

Software Engineering 1



Anna Prenzel

Einführung

Ziele der heutigen Veranstaltung:

- ✎ Use Cases als Mittel der Anforderungsspezifikation kennen lernen
- ✎ Vertiefung: verschiedene Erscheinungsformen von Anforderungen
- ✎ Aufbau eines Pflichtenhefts kennen lernen

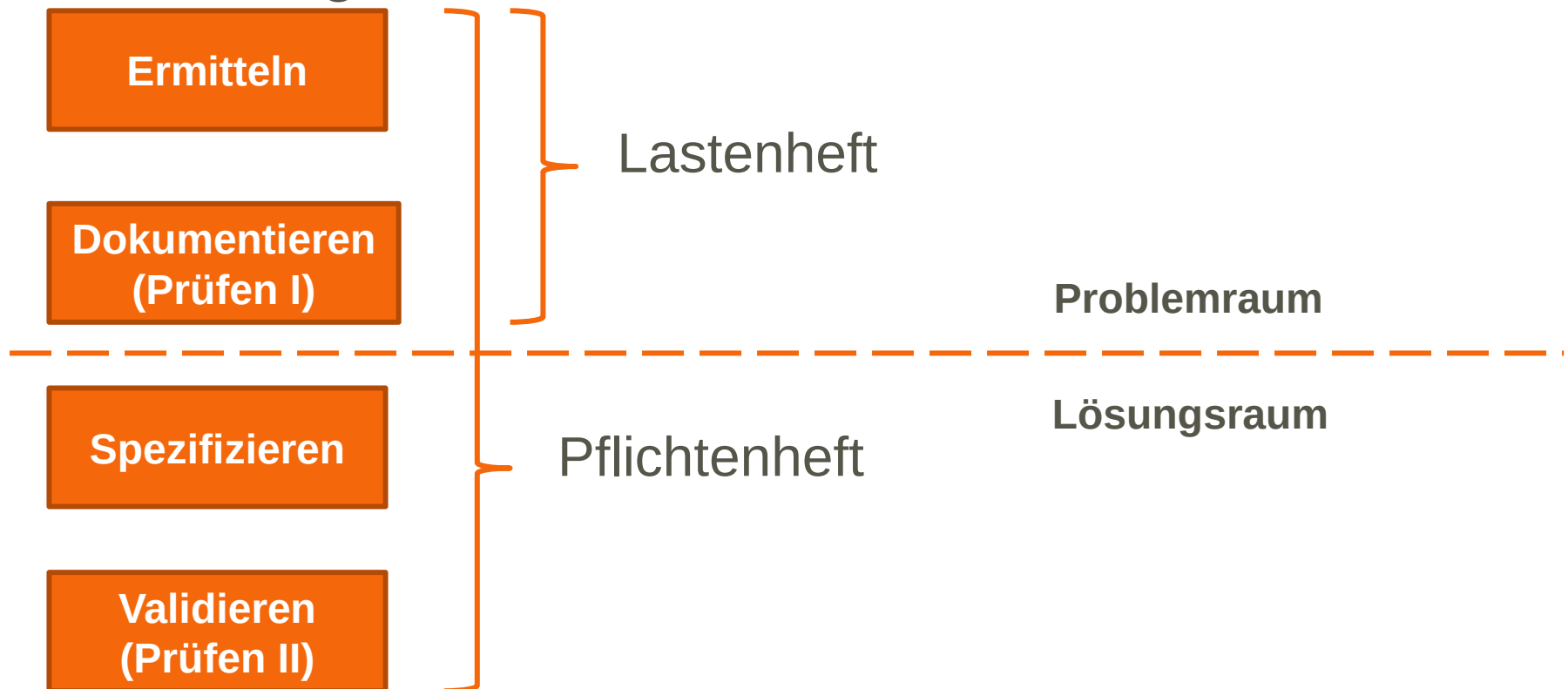
Wiederholung

Themen, die wir besprochen haben

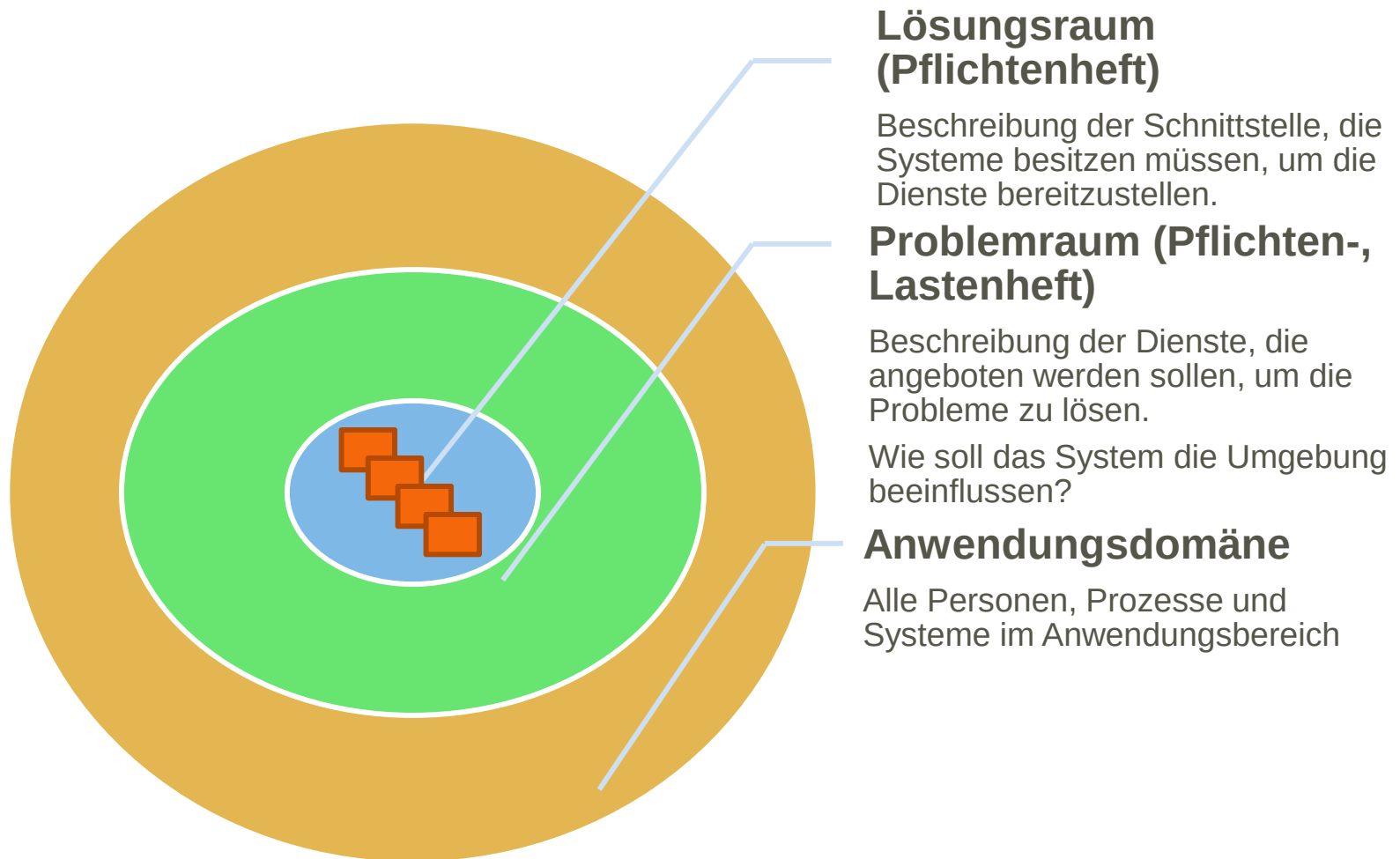
- ✎ Funktionale / nicht-funktionale Anforderungen
- ✎ Dokumentation im Lasten- und Pflichtenheft
- ✎ Probleme bei der Erfassung von Anforderungen
- ✎ Qualitätskriterien für Anforderungen
- ✎ Stufen des Requirements Engineering
- ✎ Aufbau des Lastenhefts zur Dokumentation von Anforderungen

Spezifikation von Anforderungen

RE als weiche Übergangsphase zwischen Problem- und Lösungsraum



Spezifikation von Anforderungen



Spezifikation von Anforderungen

Definition (Spezifikation)

Spezifikation ist die exakte, vollständige und für eine Überprüfung geeignete Beschreibung eines Gegenstands oder eines Prozesses.

- ✎ In den Phasen „Ermitteln“ und „Dokumentieren“ können Anforderungen noch einen informellen Charakter haben.
- ✎ In der Phase „Spezifizieren“ werden Anforderungen konkreter und detaillierter beschrieben.
- ✎ Ziel: Verbindliche Grundlage für den SW-Entwurf schaffen.

Spezifikation von Anforderungen

Etablierte Methode: Use Cases (Anwendungsfälle)

Definition (Use Case)

Ein Use Case ist die Beschreibung einer Abfolge von zielgerichteten Interaktionen zwischen Akteuren und einem System. Use Cases beschreiben das extern sichtbare Systemverhalten in der Sprache und aus der Sicht der Anwender. Mit einem Use Case wird durch das System ein für den Anwender sinnvoller Dienst erbracht oder ein benutzbares Ergebnis erreicht.

Spezifikation von Anforderungen

Use Case in natürlicher Sprache

Beispiel „Geld abheben“

„Das System zeigt die verfügbaren Konten an, von denen Geld abgehoben werden kann. Der Nutzer wählt ein Konto aus. Das System fragt nach dem Betrag, der ausgezahlt werden soll und der Nutzer gibt diesen an. Das System bucht den Betrag vom Konto ab und gibt das Geld aus. Anschließend gibt das System die Bankkarte aus. Wenn der Nutzer die Bankkarte entnommen hat, zeigt das System eine Willkommensnachricht.“

Use Case mit Schablone

Identifikator		
Name		
Zusammenfassung		
Priorität		
Vorbedingungen		
Nachbedingungen		
Hauptakteur		
Nebenakteur		
Auslöser		
Standardablauf	Schritt	Aktion
Erweiterungen	Schritt	Verzweigung/...
Offene Fragen		

Ein Anwender, der das System benutzt, um ein Ziel zu erreichen.

Weitere Akteure, die zur Bereitstellung des Dienstes für den Hauptakteur benötigt werden.

Es können beliebige weitere Felder ergänzt werden (z.B. Autor, Teilsystem und Stakeholder).

Beispiel 1 – System „Geldautomat“

Identifikator	/PF20/	
Name	<i>Geld abheben.</i>	
Zusammenfassung	<i>Ein Nutzer hebt am Terminal Geld von einem seiner Konten ab.</i>	
Priorität	<i>hoch</i>	
Vorbedingungen	<i>Der Nutzer hat seine Karte in das Terminal gesteckt.</i>	
Nachbedingungen	<i>Der Nutzer hat die gewünschte Menge Geld und seine Karte erhalten.</i>	
Hauptakteur	<i>Bankkunde</i>	
Standardablauf	Schritt	Aktion
	1	<i>System zeigt die Konten an.</i>
	2	<i>Nutzer wählt ein Konto aus.</i>
	3	<i>System fragt nach dem Betrag.</i>
	4	<i>Nutzer gibt den Betrag ein.</i>
	5	<i>System bucht den Betrag ab und gibt das Geld aus.</i>
	6	<i>Nutzer entnimmt das Geld.</i>

	7	<i>System gibt die Karte aus.</i>
	8	<i>Nutzer entnimmt die Karte.</i>
	9	<i>System zeigt Willkommenstext.</i>
Erweiterungen	Schritt	Verzweigung/Aktion
	5.a	Der Betrag ist auf dem Konto nicht gedeckt.
	5.a.1	<i>Das System teilt dem Nutzer mit, dass das Konto nicht gedeckt ist.</i>
	5.a.2	<i>Schritte 7 - 9</i>
Offene Fragen		Sollte das System den aktualisierten Kontostand anzeigen?

Beispiel 2 – System „Videothek“

Identifikator	<i>/PF40/</i>
Name	<i>Video ausleihen</i>
Zusammenfassung	<i>Ein Nutzer leiht sich in der Videothek ein Video aus.</i>
Priorität	<i>hoch</i>
Vorbedingungen	<i>Der Mitarbeiter ist am System angemeldet und hat die Aktion „Video ausleihen“ aufgerufen.</i>
Nachbedingungen	Bei Erfolg: <i>Das Video wurde an den Kunden verliehen, die Transaktion wurde abgespeichert und das Video wurde in der Bestandsliste als ausgeliehen gekennzeichnet.</i> Im Fall eines Fehlers: <i>Das Video wurde nicht verliehen und die Bestandsliste bleibt unverändert.</i>
Hauptakteur	<i>Mitarbeiter der Videothek</i>
Nebenakteur	Kunde

Standardablauf	Schritt	Aktion
	1	<i>Kunde bringt das gewählte Video an die Theke.</i>
	2	<i>Mitarbeiter bittet den Kunden um die Mitgliedskarte.</i>
	3	<i>Kunde übergibt dem Mitarbeiter die Mitgliedskarte.</i>
	4	<i>Mitarbeiter lässt die Mitgliedskarte vom System einlesen.</i>
	5	<i>System zeigt die Kundendaten an.</i>
	6	<i>Mitarbeiter trägt die Identifikationsnummer des Videos in das System ein.</i>
	7	<i>System zeigt die Details und die Ausleihgebühr des Videos an.</i>
	8	<i>Mitarbeiter wählt die Aktion „Ausleihen“.</i>
	9	<i>System fordert Mitarbeiter zur Eingabe des bezahlten Geldbetrags auf.</i>
	10	<i>Mitarbeiter teilt dem Kunden die Ausleihgebühr mit.</i>

	11	<i>Kunde übergibt dem Mitarbeiter einen Geldbetrag.</i>
	12	<i>Mitarbeiter gibt den bezahlten Betrag in das System ein.</i>
	13	<i>System zeigt das Rückgeld für den Kunden an.</i>
	14	<i>Mitarbeiter übergibt dem Kunden das Rückgeld und das ausgeliehene Video und bestätigt den Erfolg der Transaktion im System.</i>
	15	<i>System markiert das Video in der Bestandsliste als ausgeliehen und zeigt eine Bestätigungsnachricht an (Erfolg).</i>
Erweiterungen	Schritt	Verzweigung/Aktion
	3.a	<i>Kunde hat keine Mitgliedskarte</i>
	3.a.1	<i>Mitarbeiter registriert den Kunden (Use Case XY)</i>
	3.a.2	<i>Weiter ab Schritt 3.</i>
	11.a	<i>Kunde hat nicht genügend Geld.</i>
	11.a.1	<i>Mitarbeiter bricht die Transaktion am System ab (Fehler).</i>

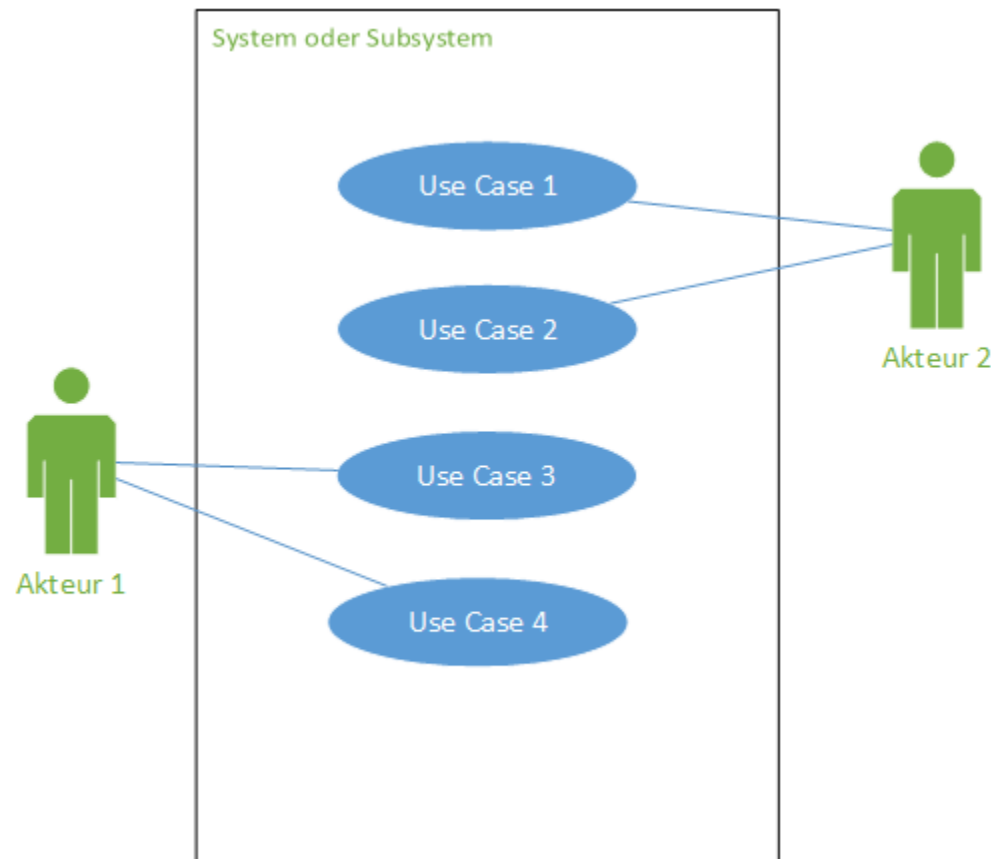
Beispiel 3 - Textverarbeitungssoftware

Identifikator	<i>/PF30/</i>
Name	<i>Suchen und Ersetzen</i>
Zusammenfassung	<i>Jedes Vorkommen eines bestimmten Textes im Dokument wird durch einen neuen Text ersetzt.</i>
Priorität	<i>Mittel</i>
Vorbedingungen	<i>Ein Dokument ist geöffnet und kann bearbeitet werden.</i>
Hauptakteur	<i>Nutzer</i>

Standardablauf	Schritt	Aktion
	1	<i>Nutzer wählt die Aktion „Suchen und Ersetzen“</i>
	2	<i>Software fordert den Nutzer auf, den Suchtext und den Ersetzungstext einzugeben.</i>
	3	<i>Nutzer gibt den Such- und den Ersetzungstext ein und gibt an, dass alle Vorkommen ersetzt werden sollen.</i>
	4	<i>Software ersetzt alle Vorkommen des Suchtextes durch den Ersetzungstext.</i>
Erweiterungen	Schritt	Verzweigung/Aktion
	3.a	<i>Nutzer gibt an, dass nur das erste Vorkommen ersetzt werden soll.</i>
	3.a.1	<i>Software ersetzt das erste Vorkommen des Suchtextes durch den Ersetzungstext und hebt die restlichen Vorkommen des Suchtextes hervor.</i>
	3.b	<i>Nutzer gibt an, dass nur gesucht, und nicht ersetzt werden soll.</i>
	3.b.1	<i>Software hebt alle Vorkommen des Suchtextes hervor.</i>

Spezifikation von Anforderungen

Use-Case-Diagramme

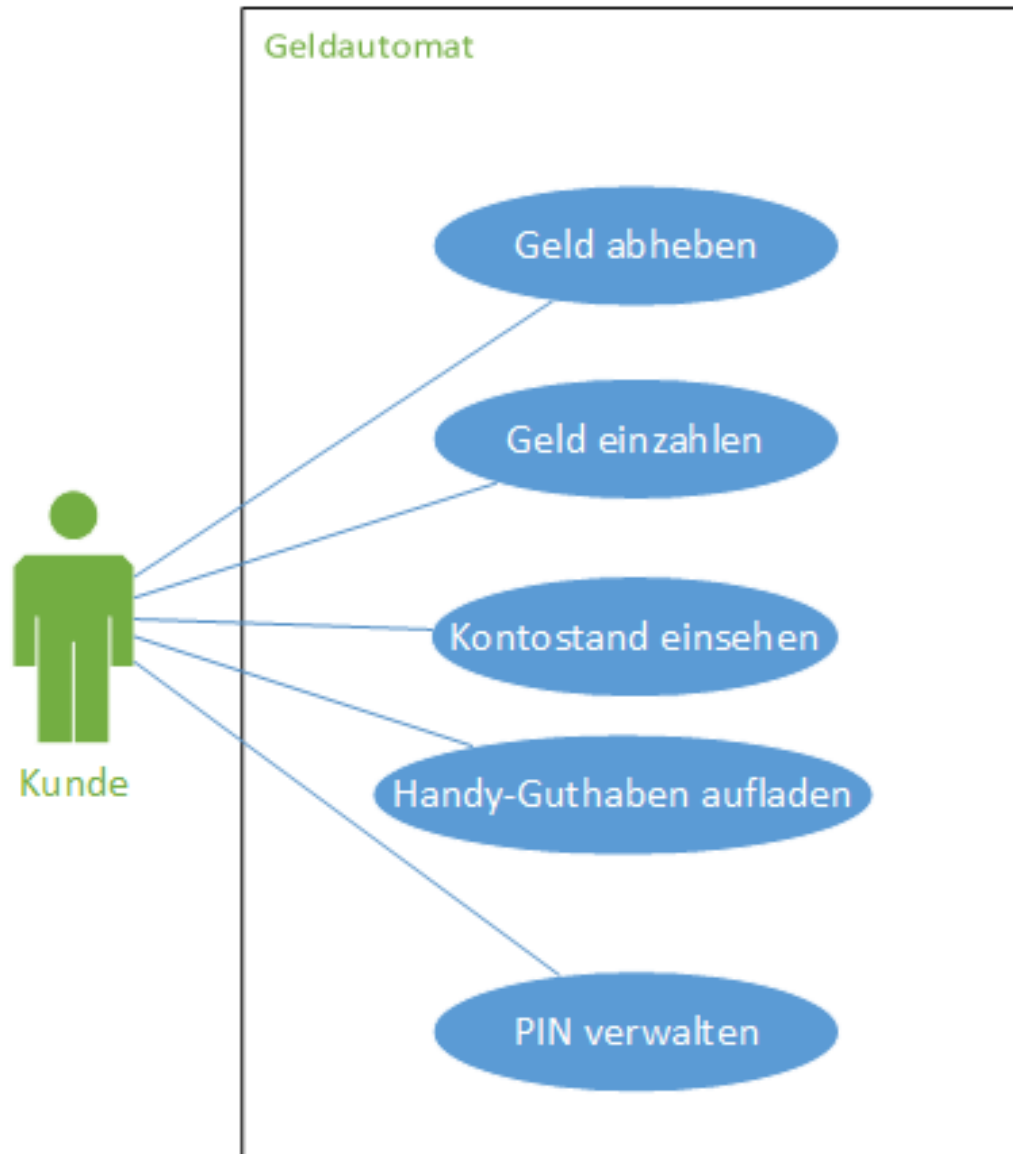


Spezifikation von Anforderungen

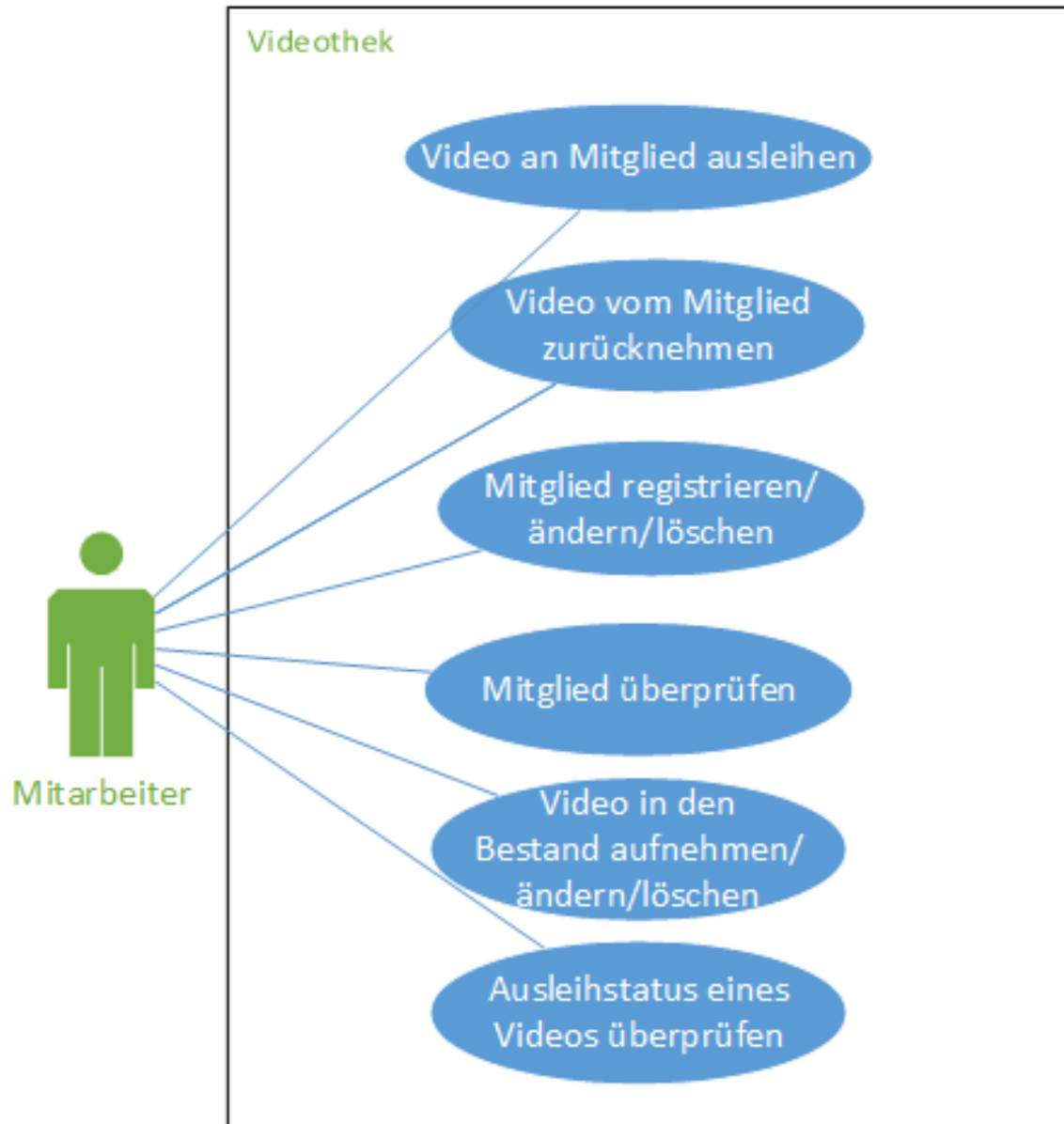
Use-Case-Diagramme

- ☞ Übersicht über Akteure und die zugehörigen Use Cases
- ☞ Übersicht über Teilsysteme und deren Use Cases
- ☞ erleichtert Überprüfung der Vollständigkeit

Use-Case-Diagramm „Geldautomat“



Use-Case-Diagramm „Videothek“



Spezifikation von Anforderungen

Arten von Use Cases

Business Use Cases

- ∞ beschreiben Geschäftsprozesse in oder zwischen Unternehmen/Organisationen
- ∞ können sich über einen großen Zeitraum erstrecken (z.B. Tage, Wochen)
- ∞ können von verschiedenen Akteuren mit unterschiedlichen Systemen ausgeführt werden
- ∞ Beispiele: „Auslandssemester organisieren“, „Student immatrikulieren“

Spezifikation von Anforderungen

Arten von Use Cases

Business Use Cases

Use Case „Lieferauftrag ausführen“

1. Kundendaten abrufen
2. Lieferbarkeit prüfen
3. Rechnung erstellen
4. Auftrag vom Lager ausführen lassen
5. Rechnungskopie an Buchhaltung geben

Spezifikation von Anforderungen

Arten von Use Cases

System Use Cases

- ✎ ein oder mehrere System Use Cases setzen einen Business Use Case um
- ✎ jeder System Use Case setzt ein Anwenderziel um
- ✎ Zweck: abstrakte Beschreibung der Systemschnittstelle

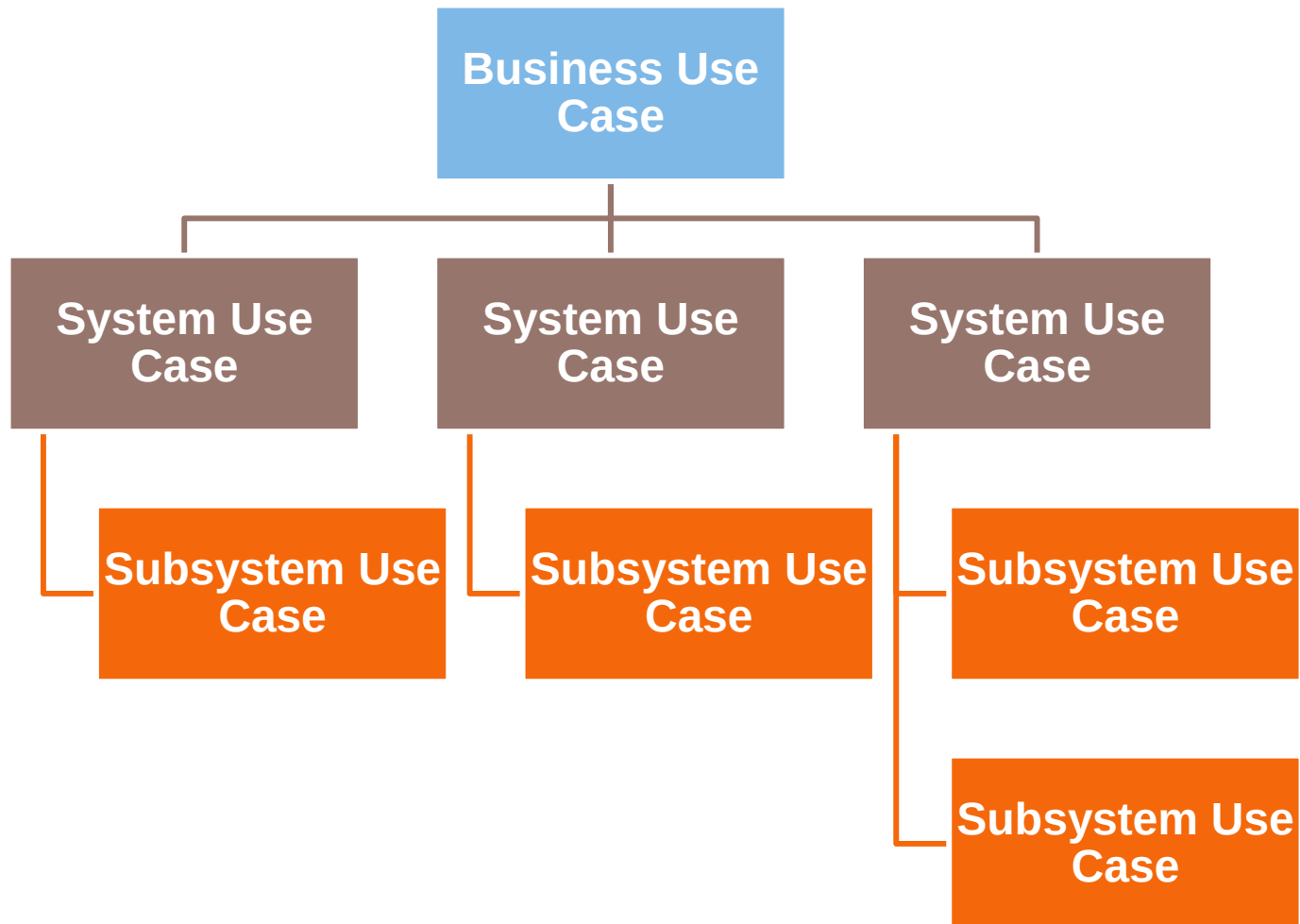
Spezifikation von Anforderungen

Arten von Use Cases

Subsystem Use Cases

- ⌘ auch Unterfunktion genannt
- ⌘ Komplexe Ablaufschritte in System Use Cases können in eigenständige Subsystem Use Cases ausgelagert werden.
- ⌘ Subsystem Use Cases werden oft von mehreren System Use Cases genutzt (z.B. „Login“).

Spezifikation von Anforderungen



Spezifikation von Anforderungen

Worauf Sie achten sollten

- ✎ Details der Benutzerschnittstelle aussparen → GUI wird später von Usability-Experten entworfen
- ✎ nicht zu viele Details im Use Case angeben → auf Lesbarkeit achten
- ✎ richtige Granularität finden → Aufteilung in System- und Subsystem Use Cases

Falsch – Lösungsabhängig:

1. Der Akteur klickt links in der Navigation auf “Vertrag anlegen”.
2. Das System zeigt die Erfassungsmaske “Neuen Vertrag anlegen”.
3. Der Akteur gibt die Daten ein und klickt auf OK.
4. Das System speichert die Daten.
5. Das System zeigt die Vertragsübersicht an.

Besser – Lösungsunabhängig:

1. Der Akteur ruft die Funktion zur Anlage eines Vertrags auf.
2. Das System bietet die Möglichkeit, Vertragsdaten einzugeben.
3. Der Akteur gibt die Vertragsdaten ein und bestätigt die Eingabe.
4. Das System legt den Vertrag mit den angegebenen Daten an.
5. Das System bietet eine Zusammenfassung der erfassten Daten.

1. Der Akteur gibt den Namen ein.</>
2. Der Akteur gibt den Nachnamen ein.</>
3. Der Akteur gibt das Geburtsdatum ein.</>
4. ...

1. Der Akteur gibt die Kunden- und Vertragsdaten ein.

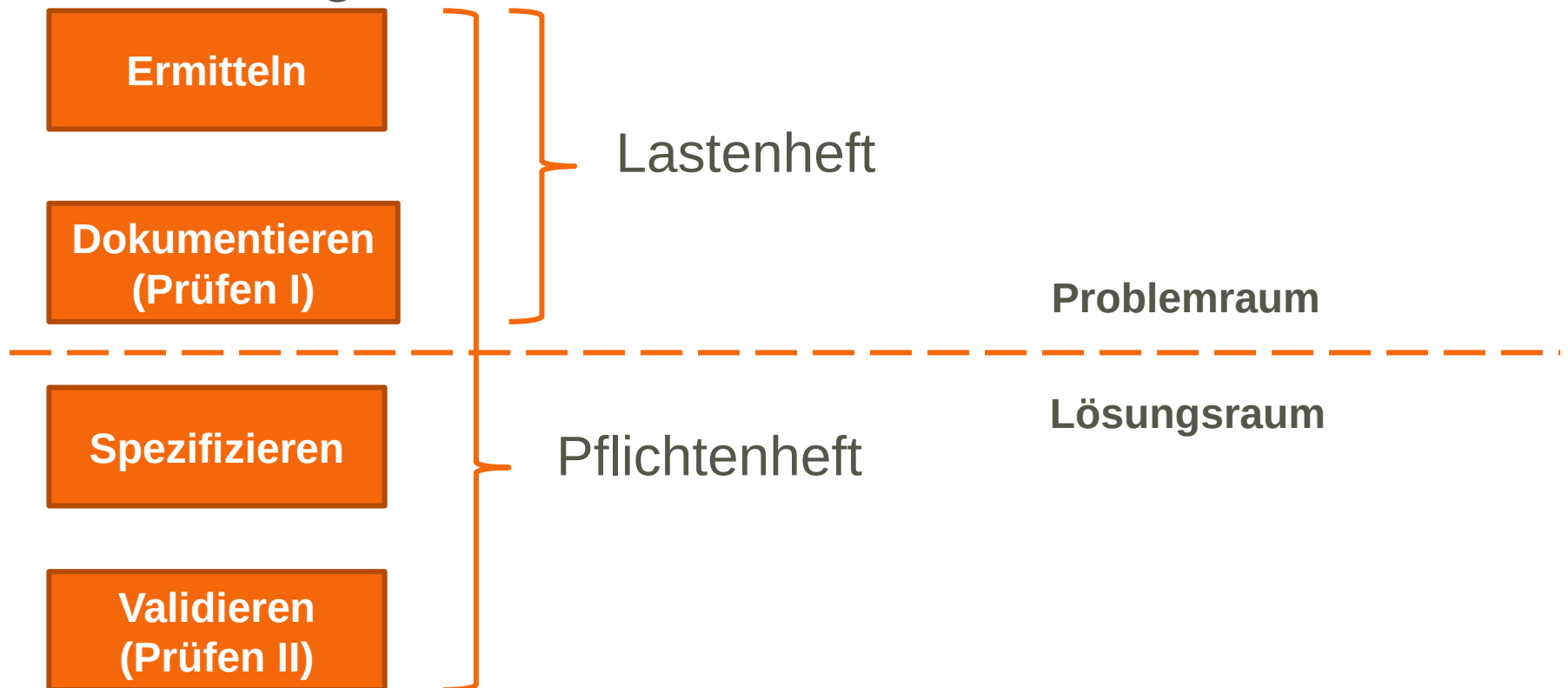
Vertiefung

Was wir vertiefen wollen

- ✎ Unterschiedliche Erscheinungsformen funktionaler Anforderungen je nachdem,
 - ob sie im Lastenheft oder Pflichtenheft dokumentiert sind
 - in welcher RE-Stufe sie dokumentiert werden (Problemraum oder Lösungsraum)

Vertiefung

RE als weiche Übergangsphase zwischen Problem- und Lösungsraum



Vertiefung

Zeiterfasser	Lastenheft	Pflichtenheft
Problem- raum	„Das System sollte eine Übersicht über die Zeitbuchungen bieten.“	„Das System sollte eine Übersicht über alle Zeitbuchungen an einem frei wählbaren Tag und für ein frei wählbares Projekt bieten.“
Lösungs- raum	—	Use Case 1. Nutzer wählt ein Projekt aus. 2. Nutzer wählt einen Tag aus. 3. System zeigt alle Buchungen für diesen Tag an, die zu dem gewählten Projekt gehören.

Vertiefung

Zeichensoftware	Lastenheft	Pflichtenheft
Problemraum	„Das System sollte dem Nutzer die Möglichkeit bieten, Kreise zu zeichnen.“	„Das System sollte in der Lage sein, einen Kreis zu zeichnen, der vom Nutzer durch die Angabe von Mittelpunkt und Radius definiert wird.“
Lösungsraum		Use Case 1. Nutzer wählt die Aktion „Kreis“. 2. Nutzer legt den Mittelpunkt des Kreises an einem beliebigen Punkt in der Arbeitsfläche fest. 3. Nutzer legt den Radius des Kreises fest. 4. System zeichnet einen Kreis mit dem festgelegten Mittelpunkt und Radius.

Vertiefung

Zeichensoftware	Lastenheft	Pflichtenheft
Problemraum	„Das System sollte dem Nutzer die Möglichkeit bieten, Kreise zu zeichnen.“	„Das System soll die Möglichkeit bieten, Kreise zu zeichnen.“ Negatives Beispiel: „Der Mittelpunkt wird durch Klick auf die Arbeitsfläche festgelegt.“
Lösungsraum	—	Use Case 1. Nutzer wählt die Aktion „Kreis“. 2. Nutzer legt den Mittelpunkt des Kreises an einem beliebigen Punkt in der Arbeitsfläche fest. 3. Nutzer legt den Radius des Kreises fest. 4. System zeichnet einen Kreis mit dem festgelegten Mittelpunkt und Radius.

Vertiefung

Zeichensoftware	Lastenheft	Pflichtenheft
Problemraum	„Das System sollte dem Nutzer die Möglichkeit bieten, Kreise zu zeichnen.“	„Das System sollte in der Lage sein, einen Kreis zu zeichnen, der vom Nutzer durch die Angabe von Mittelpunkt und Radius definiert wird.“
Lösungsraum	—	Use Case 1. Nutzer wählt die Aktion „Kreis“. 2. System fordert den Nutzer zur Angabe eines Radius auf. 3. Nutzer gibt den Radius des Kreises an. 4. System zeichnet einen Kreis mit dem festgelegten Radius in die Mitte der Arbeitsfläche. 5. Nutzer ändert die Position des Kreises.

Vertiefung

Zeichensoftware	Lastenheft	Pflichtenheft
Problemraum	„Das System sollte dem Nutzer die Möglichkeit bieten, einen Kreis zu zeichnen.“	„Das System sollte dem Nutzer die Möglichkeit bieten, einen Kreis zu zeichnen.“
Lösungsraum	—	Use Case 1. Nutzer wählt die Aktion „Kreis“. 2. Nutzer legt die Position des Kreises an einem beliebigen Punkt in der Arbeitsfläche fest. 3. System zeichnet einen Kreis mit einem Standardradius. 4. Nutzer ändert ggf. die Größe und Position des Kreises.

Vertiefung

Formulierung von Anforderungen

- ⌘ Gefahr 1 – Passiv statt Aktiv (Folie 9)
- ⌘ Gefahr 2 – Substantive ohne Bezugsindex
 - **Beispiel:** „Die Daten sollen dem Nutzer auf dem Terminal angezeigt werden.“
 - **Besser:** „Das System zeigt dem registrierten Nutzer seine Rechnungsdaten auf dem Terminal an, an dem er angemeldet ist.“

Vertiefung

Formulierung von Anforderungen

⌘ Gefahr 3 – Unvollständig spezifizierte Bedingungen

- **Beispiel:** „Das Restaurantsystem soll einem registrierten Gast bei einem Alter von über 17 Jahren alle im Lokal angebotenen Getränke anzeigen.“
- **Problem:** Was passiert bei einem Alter von unter 16 Jahren und bei einem Alter von 16-17 Jahren?

Vertiefung

**Verbessern Sie folgende
Anforderung:**

„Zur Anmeldung des Nutzers werden
die Login-Daten eingegeben.“

Pflichtenheft

Mögliche Gliederung Pflichtenheft (DIN 69905)

1. Zielbestimmung

2. Produkteinsatz

3. **Produktübersicht**

4. **Technische Produktumgebung**

5. Produktfunktionen

6. Produktdaten

Anwendungs-
bereiche,
Zielgruppen,
Betriebsbedingungen

Use-Case-
Diagramm

Use-Cases

Pflichtenheft

- 7. Produktleistungen
- 8. **Benutzeroberfläche**
- 9. Nichtfunktionale Anforderungen
- 10. **Globale Testszenarien/Testfälle**
- 11. **Entwicklungsumgebung**
- 12. Ergänzungen

Pflichtenheft

Produkteinsatz

„Die [Software zur] Textverarbeitung soll im Redaktionsbetrieb einer Zeitung (Büro) eingesetzt werden. Sie wird von den Redakteuren bedient, die EDV-kundig sind, aber bislang nur DROW-3 benutzt haben. Das Produkt wird von einer EDV-Systemgruppe installiert und vollzeitbetreut.“

Produktumgebung

- ∞ *„Die Textverarbeitung soll von den Redakteuren auf Rechnern mit Intel-Prozessoren unter dem Betriebssystem OS/2 mit üblichem Umfeld: Maus, Keyboard, Farbmonitor in SVGA-Auflösung [eingesetzt werden].*
- ∞ *Die Ausgabe erfolgt zunächst auf HP5-kompatiblen Laserdruckern und dann auf einer Satzmaschine der Fa. Linotype mit den dort beschafften Zeichensätzen (in 1200dpi).*
- ∞ *Damit der Datentransport zur Satzmaschine sinnvoll erfolgen kann, müssen die Redakteursrechner über Ethernet mit der Satzmaschine verbunden sein.*
- ∞ *Die Software sollte Daten aus dem Office-Paket übernehmen können, und Bilder aus Adobe-Photoshop.“*

Produktfunktionen

- ∞ Use Cases mit Priorisierung
- ∞ Beispiel: Use Case für Silbentrennung
- ∞ Identifikator: /PF10/, /PF11/ usw.
- ∞ Referenz auf Lastenheft

Produktleistungen

„Bei der Silbentrennung sollte die Fehlerrate nicht über 0.5% liegen.“

Benutzeroberfläche

- ∞ *„Die Gesamtseite muß auf dem Monitor dargestellt werden können. Die Anweisungen zum Satz, Druck etc. sollen über Fall-Down-Menüs mit Windows96-Look&Feel gehandhabt werden.“*

Nicht funktionale Anforderungen

- ∞ *„Bei Störungen im Netzwerk dürfen keine Daten verlorengehen.“*

Globale Testfälle

- ∞ *Zum Test des verlangten Umgangs mit Netzstörungen wird ein Arbeitsplatz während einer Artikelerstellung vom Netz getrennt. Es dürfen keine Daten verlorengehen, und die Daten sollten transparent an einem anderen Arbeitsplatz weiterbearbeitet werden können.*

Entwicklungsumgebung

„Entwickelt wird unter und für das Betriebssystem Windows95 auf Rechnern der Intel-Pentium-Architektur. Es wird ein InterData-C-Compiler und die Internetsoftware Explorer-X.3 benutzt.“

Quelle: <http://cs.uni-muenster.de/u/lammers/EDU/PubArchiv/Scripten/PP-Script99/Pflichtenheft.html>

Pflichtenheft

Anwendung in Ihrer Projektaufgabe

- ☞ Spezifizieren Sie die ermittelten Anforderungen in einem Pflichtenheft!
- ☞ Stellen Sie Bezüge zu den Anforderungen im Lastenheft her (Nachverfolgbarkeit).
- ☞ Umfang: 10 - 15 Seiten und Use Cases

Female Software Developer



what my mom thinks I
do



what other moms think
I do



what my spouse & kids
think I do



what the movies think
I do



what male developers
think I do



what I really do

Software Developers' Friends



Friends that friends
think I have



Friends my mom thinks
I have



Friends society thinks
I have



Friends my colleagues
think I have



Friends I think I have



Friends I really have

powered by uthinkido.com 

Software Engineer



What my friends think I do.



What my mom thinks I do.



What society thinks I do.



What my boss thinks I do.



What I think I do.



What I actually do.