downloads erkennen und z. B. auch über einen Link den Download abbrechen.



Download im Hintergrund in Chrome

Kontext

Begründung

- Aktionen, für deren Ausführung das System längere Zeit benötigt und die während der Ausführung keine Interaktion mit dem Benutzer verlangen
- Aktionen, die das System unabhängig von anderen Aktionen im Hintergrund ausführen kann

Die Ausführung langdauernder Aktionen im Hintergrund erlaubt Benutzern ein effizienteres Arbeiten mit dem System: Benutzer müssen nicht auf das Ende der Ausführung warten, sondern können parallel weiter mit dem System arbeiten.

Rationale

Risiken, Nachteile, Kosten

Die Ausführung einer Aktion im Hintergrund unterbricht in vielen Fällen eine Dialoginteraktion zwischen Benutzer und System: das System erwartet, dass Benutzer reagiert. Benutzer, sondern führt für längere Zeit einen Verarbeitungsschritt, der deshalb leicht erkennbar sein muss, dass die gewünschte Aktion nicht mehr ausgeführt wird. Gleiches gilt, wenn der Verarbeitungsschritt abgeschlossen ist, aber Benutzer gut sichtbar und verständlich informiert werden müssen auf die Ausführung der Aktion warten.

Zusammenspiel

Ergänzung

Abbruch

Abbruch von Aktionen, die im Hintergrund ausgeführt werden.

Kosten

Zusammenspiel

Anforderungsspezifikation

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

Vorgabe

Darstellung

Benötigt

Vorgabe für die einheitliche Darstellung der im Hintergrund ausgeführten Aktionen.

Spezifizierte, für welche Interaktionen das Usability Pattern „Ausführung im Hintergrund“ eingesetzt wird. Annotiere und ergänze dazu vorhan-

Annotation

@ Schritt

Hintergrundausführung

Analyse

Spezifikation

Der Verarbeitungsschritt kann auf Wunsch des Benutzers im Hintergrund ausgeführt werden.

Parameter

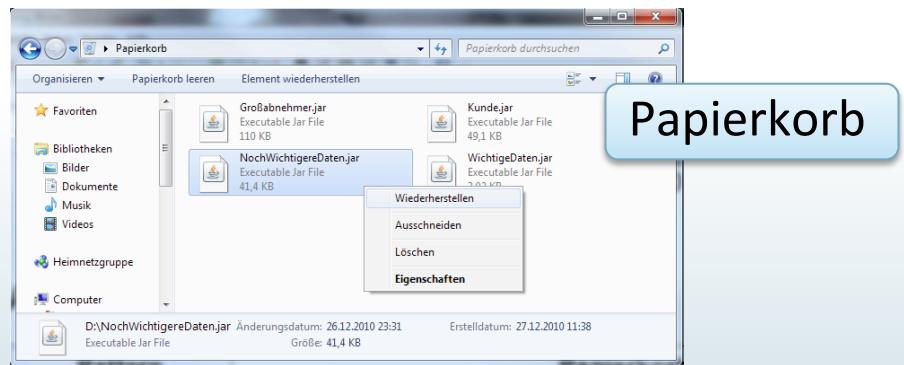
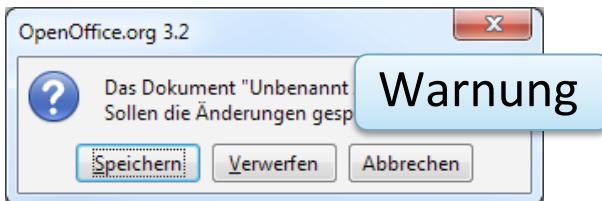
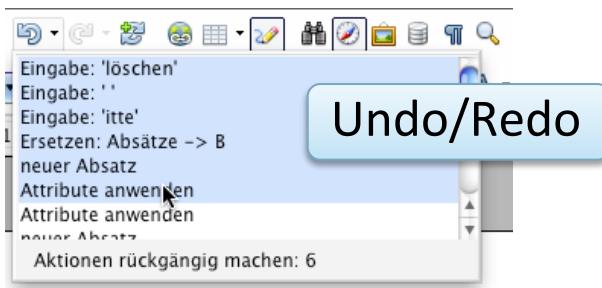
Standardmäßig im Hintergrund

Optional

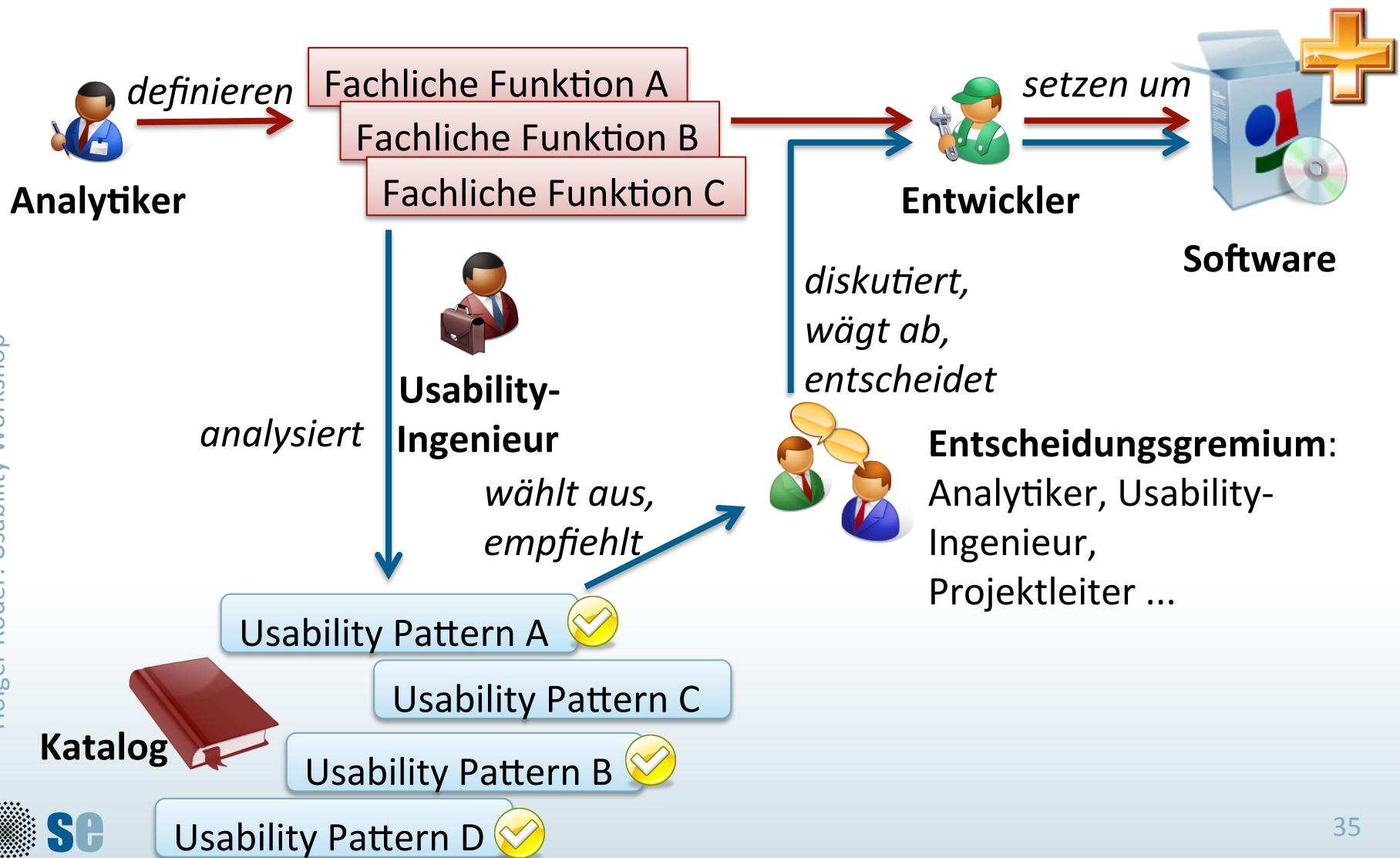
Angabe, ob der Schritt standardmäßig automatisch im Hintergrund ausgeführt wird oder ob Benutzer die Hintergrundausführung zunächst explizit auswählen müssen.

Katalog von Usability Patterns

20 Usability Patterns für interaktive Informationssysteme



Usability Patterns: Prozess

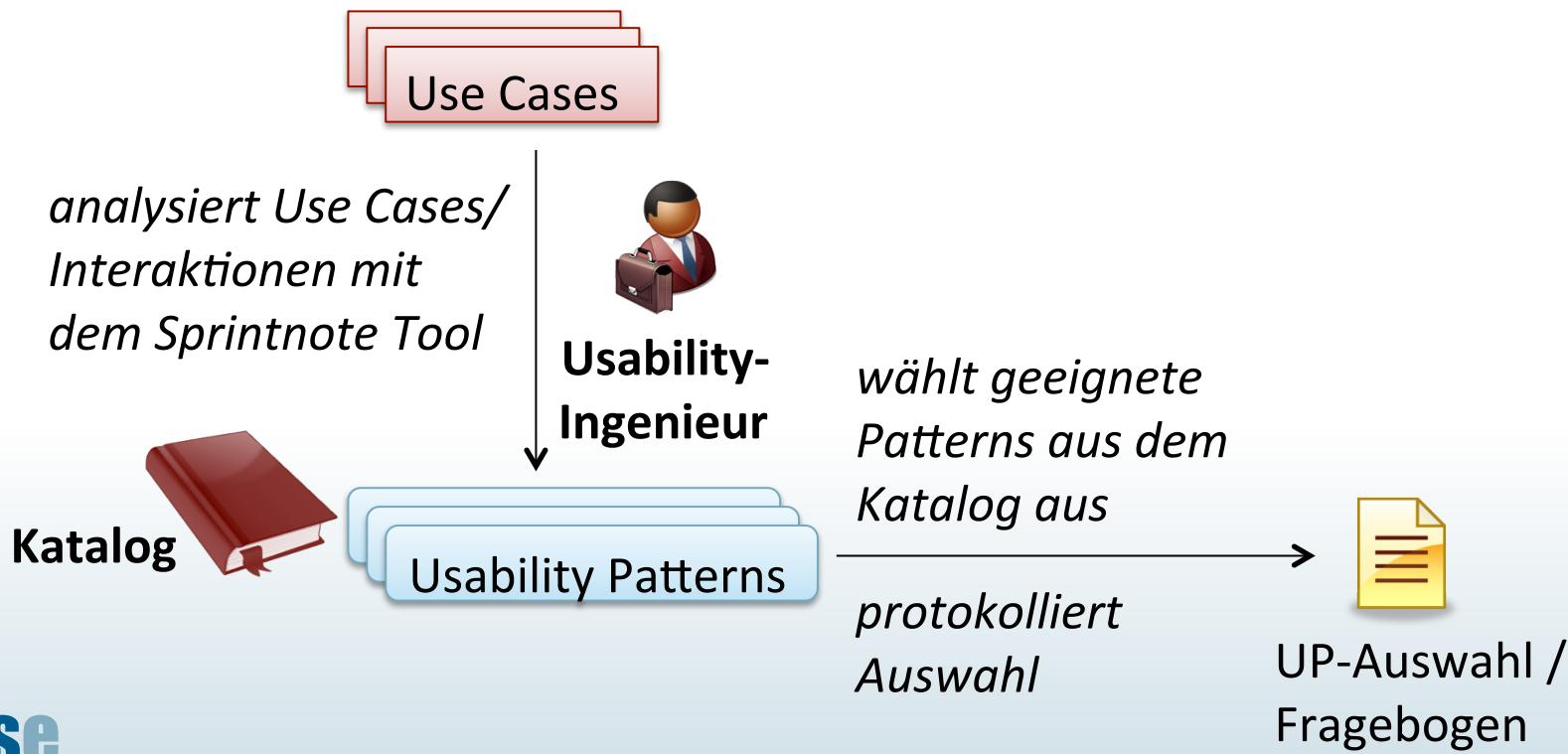


Usability Patterns im Sopra

- Sie sollen im Sopra Usability Patterns bei der Entwicklung nutzen, um die Usability Ihrer Software zu verbessern!
 - Sie erhalten dazu einen Katalog mit 20 Usability Patterns.
-
- **Schritt 1: Auswahl geeigneter Usability Patterns**
 - Schritt 2: Spezifizierung der konkreten Usability Features ihrer Software (→ Usability-Workshop 2)
 - Schritt 3: Umsetzung und Test der Usability Features

Auswahl geeigneter Patterns

- Lesen Sie den Katalog durch und prüfen Sie, welche Patterns für das Sprintnote Tool geeignet erscheinen!
- Dokumentieren Sie Ihre **unverbindliche** (!) Auswahl in einem Fragebogen bis nächsten Freitag, 04.11.2011!



Spezifikation der Usability Features



- Thema des zweiten Workshops am 04.11.2011. **Ausblick:**
 - Spezifizieren Sie die konkreten Usability Features ihrer Software durch Erweiterung Ihrer Use Cases!
 - Nutzen Sie die verfügbaren Spezifikationsschablonen!
 - Setzen Sie den Use-Case-Editor Tulip ein, der diese Erweiterungen unterstützt!

Tulip

«Wie können wir eigentlich Use Cases spezifizieren?»



Tulip

- Tulip ist ein **Use-Case-Editor**, der seit Anfang 2011 in der Abteilung SE entwickelt wird (Diplomarbeiten etc.).
- Tulip unterstützt direkt die Spezifikation von Usability Features durch Erweiterung von Use Cases. Der Usability-Pattern-Katalog ist in Tulip integriert.
- <http://www.iste.uni-stuttgart.de/se/werkzeuge/tulip.html>



Verwaltung der Spezifikation („Projekt“)

Usability Pattern Katalog

Verwaltung der Akteure/UCs/...

Hauptfenster: Darstellung der Akteure/Patterns/UCs

Übersicht über die Inhalte der Use-Case-Spezifikation

The screenshot shows the Tulip Online application interface. At the top, there's a toolbar with various icons like 'Neues Projekt', 'Projekt laden', 'Speichern', etc. Below the toolbar is a menu bar with 'Bearbeiten', 'Löschen', 'Undo', 'Redo'. The main area is divided into several sections:

- Left Sidebar:** Shows a tree view of the project structure under 'Tulip - Online'. It includes sections for 'Abbruch', 'Direkte Validierung', 'Verarbeitungsanzeige', 'Use Cases' (with sub-options like 'Anmeldung und Verwaltung', 'Bankgeschäfte', 'Überweisung tätigen'), 'Use-Case-Diagramme', and 'Übersichtsdiagramm für Bankku...'. A red box highlights the 'Use Cases' section.
- Top Right:** A yellow box labeled 'Usability Pattern Katalog' contains a list of patterns: 'Überweisung tätigen (200)', 'Bankkunde', 'Mittel', and 'Benutzersicht'.
- Main Content Area:** A large yellow box labeled 'Übersicht über die Inhalte der Use-Case-Spezifikation' contains a use case diagram for 'Überweisung tätigen (200)'. It shows a 'Normalablauf' (Normal Flow) with steps: 1. Bankkunde wählt Aktion "Überweisung tätigen", 2. System zeigt Überweisungsdialog, 3. Bankkunde gibt Überweisungsdaten ein, 4. System prüft Überweisungsdaten, 5. System führt Überweisung durch, 6. System zeigt Bestätigung. A red box highlights the 'Normalablauf' section.
- Bottom:** A red box highlights the 'Abbruch' (Cancel) button at the bottom of the main window.

Tulip - Online Banking System (OnlineBankingSystem.tulip)

Neuer UC...

Übernehmen Abbrechen

Kontoauszug anfordern

ID:

Ziel:

Der Kunde möchte seinen aktuellen Kontoauszug als PDF-Dokument

Beschreibung:

Akteure: Bankkunde

Aktor hinzufügen

Priorität: Hoch

Ebene: Benutzersicht

Akteure im UC
„Stern“ = Haupakteur

Normalablauf

Vorbedingung: Der Kunde ist am System angemeldet

Bedingung hinzufügen

1 System Das System zeigt den Dialog "Kontoauszug" an.

2 Bankkunde

Schritt hinzufügen

Nachbedingung:

Hinweise auf fehlende Angaben

Verzweigung in Sonderfall „Blitz“

Akteurliste

SG

Tulip - Online Banking System (OnlineBankingSystem.tulip)

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

Übernehmen Abbrechen

Übersichtsdiagramm für Bankkunden

Beschreibung:

Übersicht über die wesentlichen Use Cases für Bankkunden

Einfügen von UCs per Drag'n'Drop

Use-Case-Diagramm:

Übersichtsdiagramm für Bankkunden

Online Banking System

Am System anmelden

Überweisung tätigen

Bankkunde

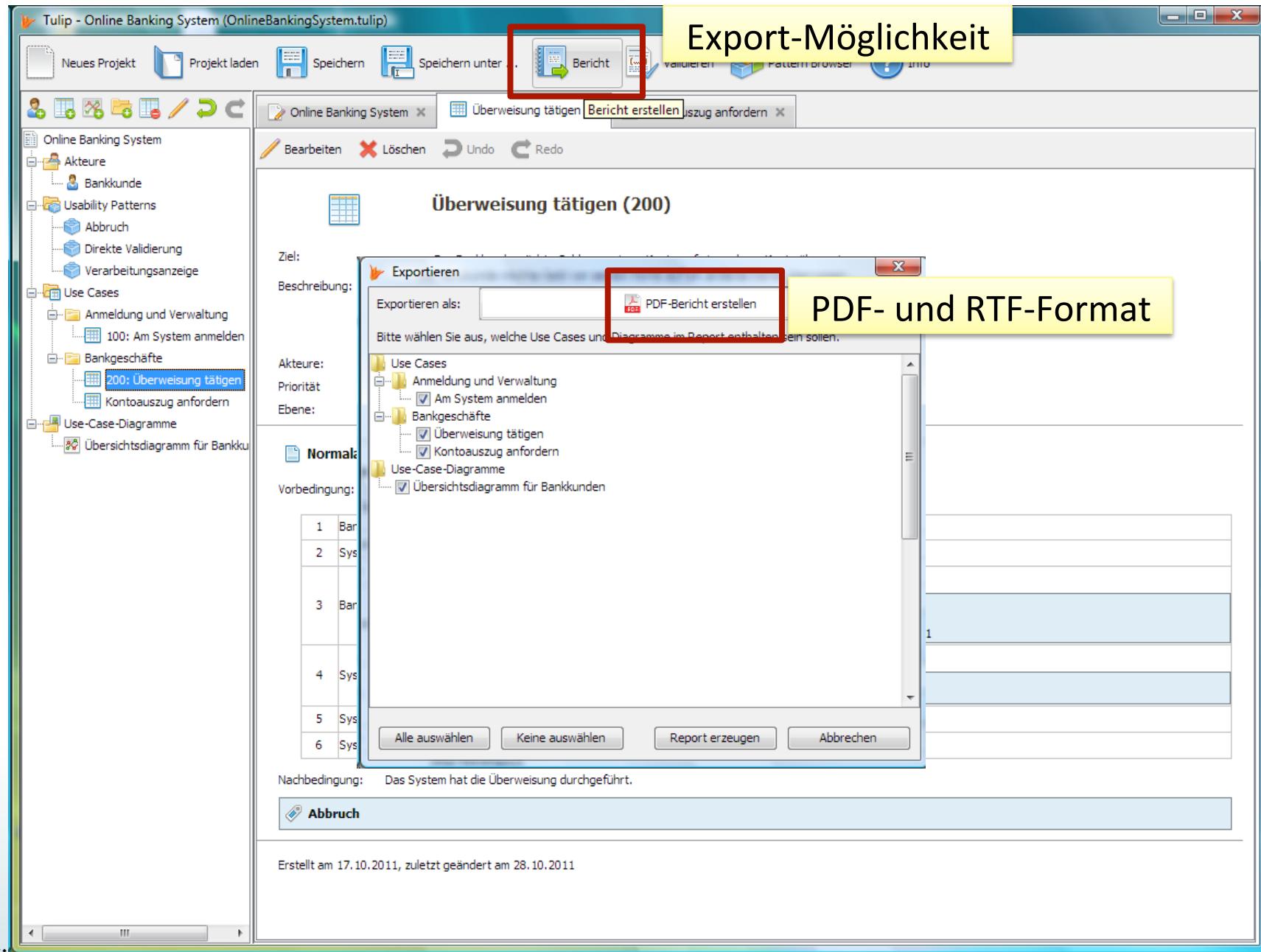
Use-Case-Diagramme

Übersichtsdiagramm für Bankkunden

Einfache Use-Case-Diagramme

43

```
graph TD; Actor((Bankkunde)) --- UC1((Am System anmelden)); Actor --- UC2((Überweisung tätigen)); UC1 --- System[Online Banking System]; UC2 --- System;
```



Online Banking System.rtf [Kompatibilitätsmodus]

Schriftart: Helvetica 10 Absatzzeichen und -abstände: 0 0 Typografie: f 123 A Formatvorlagen: Aa Einfügen: Textfeld Designs: Aa

Absatz: 1 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

3. Use Cases

3.1. Use Cases/Anmeldung und Verwaltung

100: Am System anmelden

Ziel:	Der Bankkunde möchte sich am System anmelden und legitimieren, um die Funktionen des Online Bankings nutzen zu können.	
Akteure:	Bankkunde	
Beschreibung:	Der Bankkunde ruft das System auf und meldet sich mit Kontonummer und PIN am System an.	
Ebene:	Benutzersicht	
Priorität	Mittel	
Normalablauf		
Vorbedingung: Das System ist in Betrieb. Der Bankkunde ist nicht am System angemeldet.		
1	Bankkunde	ruft System auf
2	System	zeigt Anmeldedialog
3	Bankkunde	gibt Anmelddaten zur Authentifizierung ein Direkte Validierung Gültige Werte: Kontonummer muss numerisch und 7-stellig sein.
4	System	prüft Anmelddaten Bedingung für Sonderfall Anmeldungsdaten sind ungültig Alternativablauf 4a Verarbeitungsanzeige
5	System	zeigt Willkommensdialog
Nachbedingung: Der Bankkunde ist am System angemeldet.		
Alternativablauf 4a		
Vorbedingung: Anmeldungsdaten sind ungültig		
4a1	System	zeigt Fehlermeldung
Nachbedingung: Weiter in Schritt 2		

RTF-Export

Einfache Integration
in Anforderungs-
spezifikation

Tulip

- Nutzen Sie Tulip für die bequeme Spezifizierung Ihrer Use Cases und der Usability Features im Sopra! ☺
- Helfen Sie uns bei der Weiterentwicklung: Bug Reports, Wünsche, Anregungen, Mitentwicklung ...
- **Support** gibt es im Tulip-Forum in ILIAS!



Ende des Workshops

«*Haben Sie noch Fragen?*»

Usability-Workshop 2:
Freitag, 4. November, 11:30 Uhr

