



Übung 3: Vorgehen in Software-Projekten

Ziel dieser Übung ist, dass Sie anhand eines kleinen Beispiels verstehen, in welchen Schritten in Software-Entwicklungsprojekten vorgegangen wird. Sie sollten in der Lage sein, sich ein einfaches Vorgehen selbst zu überlegen und dieses als Flussdiagramm zu modellieren. Und Sie wissen, wie Sie Rollen, Produkte/Ergebnisse und Aktivitäten darstellen können.

Schließlich lernen Sie noch wesentliche Einflussgrößen auf Projekte kennen und untersuchen diese am folgenden Beispiel.

Rahmengeschichte

Herzlichen Glückwunsch! Sie haben gerade eine neue Rosenheimer Software- und Consulting Firma (RoSCoF) gegründet und stellen ab heute Webapplikationen für Ihre Kunden her.

Ihre ersten Kunden sind Silvio Bocaccio¹ und Umberto Rossi, beide sind Inhaber der Pizzeria „Il Pappagallo“. Ihre beiden Kunden möchten von Ihnen eine Webapplikation gebaut haben, mit der man über das Internet Pizza und andere Speisen bestellen kann; sie möchten also eine Bringdienst-Software. Beide Kunden stehen Ihnen pro Tag für einige Stunden vor Ort zur Verfügung, um Ihnen die Anforderungen an die Webapplikation zu nennen.

Sie entschließen sich, keine vollständige Spezifikation (sie wissen ja noch nicht, was das ist) zu schreiben! Ihre Kunden sollen die Anforderungen an die Funktionen/Features der Webapplikation auf **Karteikarten** schreiben. Auf einer Karteikarte steht in höchstens drei Sätzen, wie eine geforderte Funktion aussehen soll.

Beispiel für eine Karteikarte:

Customer Story and Task Card		BIN Development \ COLA	
DATE: 3/19/98	TYPE OF ACTIVITY: NEW: <input checked="" type="checkbox"/> FIX: <input type="checkbox"/> ENHANCE: <input type="checkbox"/> FUNC. TEST: <input type="checkbox"/>		
STORY NUMBER: 1275 1275	PRIORITY: USER: <input type="checkbox"/> TECH: <input type="checkbox"/>		
PRIOR REFERENCE:	RISK: <input type="checkbox"/> TECH ESTIMATE: <input type="checkbox"/>		
TASK DESCRIPTION:			
SPLIT COLA: When the COLA rate chgs. in the middle of the BIW Pay Period, we will want to pay the 1 st week of the pay period at the OLD COLA rate and the 2 nd week of the pay period at the NEW COLA rate. Should occur automatically based on system design.			
NOTES: on system design			
For the OT, we will run a m/frame program that will pay or calc the COLA on the 2 nd week of OT. The plant currently retransmits the hours data for the 2 nd week exclusively so that we can calc COLA. This will come into the Model as a "2144" COLA			
TASK TRACKING: Gross Pay Adjustment. Create RM Boundary and Place in DEEnt Express COLA			
Date	Status	To Do	Comments

¹ Namen der Kunden und die Pizzeria sind frei erfunden. Übereinstimmungen mit lebenden Personen und/oder Lokalen wären rein zufällig.

Aufgabe 1: Erstellung eines Vorgehensmodells

Ihre Aufgabe als Manager von RoSCoF besteht darin, dass Sie sich für Ihre fünf ambitionierten Mitarbeiter ein ***Vorgehen(smodell)*** überlegen, mit dem die Webapplikation programmiert werden kann.

- a) Überlegen Sie sich, wie die Anforderungen auf Karteikarten in eine ausgelieferte Teil-Webapplikation umgesetzt werden.

In Ihrem Vorgehen müssen folgende Aspekte berücksichtigt werden:

- Programmierung der Anforderungen von Karteikarten
- Entwicklertest der Anforderungen von Karteikarten (und/oder Entwicklung eines Unit-Tests für eine Anforderung)
- Auslieferung der Software an Silvio und Umberto.
- Abnahmetest der Anforderungen von einzelnen Karteikarten durch den Kunden.

- b) Anforderungen an das Vorgehen.

Definieren Sie:

- ***Rollen*** für die 5 Mitarbeiter (z.B. Tester, Entwickler, ...), zusätzlich ist die Rolle Kunde enthalten
- ***Aktivitäten*** in der Entwicklung (z.B. Quelltext schreiben, Software testen...)
- ***Zwischenergebnisse*** (z.B. Quelltext)
- Einen ***Ablauf***, der darstellt, in welcher Reihenfolge die Aktivitäten ausgeführt werden, zu welchem Zwischenergebnis diese beitragen und wer für die Aktivitäten verantwortlich ist.

- c) Dokumentieren Sie schließlich Ihren Ablauf, d.h. auch Rollen, Aktivitäten und Zwischenergebnisse, als Flussdiagramm etwa mithilfe des Werkzeugs MS-Visio.

Als Hilfestellung finden Sie ***im Anhang*** Flussdiagramm-Symbole (nach DIN 66001) sowie ein Beispiel mit so genannten „Swimlanes“.

Aufgabe 2: Einflussgrößen auf Projekte

Als Manager von RoSCoF möchten Sie natürlich auch eine geeignete Strategie vorgeben können. Dazu hilft es, Ihr Entwicklungsprojekt für die Bringdienst-Software einzuordnen.

Boehm und Turner haben in ihrem Buch „Balancing Agility and Discipline“ ein Einordnungsschema für Projekte vorgeschlagen. Das Schema hat fünf Kriterien:

Kriterium	Beschreibung
Personnel	Wie gut sind die Personen im Projekt qualifiziert? Wie groß ist der Anteil der IT-Profis (Cockburn Level 2 und 3) ² ? Mögliche Stufen sind hier 35%, 30%, 25%, 20%, 15%
Dynamism	Wie groß ist der Anteil der Anforderungen, die sich monatlich ändern? Mögliche Stufen sind hier 50%, 30%, 10%, 5%, 1%
Culture	Wie groß ist der Anteil der Personen, die möglichst viele Freiheiten bei der Gestaltung ihrer Projekte haben wollen? Mögliche Stufen sind hier: 90%, 70%, 50%, 30%, 10%
Size	Wie viele Personen arbeiten im Projekt mit? Mögliche Stufen sind hier: 5, 10, 50, 100, 500, ...
Criticality	Wie (sicherheits-)kritisch ist das Projekt? Mögliche Stufen sind hier: Verlust von Komfort, Verlust von (frei verfügbaren) Finanzmitteln, Verlust von essentiellen Finanzmitteln (=Firma pleite), Verlust eines Lebens, Verlust vieler Leben

- a) Zeichnen Sie für das in Aufgabe 1 skizzierte Projekt ein Kiviat-Diagramm. (Finden Sie erst heraus, was ein Kiviat-Diagramm überhaupt ist!)
Treffen Sie gegebenenfalls geeignete Annahmen, welche Stufen für Ihr Projekt passend sein könnten und dokumentieren Sie diese.

- b) (Für Fortgeschrittene)

Erstellen Sie ein **Excel-Sheet**, bei dem Sie für ein beliebiges Projekt jeweils die 5 Kriterien angeben können – am besten per Drop-Down Liste³ – und automatisch ein entsprechendes Kiviat-Diagramm angezeigt wird!

² Cockburn meint mit Level 2 und 3:

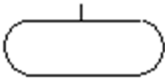

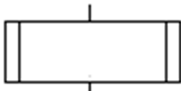
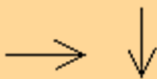
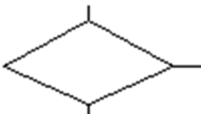
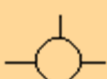
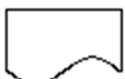
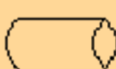
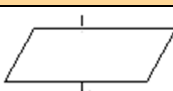
Level	Characteristics
3	Able to revise a method (break its rules) to fit an unprecedented new situation.
2	Able to tailor a method to fit a predated new situation.
1A	With training, able to perform discretionary method steps (e.g., sizing stories to fit increments, composing patterns, compound refactoring, complex COTS integration). With experience can become Level 2.
1B	With training, able to perform procedural method steps (e.g. coding a simple method, simple refactoring, following coding standards and CM procedures, running tests). With experience can master some Level 1A skills.
-1	May have technical skills, but unable or unwilling to collaborate or follow shared methods.

³ In Excel 2016: Daten → Datentools → Datenüberprüfung... → Zulassen: Liste



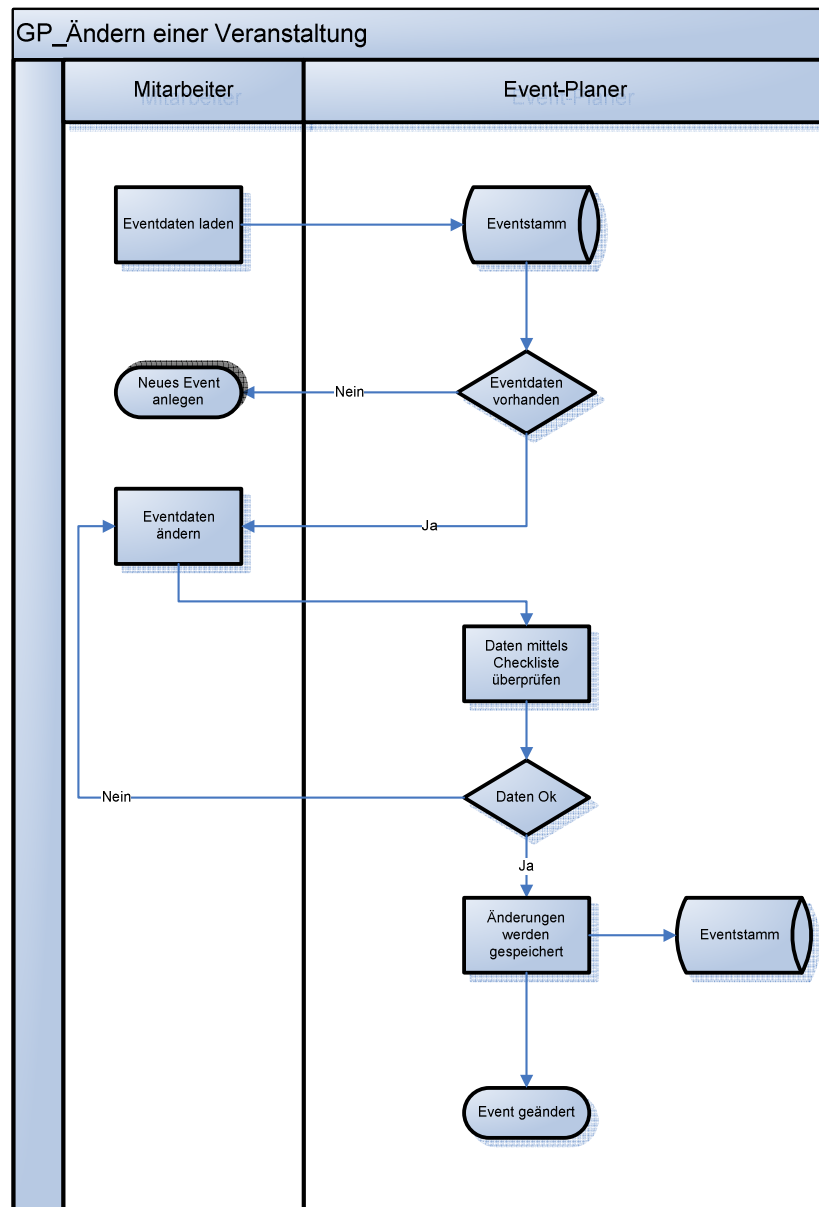
Anhang (Flussdiagramm-Symbole nach DIN 66001)

Flussdiagramme werden in der Informatik häufig verwendet. Sie dienen etwa dazu Programmabläufe und Algorithmen zu dokumentieren (Programmablaufpläne). Ebenso können Sie verwendet werden, um das Vorgehen in Software-Entwicklungsprojekten oder allgemein auch Geschäftsprozesse und andere Prozesse zu modellieren.

Symbol	Bedeutung nach DIN 66001	Bedeutung in SE 1
	Grenzstelle (terminator)	Ein Terminator ist das Start- sowie Stoppsymbol eines Vorgehensmodells. Dort endet der Prozess: z.B. Software ausgeliefert, Projekt abgebrochen, ...
	Verarbeitung allgemein (Prozess)	Mit dem „Prozess“ Symbol werden Aktivitäten modelliert, z.B. Klasse programmieren, Klasse testen ...
	Unterprogrammaufruf (predefined process); Hinweis auf Dokumentation an anderer Stelle in Form von eindeutiger Innenbeschriftung	Über „predefined process“ werden große Flussdiagramme strukturiert. Das Symbol deutet den Aufruf eines Teilprozesses (Teilvorgehens) an.
	Ablauflinien	Mithilfe der Ablauflinien wird der Kontrollfluss / Ablauf von Verarbeitungsschritt zu Verarbeitungsschritt dargestellt.
	Verzweigung (decision)	Mit dem Verzweigungssymbol werden Entscheidungen dargestellt, an denen der Kontrollfluss in verschiedene Richtungen verzweigen kann, z.B. Ein Fehler gefunden? Ja / Nein, Build erfolgreich? Ja/Nein, ...
	Verbindungsstelle (connector)	Über einen Connector werden verschiedene ggf. parallele Kontrollflüsse wieder zusammengeführt oder parallele Kontrollflüsse erzeugt.
	Daten auf Schriftstück (data on document)	Mit dem „Data on Document“ – Symbol werden Ergebnisse / Produkte / Artefakte dargestellt: z.B. Spezifikationsdokument, eine Java Klasse, ...
	Daten auf Speicher auch mit direktem Zugriff	Mit diesem Symbol wird der Zugriff auf gespeicherte Daten dargestellt. Dies können die Meldungen im Bugtracker, die Quelltexte im Repository und andere sein.
	Ein-/Ausgabe; auch: Daten, allgemein (data)	

Mit den Symbolen aus DIN 66001 ist nicht darstellbar, wer eine Aktivität durchführt bzw. wer für ein Ergebnis verantwortlich ist (Rolle). Hierzu verwenden wir die „Swimlane“ Notation, die Sie auch noch im Rahmen der UML-Aktivitätsdiagramme kennenlernen werden. MS-Visio bietet hierfür „**Funktionsübergreifende Flussdiagramm-Shapes**“ an.

Beispiel für ein Flussdiagramm mit Swimlanes⁴



⁴ Quelle: SE-2 Projekt „Eventplaner“, S. Keller et al.