



G E B I T Solutions

Die Experten für Java, Anwendungsentwicklung und Requirements Engineering

Projektdokumentationsplattform

Als Garant für Kundenzufriedenheit und Transparenz

Tilo Sauer
Mitglied der Geschäftsleitung
GEBIT Solutions GmbH

Tom Krauß
Partner und Mitglied der Geschäftsleitung
GEBIT Solutions GmbH



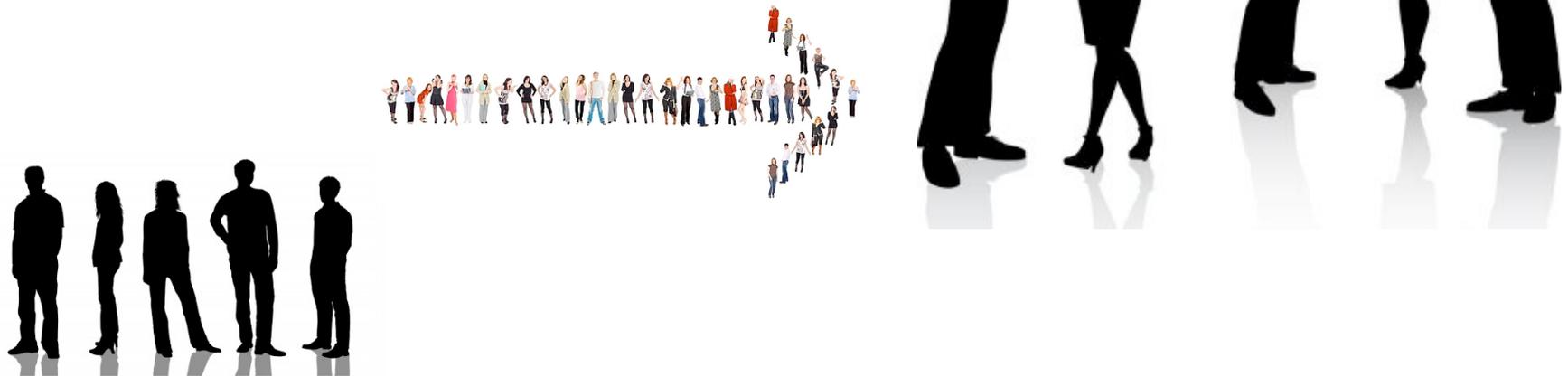
Agenda

- **Kontext / Motivation**
- **Vernetzungsplattform**
- **Disziplinen**
 - Anforderungserhebung / -management
 - Entwicklung
 - Test
 - Projektplanung /-steuerung
- **Fazit**



Immer nah am Kunden...

- **Kommunikation und Transparenz**



Das Entwicklungsvorgehen

Zielsetzungen

**Einfachere
Abstimmung der
Anforderungen**

Geschäftsbereich in den
Modellierungsprozess
mit einbeziehen

Frühzeitig
demonstrierbare
Anwendung



**Verkürzung von
„Time to
Market“**

Routine Coding
reduzieren

Besserer Re-Use



**Einfachere
Anpassungen/
Wartung**

Bessere
Nachvollziehbarkeit /
Dokumentation

Trennung von
Funktionalität und
technischem
Rahmenwerk / Plattform

Nicht den Code, sondern
das Modell anpassen



Höhere Qualität

Fehler vermeiden durch
ausgereifte
Komponenten

Mehr Konsistenz durch
Modelle



Vernetzung aller Artefakte im Projekt

- **Wir nutzen TREND/Analyst**
 - Verfügbar als kostenfreie Community Edition
 - Unabhängig vom eingesetzten Vorgehensmodell
 - Rational Unified Process
 - SCRUM
 - Eigenes...
- **Alle Projektbeteiligten haben darauf Zugriff**
 - Nicht immer möglich / unbedenklich, aber immer erstrebenswert

Gemeinsame Informationsquelle (unternehmensübergreifend)

- **Alle Beteiligten haben den selben Informationsstand und Zugriff auf die selben Informationen**
 - Projekthintergründe
 - Projektinteressen
 - Prozesse
 - Anwendungsfälle
 - Aufwände
 - Verantwortlichkeiten / Zuständigkeiten / Zuordnung
 - Projektfortschritt
 - Testinhalte und Testfortschritt
 - Kontaktinformationen (📞 , @), Abwesenheiten, ...

Modellbasierte Anforderungsanalyse

Strukturierte Beschreibung von Anforderungen mit UML und Text

- **Einfache Verwendung von UML zur Beschreibung der Fachlichkeit**
- **Vorgabe der Struktur der zu erstellenden Dokumente**
- **Vorgegebene Beziehung zwischen Dokumenttypen**

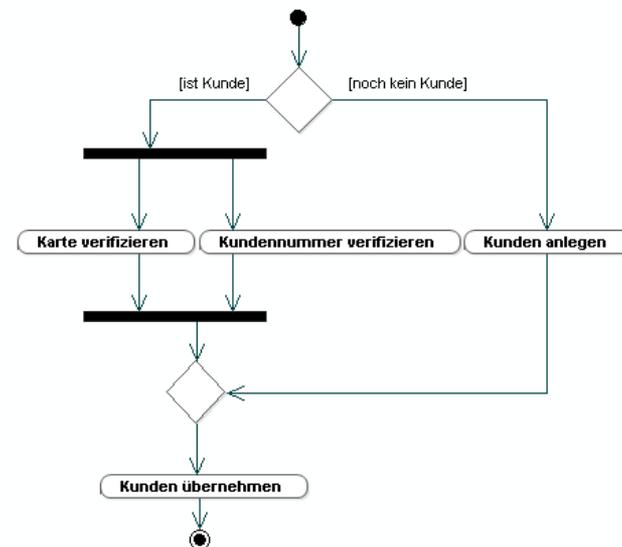
Vorbedingungen

Kunde kann sich ausweisen

Ergebnisse (Nachbedingungen)

Kundendaten im System

Standardablauf



Beschreibung der Aktionen

- **Karte verifizieren:** sollte der Kunde über eine Kundenkarte verfügen, wird diese eingelesen. Sollte sich an der Struktur der Kundendaten geändert haben, wird diese auf die Karte geschrieben, ohne dass der Prozess der Reservierung verzögert wird.
- **Kundennummer verifizieren:** Der Kunde wird über die Karte identifiziert, parallel kann eine Anfrage zur Identifizierung über die Kundennummer gestartet werden.
- **Kunden anlegen:** Sollte der Kunde keine Karte besitzen, wird ihm eine vorläufige Karte in Papierform erstellt. Später bekommt er eine Plastikkarte zugeschickt.
- **Kunden übernehmen:** Die Kundendaten werden für den weiteren Vorgang übernommen, können aber jederzeit geändert werden.

Modellbasierte Anforderungsanalyse

Ein Beispiel

Anforderungen - Anwendungsfallpaket - Traveledemo-Analysis/Flugreservierungssystem/Flugreservierung.ucp - TREND Development

File Edit Einfügen Format Navigate Suchen Project Run TREND Window Help

Debug Anforderung... Java Resource

Dokumente durchsuchen

Dokument Explorer

- Traveledemo [10.0.0.9]
 - Traveledemo-Analysis [10.0.0.9]
 - Flugreservierung 1.7
 - Flugreservierungssystem 1.9
 - Flugreservierungssystem für Reiseagenturen
 - Änderungsanträge
 - Funktionale Anforderungen
 - Flugreservierung 1.9
 - Kunden aktivieren 1.8
 - Reservierung bearbeiten 1.8
 - Reservierung stornieren 1.3
 - Reservierung ändern (Personenan...
 - Reservierung bearbeiten (Personen...
 - Reservierung bearbeiten (Option f...
 - Reservierung bearbeiten 1.2
 - Reservierung stornieren 1.2

Reservierung bear... Reservierung bear... Reservierung stor... Flugreservierung x1

Inhalt

- Zusammenfassung
- Anwendungsfalldiagramm
- Anwendungsfallspezifikationen
- Verweise von

Zusammenfassung

Dieses Anwendungsfallpaket enthält alle Anwendungsfälle, die zur Erstellung, Bearbeitung und Stornierung einer Reservierung in einer Reiseagentur benötigt werden.

Anwendungsfalldiagramm

```

    usecaseDiagram
      title Flugreservierung
      usecase UC1 as «Functional» Kunden aktivieren
      usecase UC2 as «Sub Component» Kunden
      UC1 --|> UC2 : «include»
  
```

The diagram shows a use case boundary for 'Flugreservierung'. Inside, there is a green oval use case '«Functional» Kunden aktivieren' and an orange oval use case '«Sub Component» Kunden'. A solid line with an open arrowhead points from the green use case to the orange use case, labeled with '«include»'.

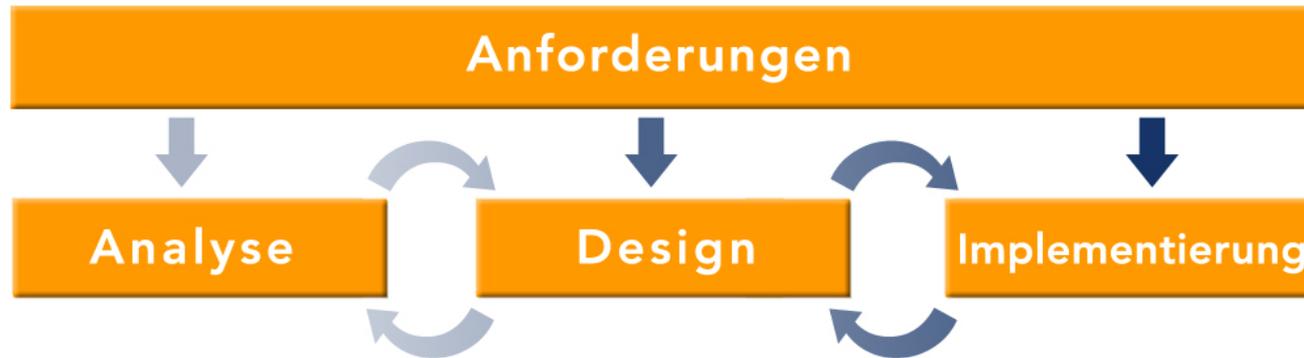
Verweise Properties Documentation

| Name | Eigenschaft |
|----------------------|-----------------------------|
| Bestätige Flug | Anwendungsfallspezifikation |
| Boardkarte drucken | Anwendungsfallspezifikation |
| Boardkarte erstellen | Anwendungsfallspezifikation |

Ausgehend Einkommend

Ansicht Bearbeitung Management

Modellbasierte Anforderungsanalyse von den Anforderungen inkrementell zur Implementierung



Generierung - Generische Komponenten



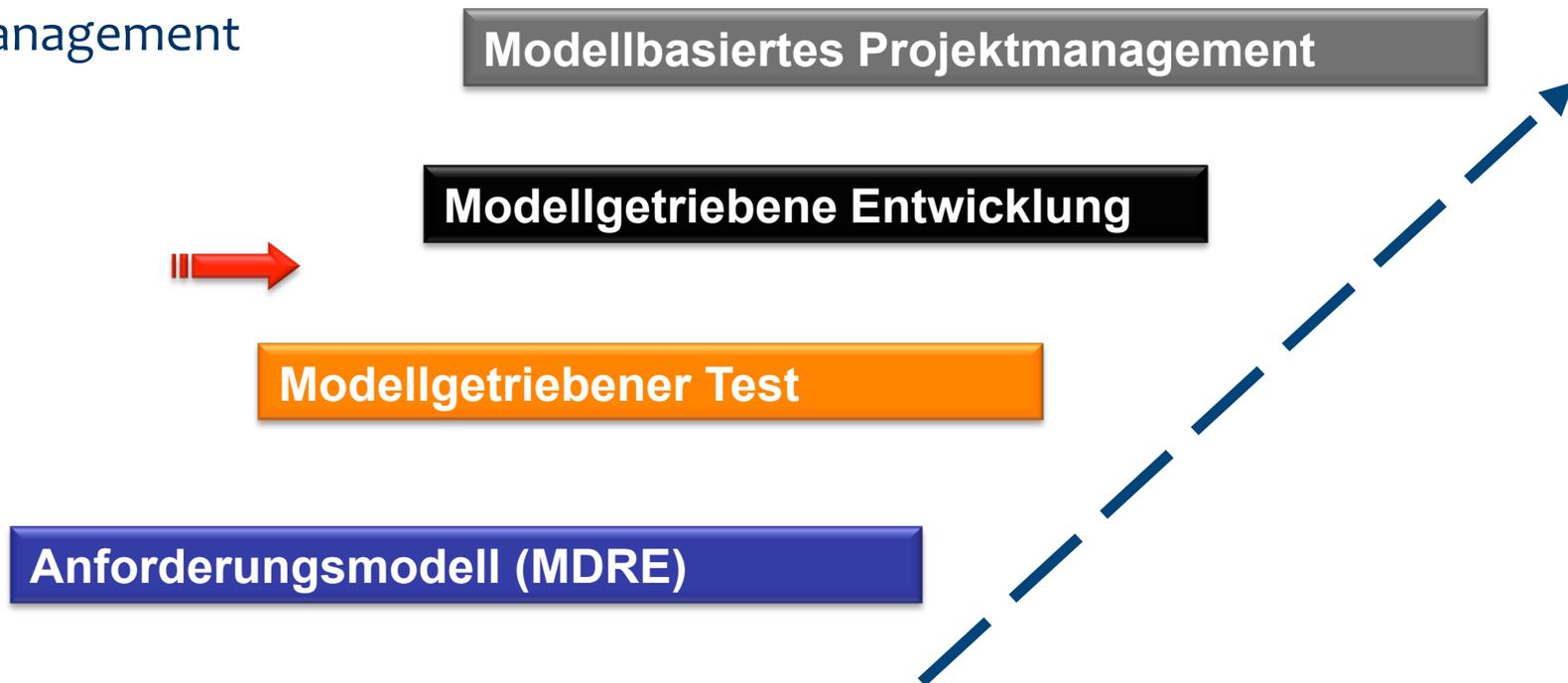
Prototyp \Rightarrow Anwendung 

Erfolgsfaktoren für einen Projektplanungs- und Steuerungsprozess

- **Pragmatisch einsetzbar**
 - Administrationsaufwand muss überschaubar sein
 - Synergieeffekte sollten so effizient wie möglich genutzt werden
- **Akzeptanz bei den Projektmitgliedern**
 - Hohe Transparenz für alle Projektbeteiligten
 - Vertrauen über Verlässlichkeit der Statusaussagen
- **Einfache Integration in bekannte Werkzeuge und existierende Verfahren**
 - Integration mit akzeptierten Projektsteuerungs-Werkzeugen wie MS Project und Excel
 - Integration mit Projektkommunikationswerkzeugen wie WIKI,

Modellbasiert: Synergieeffekte eines Anwendungsfallmodells nutzen

Use Cases als Basis für
Anforderungsanalyse
Projektmanagement
Test
v.a.m.

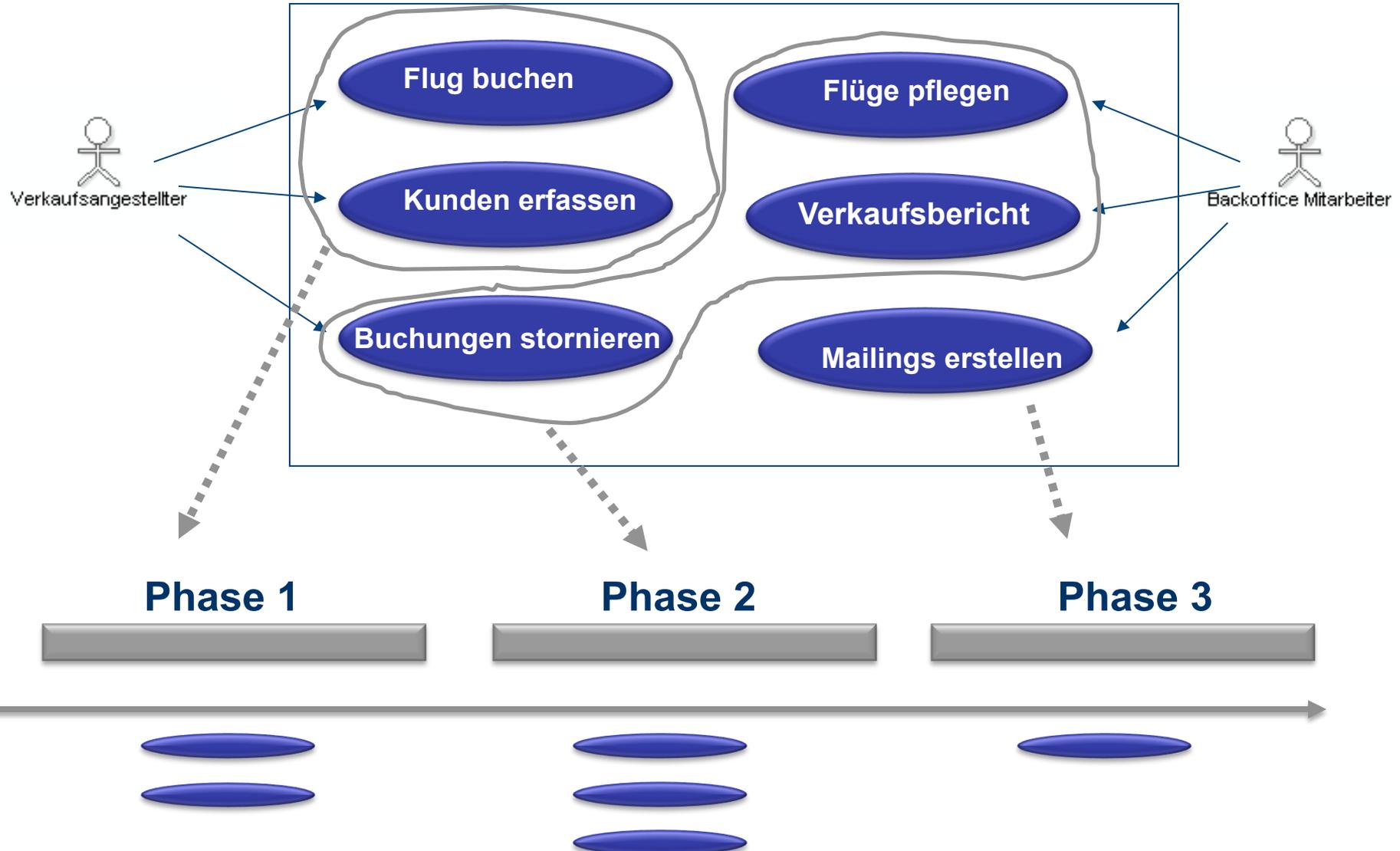


Das Planungsvorgehen

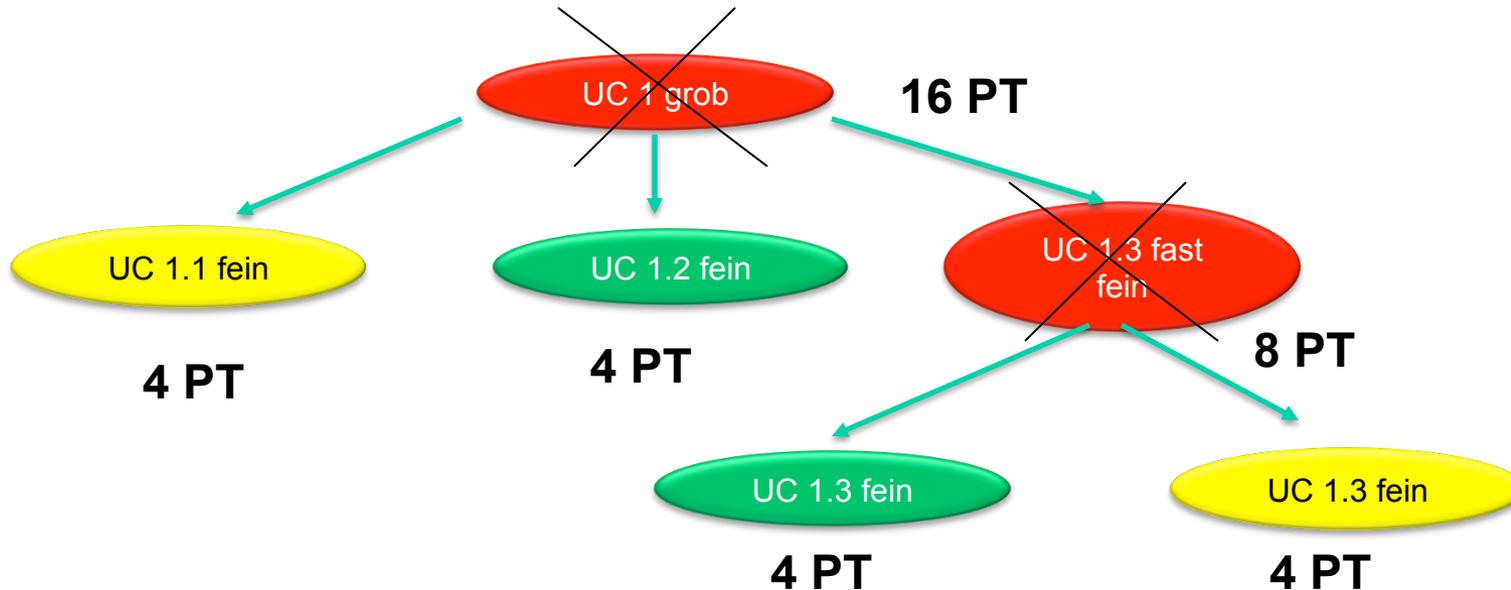
- **In der Analyse-Phase werden die Use Cases grob analysiert**
- **Zu Projektbeginn**
 - Grobschätzung der Entwicklungsaufwände pro Use Case
 - Planung der Tätigkeiten
 - Zuordnung zu Projektphase
 - Zuordnung zu Teams
- **Zu Phasenbeginn**
 - Feinplanung der Use Cases
 - Zerlegung in Tasks/Features und Zuweisung an einzelne Teammitglieder

Aufwand zur Implementierung eines Implementierungstasks sollte maximal 5 Tage betragen !

Die Grobplanung



Die Feinplanung (zu Phasenbeginn)



- In Absprache mit den Teammitgliedern Unterpakete für Aufgaben definieren
- Vorgehen ;-)

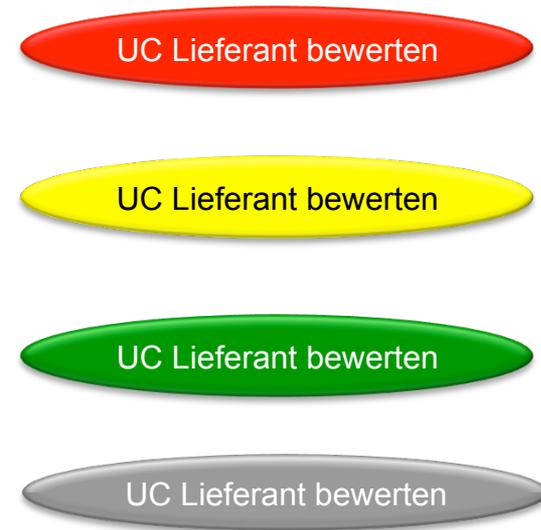
```

while (PT > 5) do
    UCx.y = UCx DIVIDED...
  
```

Statusverfolgung im Use Case-Modell

- **Der Umsetzungsstatus eines Tasks/UseCases wird per Eigenschaft definiert**
- **Mögliche Statuswerte:**
 - **geplant**
 - **implementiert**
 - **abgenommen**
 - **nicht geplant**
 - **zurückgestellt**
 - **gestrichen**

Entwickler ist fertig und Fachvertreter stimmt dem ebenfalls zu



Was abgenommen ist darf nicht mehr angefasst werden

*der; -, -, Pl. [.. u:s] <lat.>



Randbemerkungen zum Status

- **Statusübergänge nur in eine Richtung möglich**
 - Größere Änderungen in den Anforderungen nach Implementierung werden als neuer Use Case (Typ Change Request) erfasst
- **Weitere Status nach Bedarf definieren (nicht „übertreiben“)**
 - „Verschoben“
 - „Abgelehnt“

Beachte:

- ✓ Kriterien definieren, wann ein Status erreicht wird
- ✓ Verantwortlichkeiten klären
- ❖ % - Grad der Umsetzung eines Features spielt keine Rolle.
Dieser kann nicht überprüft werden

Projekt Monitoring

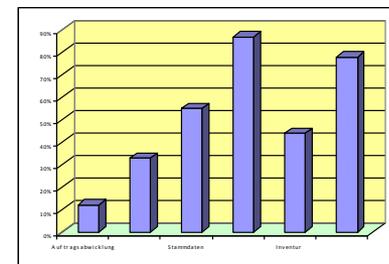
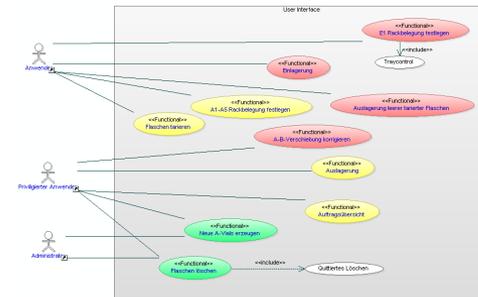
• Entwickler

- pflegen in Abstimmung mit den Fachverantwortlichen Statusänderungen über das UC-Modell

• Projektstatus für alle sofort nachvollziehbar

• Projektleiter

- Fortschrittsreporting des Erreichungsgrades erfolgt anhand des Use Case Status in der Form „x von insgesamt y geplanten PT umgesetzt“.
- Verdichtung über Teams, Phasen, Gesamtprojekt



Typischer Statusverlauf

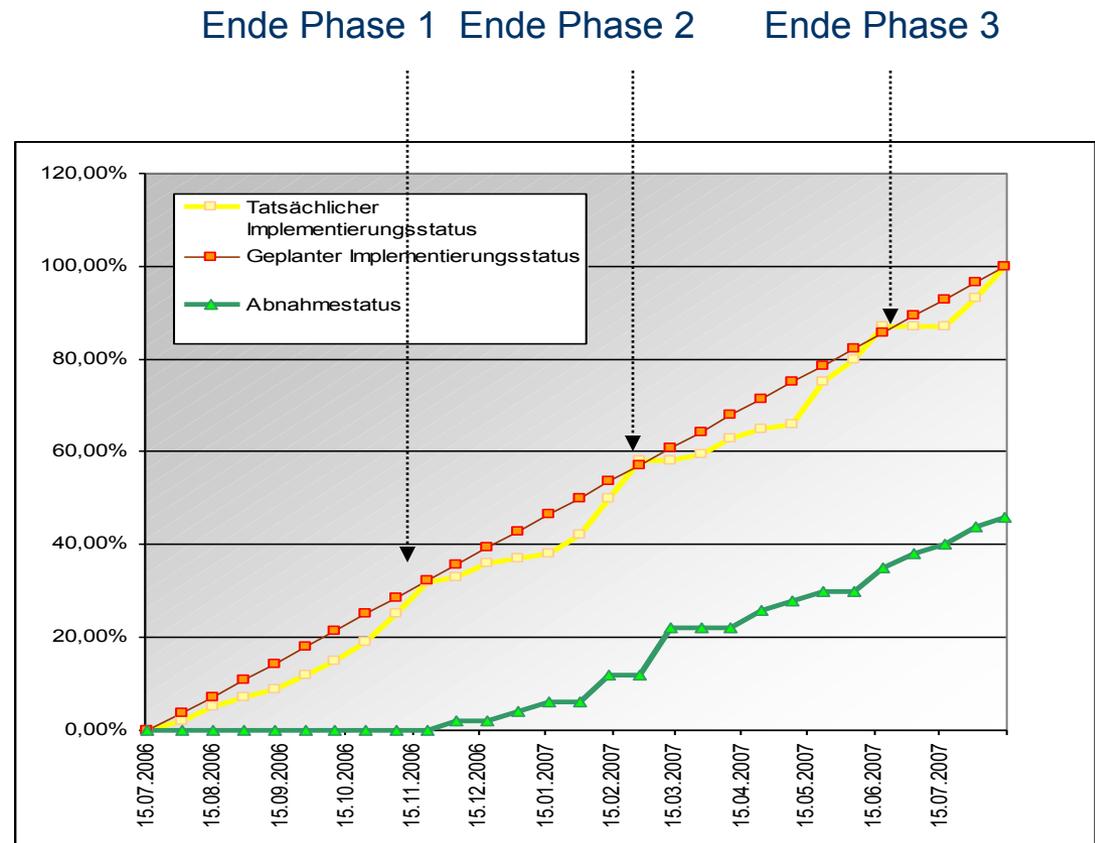
- Status „hängt während Projektphase durch“

- Prozentuelle Fertigstellung nicht berücksichtigt

- Phasenbeginn

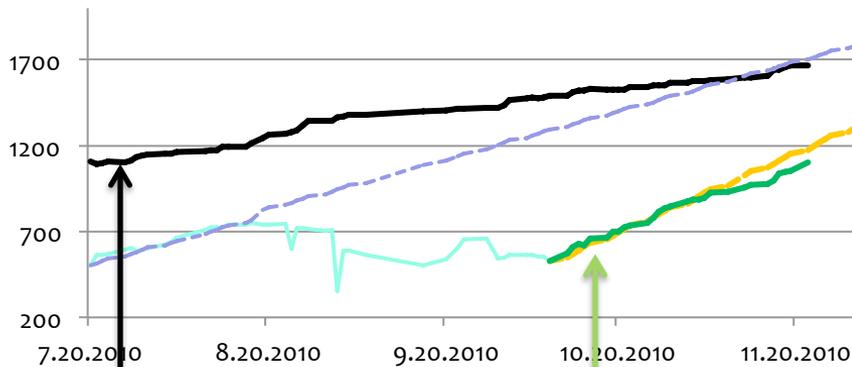
- Wenig Fortschritt
- Planung und Bug Fixing

- **Wichtig:**
Abnahmestatus ernst nehmen



Beispiele für Statusverfolgung mit Use Cases

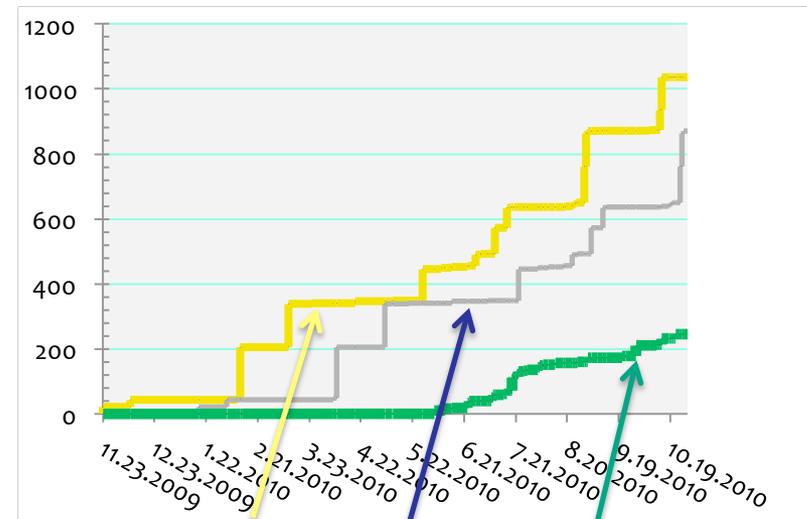
Testabdeckung über die Zeit



Umgesetzte Anwendungsfälle

Getestete Anwendungsfälle

Fertigstellung und fachliche Freigaben



Umgesetzt

Freigabe soll

Freigabe ist

- **Transparenz auf AG <-> AN Ebene kritischer Erfolgsfaktor**
- **Unabhängig vom eingesetzten Werkzeug**
- **Offenheit, Transparenz und Kommunikation schafft Vertrauen (auch bzw. gerade im Festpreisprojekt)**
- **Erfahrung in zahlreichen Projekten sehr positiv**
 - Hohe Kundenzufriedenheit
- **Wert der vernetzten Informationen ist für Kunden oft erst sichtbar, wenn der Ansatz „Früchte trägt“**