

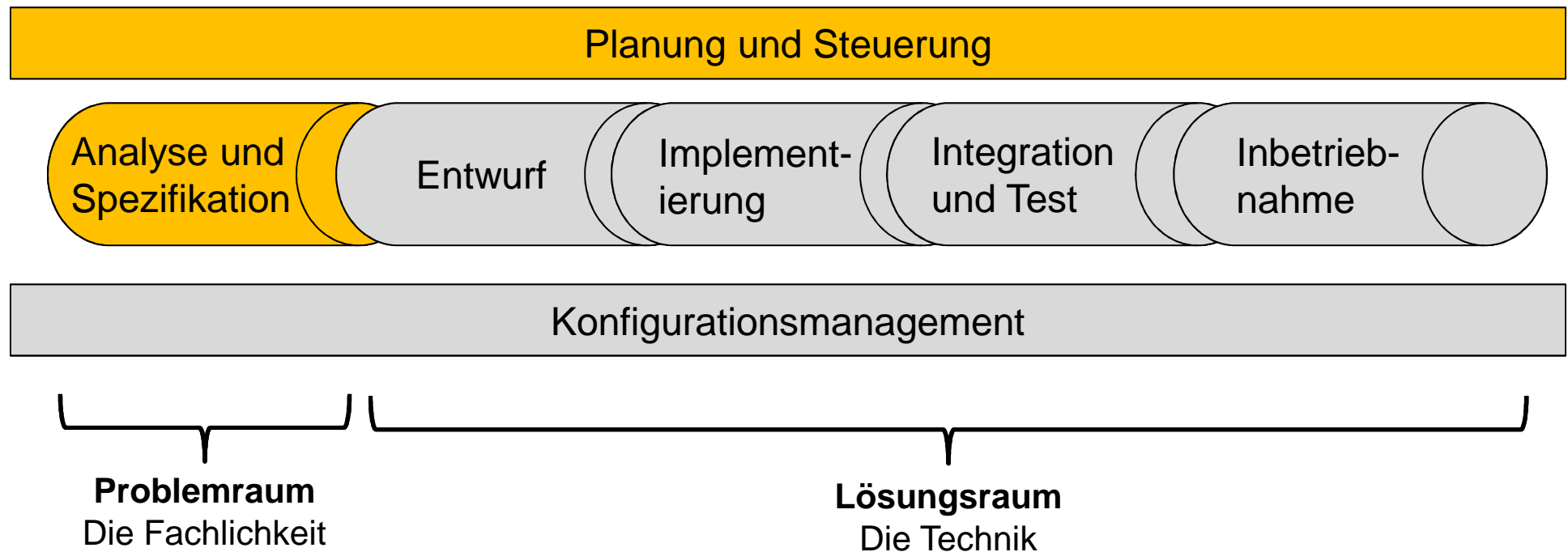
Software Engineering II

Spezifikation und Planung

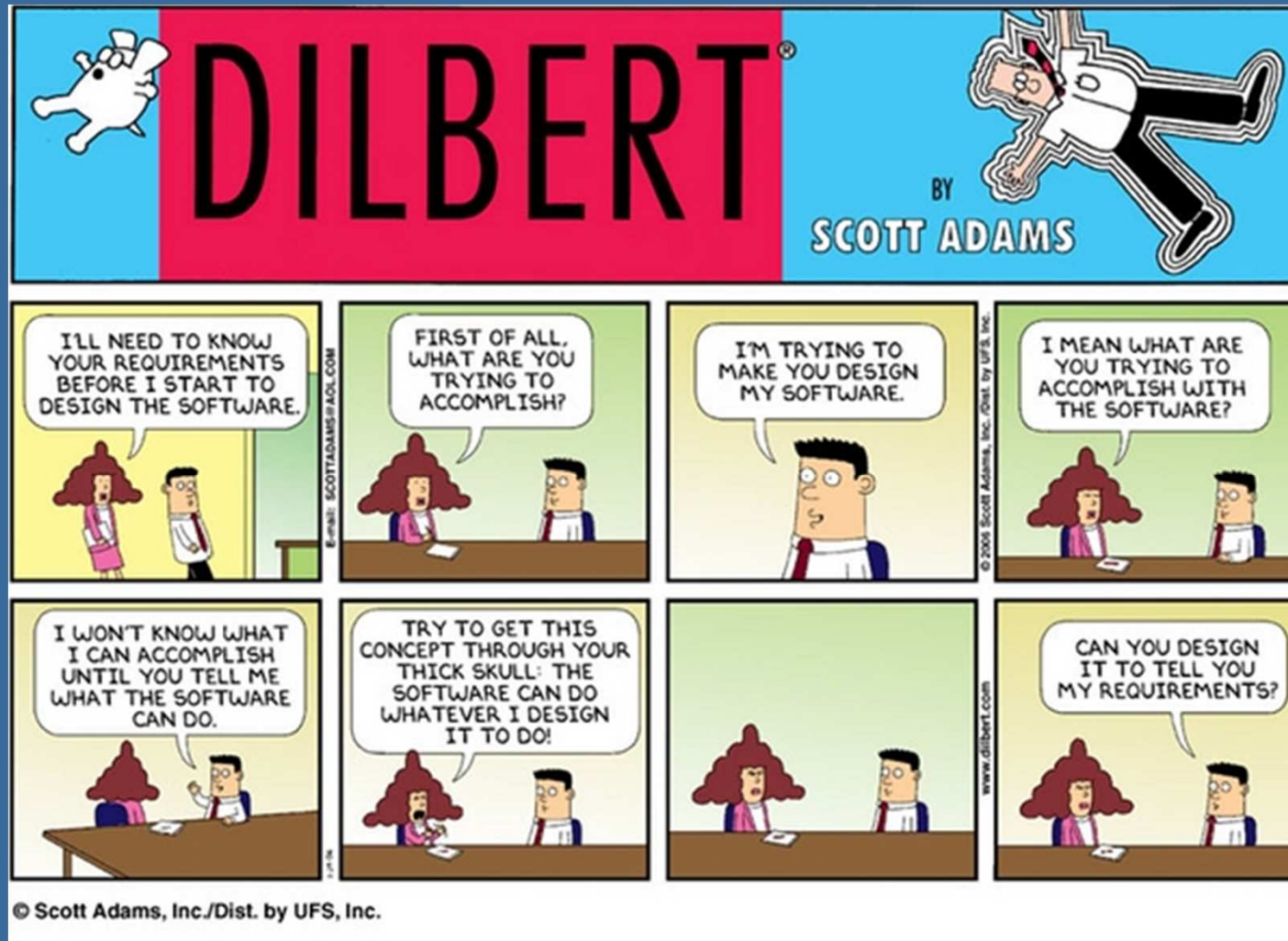
*The hardest single part of building a software system is deciding precisely what to build.
No other part of the conceptual work is as difficult as establishing the [...] requirements [...]
No other part of the work so cripples the resulting system if done wrong.
No other part is as difficult to rectify later.*
Fred Brooks, 1987

Montag, 8. April 2013
Dr. Josef Adersberger
Version 1.0

Wir sind heute auf unserer Landkarte des Software Engineering bei Spezifikation und Planung.



Analyse und Spezifikation



Das Leben ist hart: Die Axiome der Spezifikation.

Ziv's Law:

- Specifications will never be fully understood

Humphrey's Law:

- The user will never be sure of what they want until they see the system in production (if then)

Wegner's Lemma:

- An interactive system can never be fully specified, nor can it ever be fully tested

Langdon's Lemma:

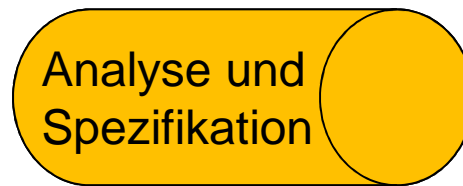
- Software evolves more rapidly as it approaches chaotic regions (without spilling into chaos)

Was ist Analyse?

- Analyse ist die Frage nach dem Was.
- Analyse ist, den Ist-Zustand zu verstehen und den Soll-Zustand zu beschreiben. Sie ist eine notwendige Vorarbeit für die Spezifikation.
- Analyse ist, zu ...
 - ... untersuchen.
 - ... erfragen.
 - ... erlernen.
 - ... verstehen.
 - ... erproben.
- Analyse ist: „The process of studying user needs to arrive at a definition of system, hardware, or software requirements.” (IEEE Std 610.12-1990)
- Typisches Ergebnisse: Dokumentierte Anforderungen, Big Pictures, Protokolle, Prototypen und Mockups.

Was ist Spezifikation?

- Spezifikation ist die Frage nach dem Wie.
- Spezifikation ist, eine präzise Vorstellung vom Problem zu bieten, das gelöst werden soll.
- Spezifikation ist, ...
 - ... konsistent zu machen.
 - ... zu beschreiben.
 - ... zu konstruieren.
 - ... zu überprüfen.
 - ... prüfbar zu machen.
 - ... umsetzbar zu machen.
- Typisches Ergebnis: Fachkonzept als Eingabe für den Entwurf, die Implementierung und den Test. Ein Fachkonzept beschreibt ...
 - die Anforderungen an das System sowie die geltenden Annahmen, Abgrenzungen und Prämissen
 - die Daten im System
 - die Funktionen im System
 - die Benutzerschnittstelle
 - die Schnittstellen zu anderen Systemen



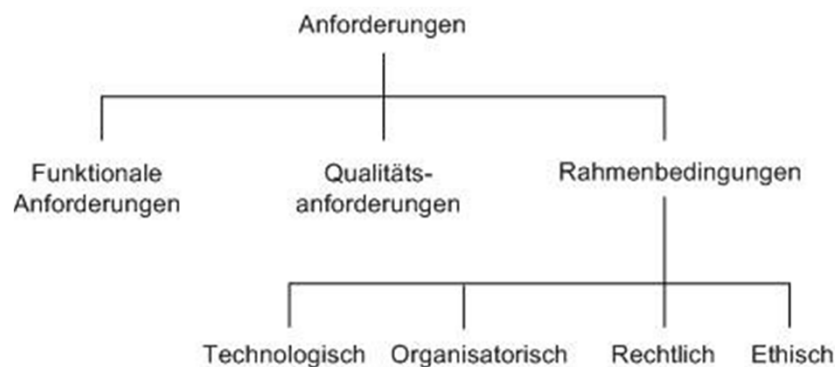
Methoden der Analyse

Methode: Strukturierte Erhebung

■ Eingabe: Strukturierte Interviews.

- Was ist das Problem?
- Was ist der Lösungsvorschlag?
- Wer will es?
- Was passiert, wenn man es nicht tut?
- Haben alle Betroffenen zugestimmt?

■ Ausgabe: Anforderungsstruktur.



Anforderungsschubladen (Typen)

Volere Requirements Shell, James & Suzanne Robertson,
<http://volere.co.uk>

Requirement #: 110	Requirement Type: 11	Event/Use Case #: 6
Description: The product shall be easy for the road engineers to use.		
Rationale: It should not be necessary for the engineers to attend training classes to be able to use the product.		
Originator: Sonia Henning, Road Engineering Supervisor		
Fit Criterion: A road engineer shall be able to use the product to successfully carry out the cited use cases within 1 hour of first encountering the product.		
Customer Satisfaction: 3	Customer Dissatisfaction: 5	
Priority: Next release	Conflicts: None	
Supporting Materials:		
History: Raised by AG. 25 Aug 05.		
		Volere <small>Copyright © Atlantic Systems Guild</small>

Anforderungskarteikarte

Methode: Sprachschablonen

User Stories (Scrum):

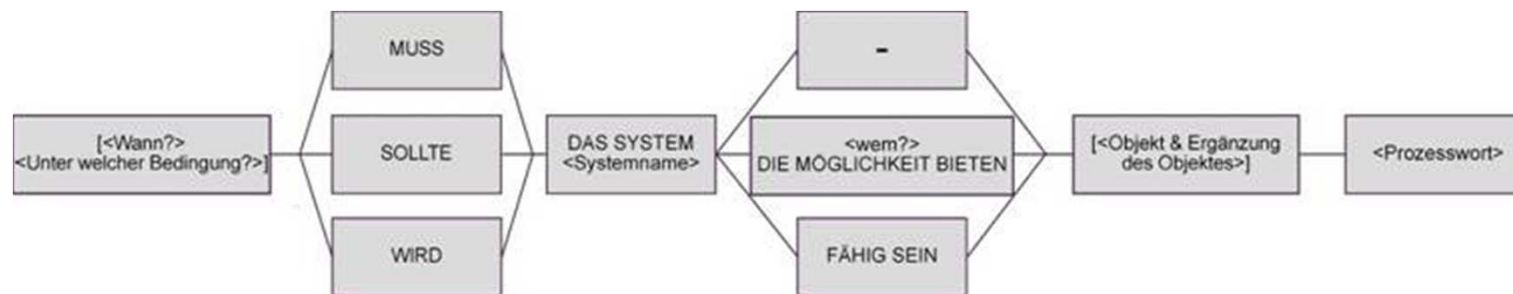
Front of card

Story		
Narrative	<small>(short name)</small>	Priority ____
As a	As a <user> <small>(role)</small>	Size ____
I want	I want to <do something> <small>(something)</small>	
So that	so that <I can accomplish goal>. <small>(goal of user)</small>	

Back of card

Acceptance	
Criteria	<small>(short name)</small>
Given	Given <input and precondition> <small>(condition)</small>
When	when <something happens with the system> <small>(event)</small>
Then	then <this is true> <small>(result)</small>

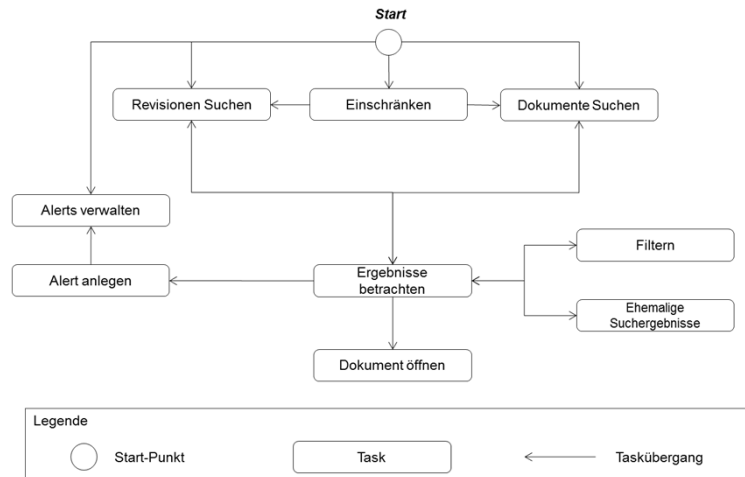
Anforderungsschema (SOPHIST):



„Nachdem das System die Geschäftsdaten gespeichert hat, muss das System dem Mitarbeiter die Möglichkeit bieten, einen Genehmigungsantrag auf dem Netzwerkdrucker zu drucken.“

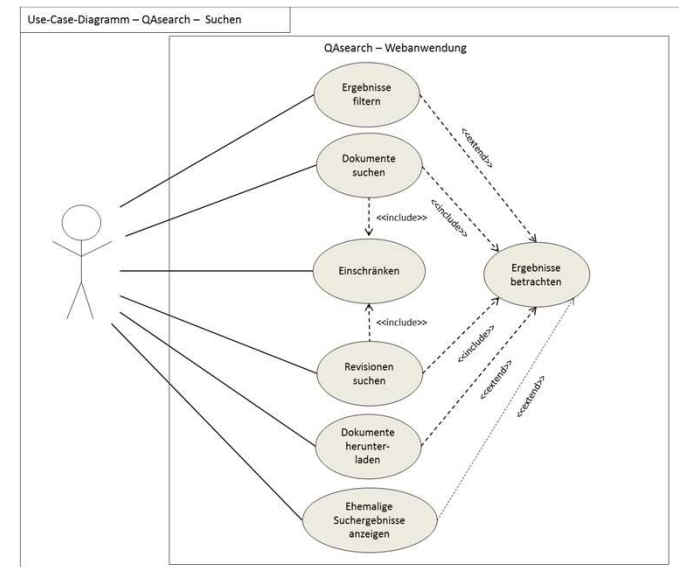
Methode: Funktionsmodelle

Task Modelle (Interaction Design):



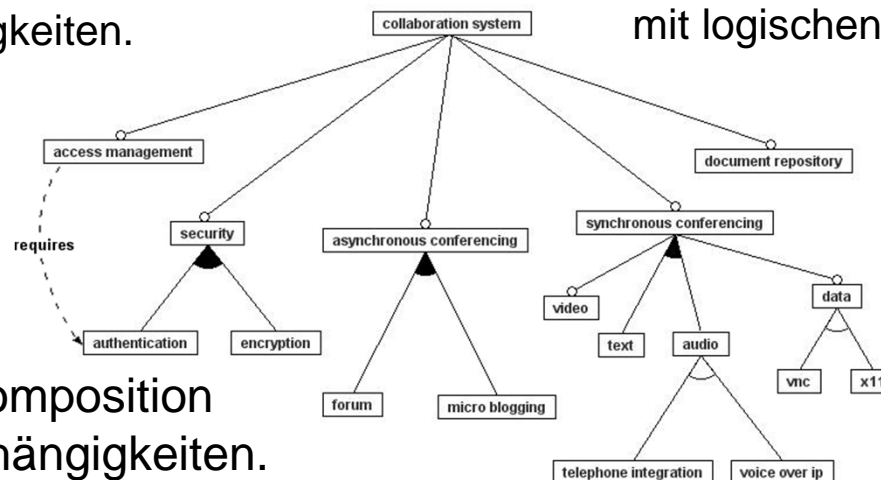
Funktionsabläufe für Personas mit temporalen Abhängigkeiten.

Use Case Modelle (UML):



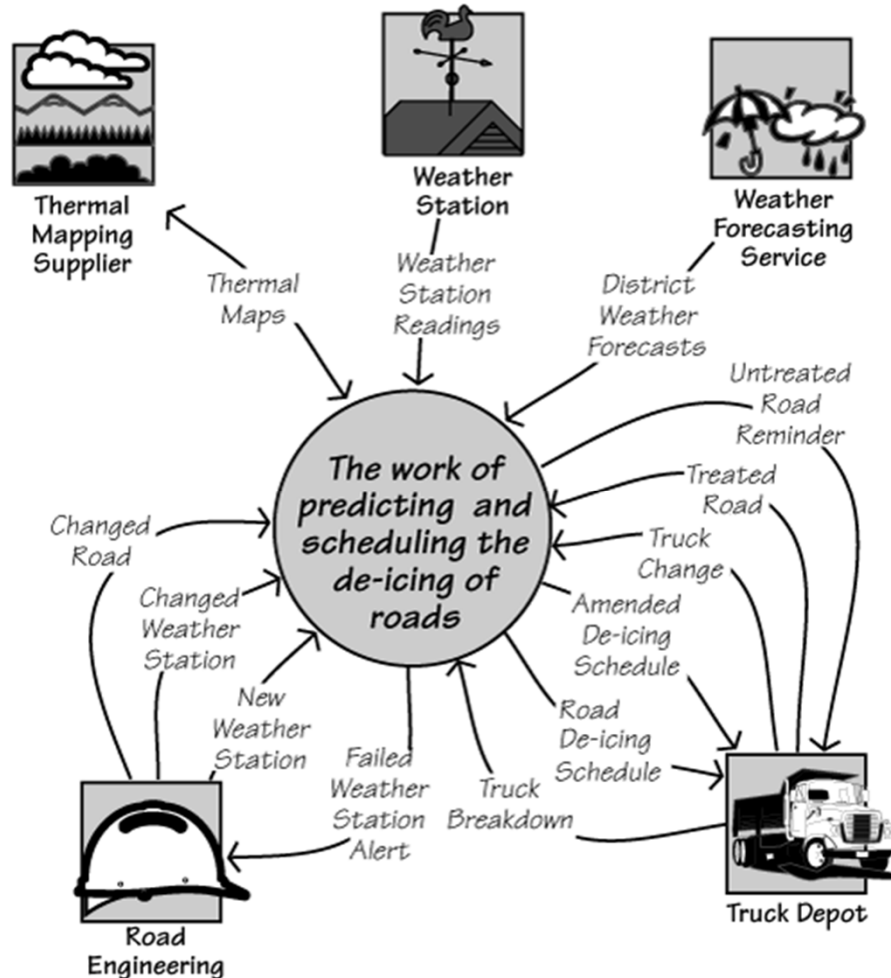
Funktionen für Benutzerrollen mit logischen Abhängigkeiten.

Feature Modelle (PLE):



Funktionale Dekomposition mit logischen Abhängigkeiten.

Methode: Kontextdiagramm



Ein Kontextdiagramm zeigt die Wechselwirkungen des Systems mit anderen Systemen. Insbesondere wird hierbei der Nachrichtenfluss betrachtet.

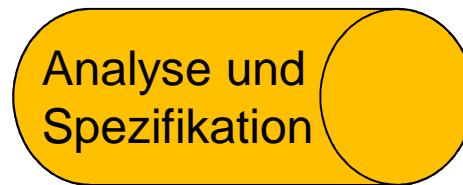
Das System wird als Blackbox betrachtet.

aus: Mastering the Requirements Process - Getting Requirements Right
by Suzanne Robertson and James Robertson

Methode: Glossar

Stichwort	Erläuterung	Synonym	Autor
Abholtermin	Zeitpunkt der Abholung von Ware von einem Lieferer durch einen Transporteur.		KT, FB
Abholung	Übernahme der Güter durch den Frachtführer. S.a. Verladung.		VDA
Abholzeitfenster	Der Zeitraum zwischen frühestem und spätestem Zeitpunkt der Abholung von Ware von einem Lieferer durch einen Transporteur.		KT, FB
Abladestelle	Stelle, an die die Güter bei einem Empfänger/Werk auszuliefern sind, Schlüsselbegriff für z.B. Rampe/Tor.		VDA

- Ein Glossar definiert alle wichtigen fachlichen Begriffe im Problemraum. Dabei ist auf eine gründliche, eindeutige, konsistente und vollständige Definition zu achten.
- Das Glossar wird im weiteren Projektverlauf ständig erweitert.
- In der Praxis hat sich herausgestellt, dass ein gutes Glossar erfolgskritisch ist.



Methoden der Spezifikation

Methode: Spezifikationsbausteine

Spezifikationsbausteine sind Vorlagen für Spezifikationsdokumente. Es können einzelne Bausteine oder sogar die komplette Vorlage übernommen werden.

PROJECT DRIVERS

1. The Purpose of the Project
2. Client, Customer, other Stakeholders
3. Users of the Product

PROJECT CONSTRAINTS

4. Mandated Constraints
5. Naming Conventions and Definitions
6. Relevant Facts and Assumptions

FUNCTIONAL REQUIREMENTS

7. The Scope of the Work
8. The Scope of the Product
9. Functional and Data Requirements

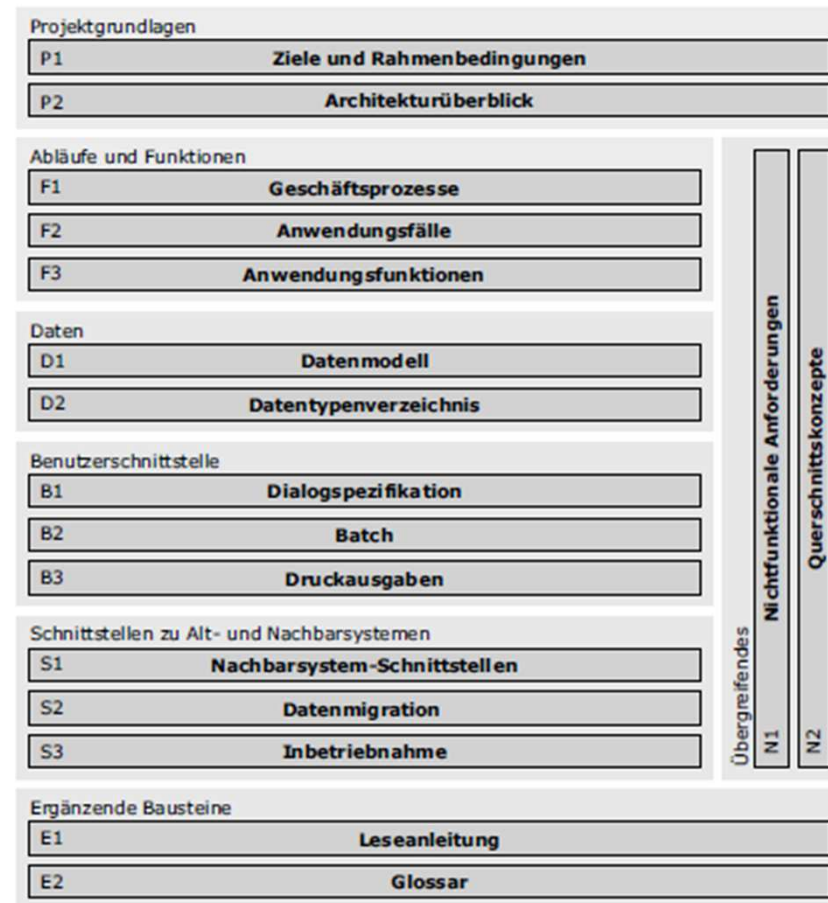
NON-FUNCTIONAL REQUIREMENTS

10. Look and Feel Requirements
11. Usability Requirements
12. Performance Requirements
13. Operational Requirements
14. Maintainability Requirements
15. Security Requirements
16. Cultural and Political Requirements
17. Legal Requirements

PROJECT ISSUES

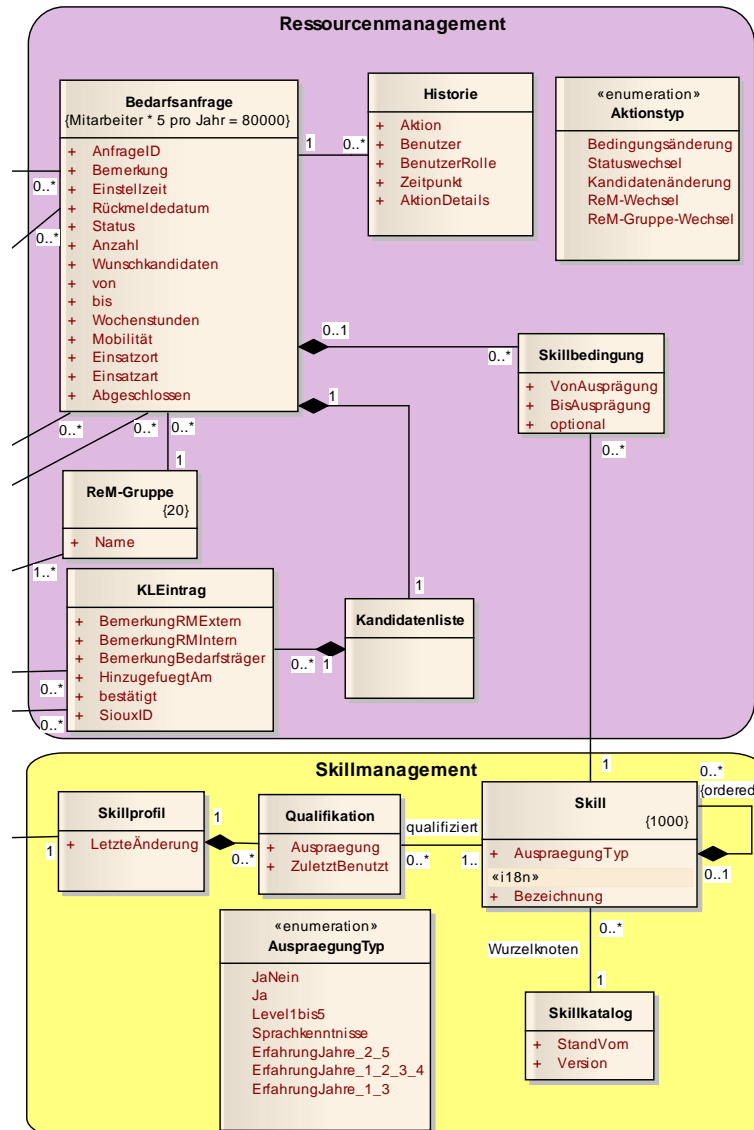
18. Open Issues
19. Off-the-Shelf Solutions
20. New Problems
21. Tasks
22. Cutover
23. Risks
24. Costs
25. User Documentation and Training
26. Waiting Room
27. Ideas for Solutions

Volere Template, James & Suzanne Robertson,
<http://volere.co.uk>



Softwaretechnik: Praxiswissen für Softwareingenieure,
 Johannes Siedersleben, 2002, Carl Hanser Verlag

Methode: Konzeptionelles Datenmodell



Ein konzeptionelles Datenmodell zeigt die fachlichen Einheiten (Entitäten) der Software, ihr Mengengerüst, ihre Attribute sowie ihre Assoziationen zueinander. UML Klassenmodelle sind ein geeignetes Diagrammformat dafür.

- Das konzeptionelle Datenmodell zeigt rein die fachliche Struktur der Daten. Es ist weder ein OO-Diagramm noch ein Datenbank-Diagramm.
- Das konzeptionelle Datenmodell kann auch bereits bei der Analyse eingesetzt werden.

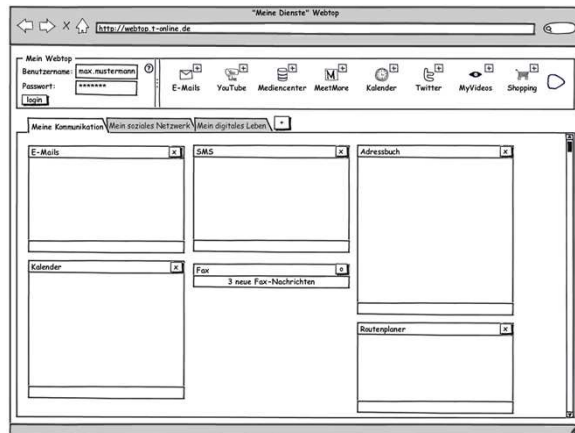
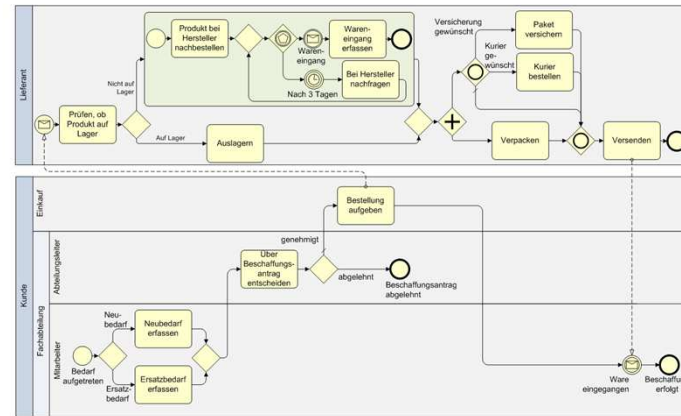
Methode: Anwendungsfallschablone

Mit einer Anwendungsfallschablone können identifizierte Anwendungsfälle (Use Cases) spezifiziert werden. So kann aus einem UML Use Case Diagramm eine Spezifikation werden.

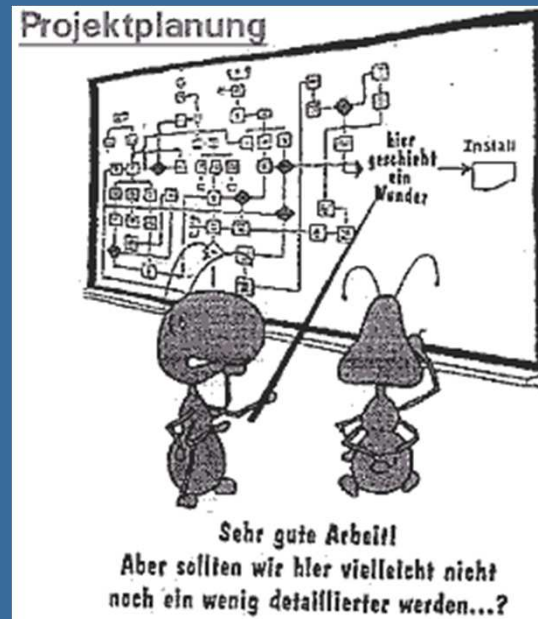
Name	Authentifizieren
Ziel	Der Kunde möchte Zugang zu einem Bankautomaten BA42 erhalten
Vorbedingung	<ul style="list-style-type: none">– Der Automat ist in Betrieb, die Willkommen-Botschaft wird angezeigt– Karte und PIN des Kunden sind verfügbar
Nachbedingung	<ul style="list-style-type: none">– Der Kunde wurde akzeptiert– Die Leistungen des BA42 stehen dem Kunden zur Verfügung
Nachbedingung im Sonderfall	Der Zugang wird verweigert, die Karte wird entweder zurückgegeben oder einbehalten, die Willkommen-Botschaft wird angezeigt
Akteure	Kunde (Hauptakteur), Banksystem

Weitere Methoden, die wir nicht behandelt haben, die aber trotzdem sehr hilfreich sind:

- Geschäftsprozessmodellierung
- Reviews
- Inspektionen
- Mockups
- ...

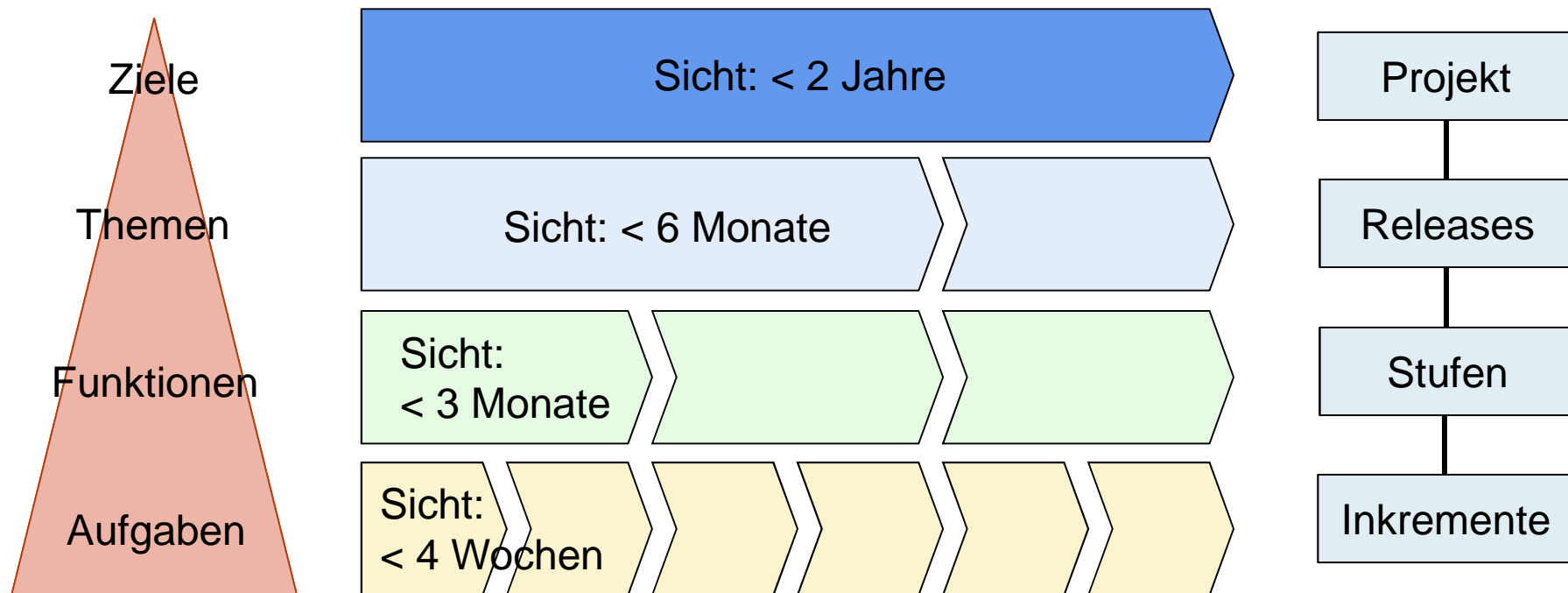


Planung



Die Planung von Software-Projekten muss auf mehreren Ebenen erfolgen.

Nicht-triviale Projekte müssen unabdingbar auf mehreren Ebenen und damit auf unterschiedlichen Sichtweiten und Granularitätsstufen geplant werden.





Das Produkt/Mission-Statement:

■ Produkt-Vision:

- QAsuch ist eine Such-Webanwendung, die alle SVN-Repositorys mitsamt der SVN-Historie für die Mitarbeiter der Firma QAware durchsucht.

■ Was ist neu:

- Es werden alle Revisionen der SVN-Repositorys der Firma QAware durchsucht.

■ Was ist das Produkt nicht:

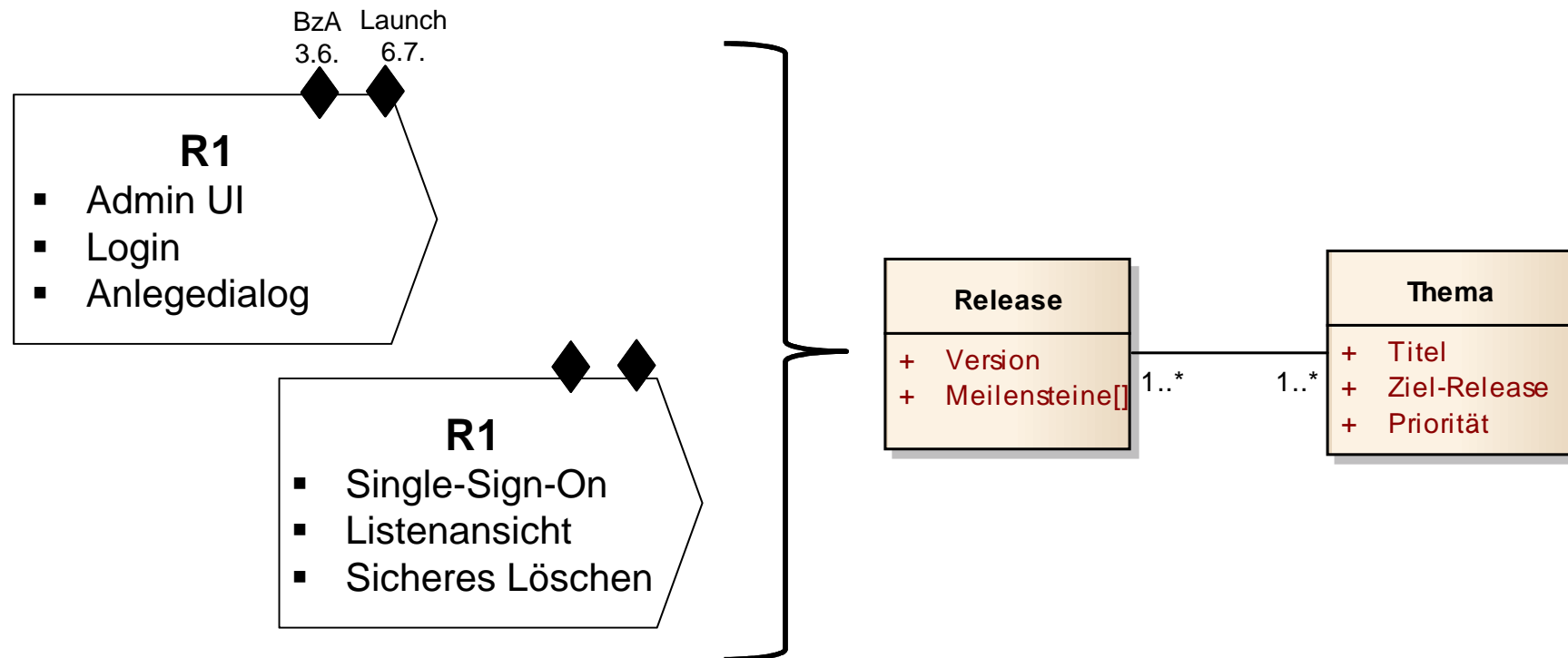
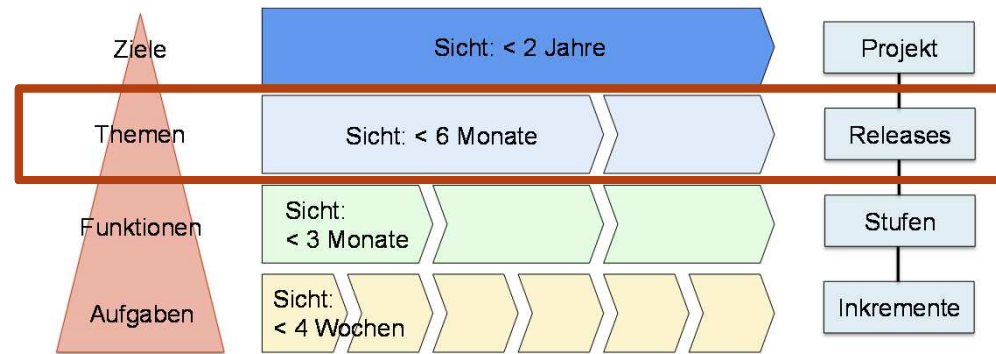
- Es werden nur die auf den Servern gespeicherten SVN-Repositorys durchsucht.
- Die Suchergebnisse werden nicht lokal auf den Rechnern des Nutzers gespeichert.

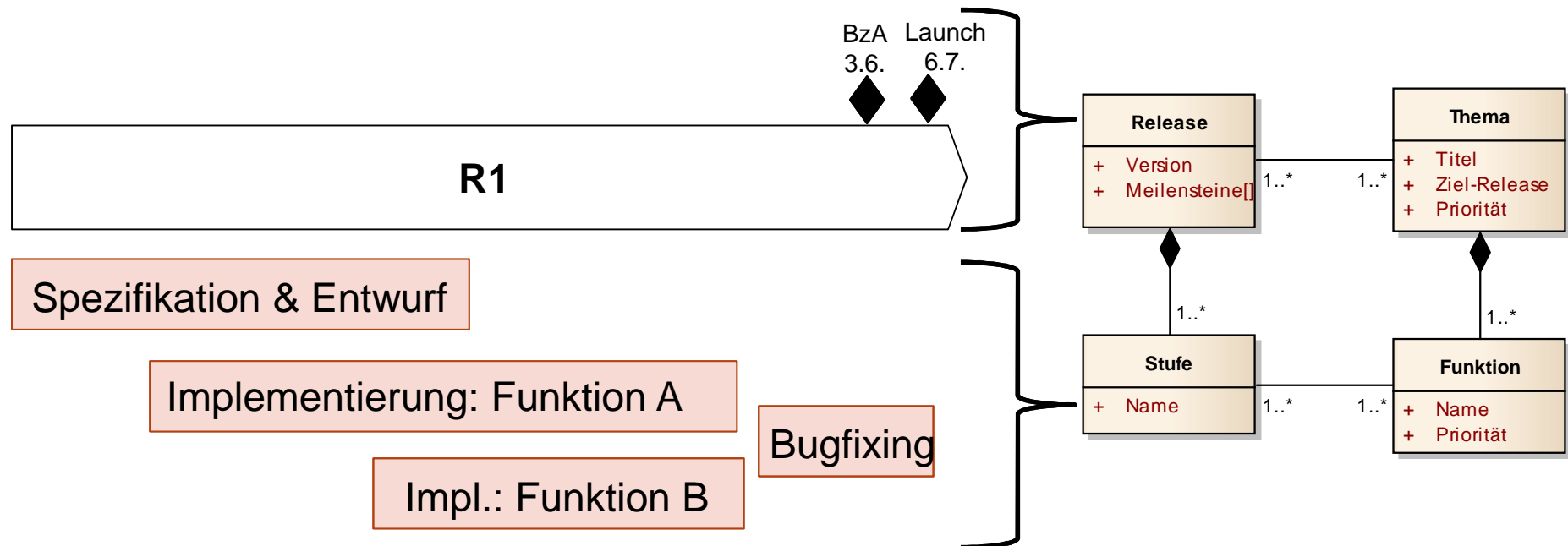
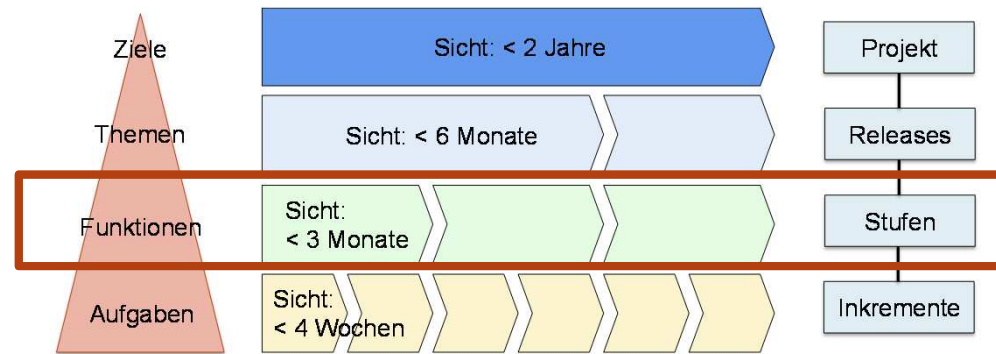
■ Funktionale Ziele:

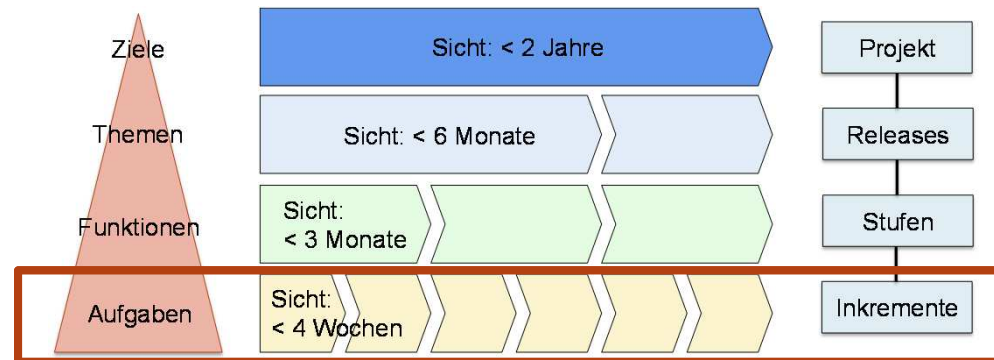
- Es steht eine Volltextsuche für die Dokumente in den SVN-Repositorys zur Verfügung.
- Der Zugriff auf die Volltextsuche ist per Web-Browser möglich.

■ Geschäftliche Ziele:

- Der Aufwand zum Finden von Dokumenten ist verringert.



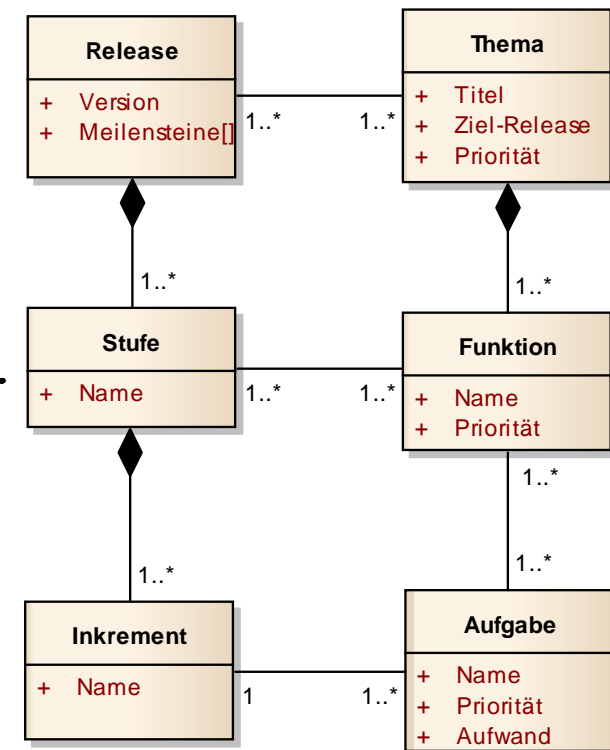




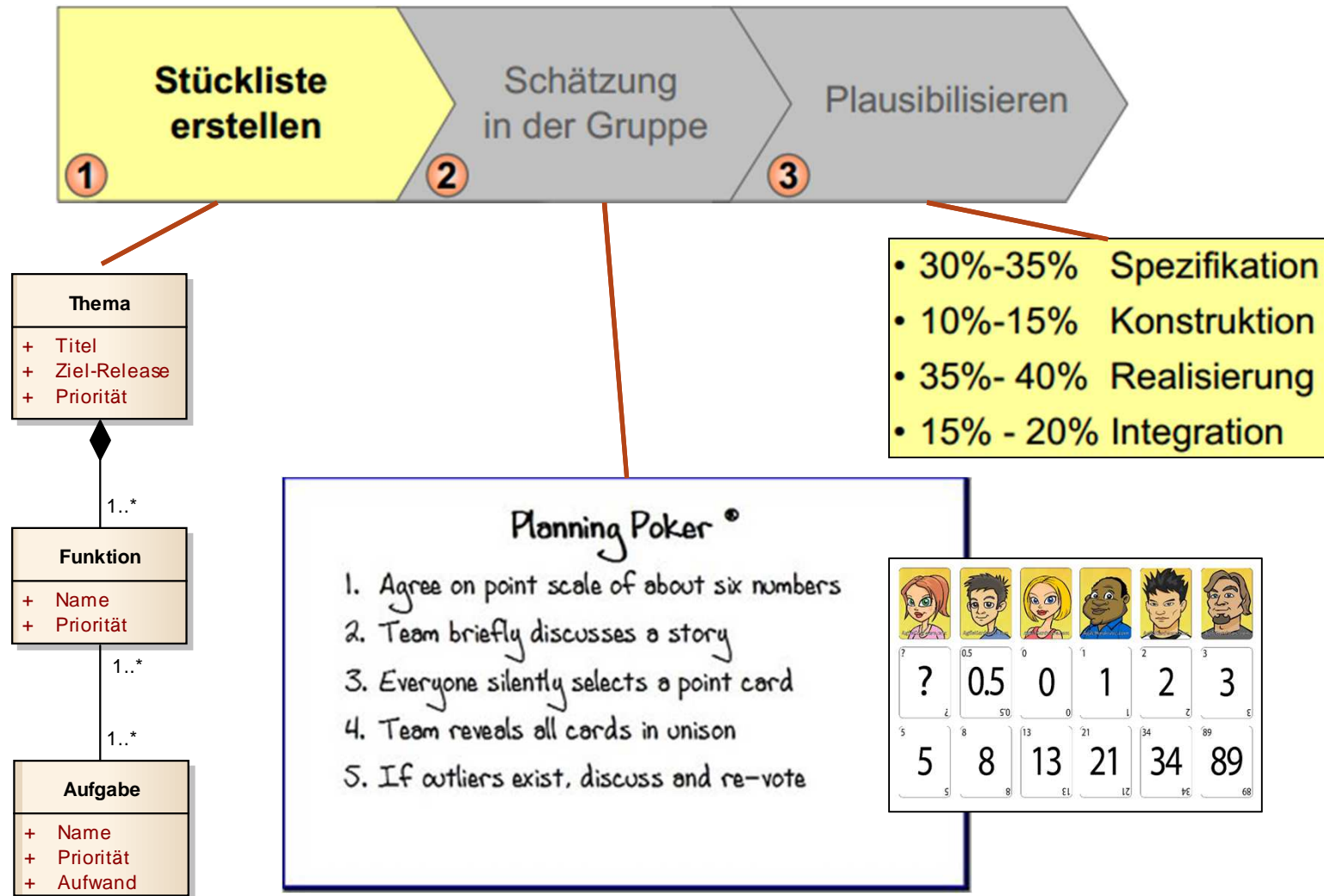
Sprint 10 (1/2) - Release 2012-1

Ziele:
 - Reporting lässt keine Fragen offen
 - CI Builds sind erfolgreich

(User-) Story	Akzeptanz Kriterien
Release 2011-5-1 bauen (inkl. Amazon, Facebook und Verivox) <ul style="list-style-type: none"> • Loop-Bug • postmessage • Homepage (Analyse + ggf. Bugfix) • Toolbar Zalando - Modul wird nicht geladen (PSNDIE-1398) • Workaround UCS Profil Problem • Clone Konfig Anpassung • PME Delta Update 	Get-well Release wird erfolgreich live gestellt und die genannten Fehler treten nicht mehr auf. <ul style="list-style-type: none"> • Homepage Modul zeigt wieder Inhalte • Zalando Modul wird erfolgreich geladen • Der Nutzer erhält korrekt Konfiguration • Security Proxy kann erfolgreich auf der Clone Umgebung installiert werden
Bugfix für 2012-1 <ul style="list-style-type: none"> • Analyse ATI Pixel: Action vs. Ausstiegsklicks (PSNDIE-1399) • Sicherheitsmodul - lässt sich nicht mehr aus der Modulliste entfernen (PSNDIE-1397) 	<ul style="list-style-type: none"> • Wir wissen, warum die Ausstiegsklicks als Actionslicks einlaufen und es gibt eine umsetzbare Lösungsvariante. • Das Modul lässt sich durch den Nutzer aus der Modulliste entfernen und bleibt raus, bis sich die Sicherheitslage wieder ändert.
Systemdokumentation erstellen <ul style="list-style-type: none"> • Inkl. CP Dokumentation 	Systemdokumentation liegt der Technik und Applikationsbetreuung vor CP Dokumentation liegt dem Modul Team vor
Betriebsschulung durchführen (inkl. Vorbereitungstelco am 24.0.25.11)	Betriebsschulung für die Applikationsbetreuung wurde durchgeführt
Überarbeitung Selectbox widget in Widgetbox (insb. für Verivox Modul)	Mit SE abzustimmen
Modul CI-Builds funktionsfähig machen, inklusive Vor-Ort Termin und Bewertung Haris Arbeitsergebnissen (TB: 2BT)	Modul CI Builds sind erfolgreich und idealerweise konnten Haris Selenium Tests integriert werden

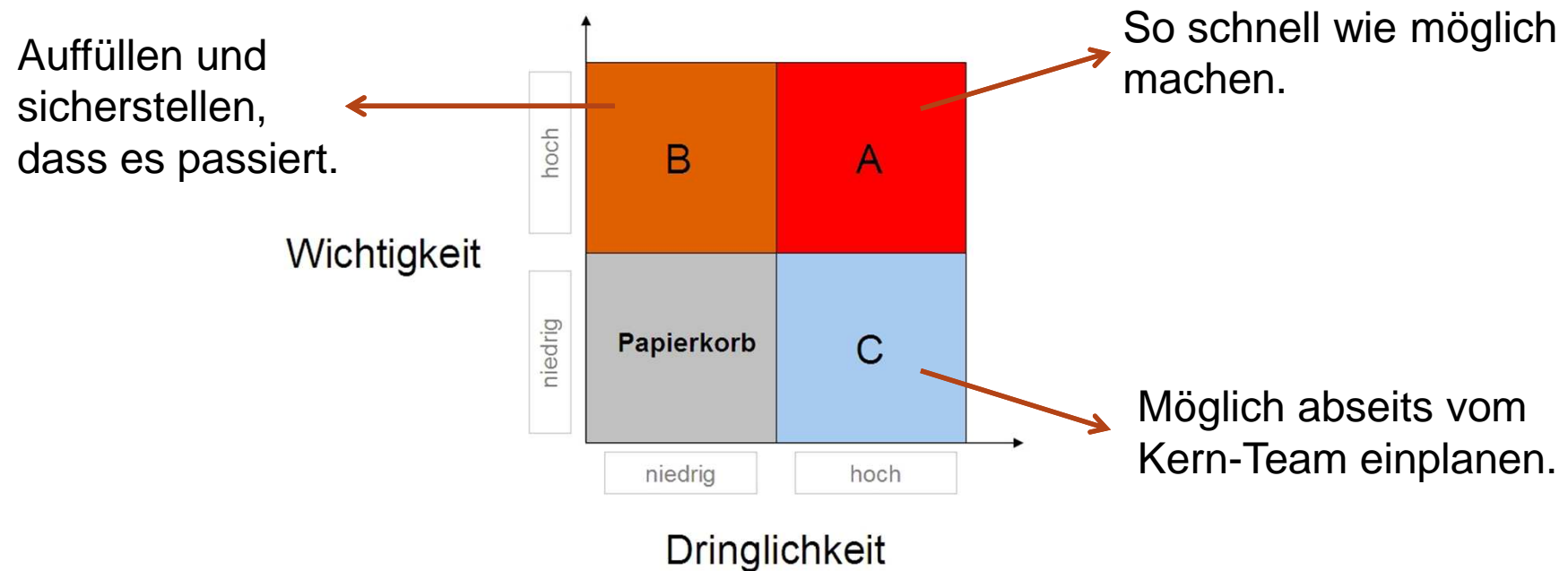


Methode: Expertenklausur (Schätzverfahren)



Methode: Eisenhower-Matrix (Priorisierungsverfahren)

- **Wichtigkeit.** Einflussfaktoren: Nutzen, Risiko, Aufwand
- **Dringlichkeit:** Was passiert wenn man es nicht macht?



- Über Eisenhower hinaus:

- **Frontloading:** Dinge mit hohem Risiko und hohem Nutzen früh einplanen.
- **Quick-Wins:** Dinge mit hohem Nutzen und geringem Aufwand früh einplanen.

Das Studienprojekt

Die ersten Schritte

HEUTE

- Themenvergabe
- Vorstellung Übung
Interaction Design

ÜBER OSTERN

- Übung: Interaction
Design Studienarbeit
- Einarbeitung
Technologie

HEUTIGE ÜBUNG (8. April)

- Peer-Review ID
- Gespräch Betreuer
- Übung: Spezifikation
und Planung

DIESE WOCHE

- Übung: Spezifikation
und Planung
- Einarbeitung
Technologie

NÄCHSTE ÜBUNG (15. April)

- Gespräch Betreuer
- Übung:
Softwarearchitektur

Im Übungszeitraum heute habt ihr jeweils 10 Minuten Zeit mit eurem Betreuer (Reihenfolge = Teamnummer)

	Team	Projekt	Technologienvariationen
Betreuer: Lautenschlager	1	Awareness System	Google Calendar API, Google App Engine, HTML5 (+optional: angular.js, bootstrap)
	2	Awareness System	Google Calendar API, Java Fx
	3	Awareness System	Google Calendar API, Android
	4	Awareness System	Google Calendar API, Google App Engine, Google Web Toolkit
	5	Awareness System	Google Calendar API, CloudFoundry, RichFaces
	6	Awareness System	Google Calendar API, CloudFoundry, Vaadin
Betreuer: Adersberger	7	Project Controlling System	Google Cloud Storage, Google Big Query, Google App Engine, Spring MVC, JavaScript Charting
	8	Project Controlling System	Hive, Dropwizard, JavaScript Charting
	9	Project Controlling System	Palo, Dropwizard, Microsoft Excel, JavaScript Charting