

DHBW SOFTWARE SYSTEMS GMBH

# Pflichtenheft

---

## Entwicklung eines CASE-Tools

Version 1.0

**12.11.2012**

Sebastian Gugel

Aaron Ochs

Markus Zukunft

# INHALT

|                                  |    |
|----------------------------------|----|
| 1. Zielbestimmung.....           | 2  |
| 2. Produkt-Einsatz.....          | 3  |
| 3. Produkt-Umgebung.....         | 3  |
| 4. Produkt-Funktionen.....       | 4  |
| 5. Produkt-Daten.....            | 18 |
| 6. Produkt-Leistungen.....       | 19 |
| 7. Qualitäts-Zielbestimmung..... | 19 |
| 8. Testszenarien .....           | 20 |
| 9. Benutzeroberfläche.....       | 22 |
| 10. Entwicklungs-Umgebung.....   | 23 |
| 11. Ergänzungen .....            | 23 |
| 13. Glossar.....                 | 25 |

# 1. ZIELBESTIMMUNG

## MUSSKRITERIEN

- Erstellen von Projekten
- Ändern von Projekten
- Löschen von Projekten
- Eingabe der Anforderungen
- Änderung der Anforderungen
- Löschen der Anforderungen
- Durchführung einer Aufwandsschätzung
- Eingabe der Anforderungsattribute
- Änderung der Anforderungsattribute
- Löschen der Anforderungsattribute
- Projekte exportieren (XML)
- Vergleich und Anpassung von Erfahrungswerten mit Datenbankwerten
- Eingabe von tatsächlichen Werten

## WUNSCHKRITERIEN

- Unterstützung verschiedener Kostenschätzungsmodelle
- Unterstützung mehrerer Benutzer
- Lastenheft drucken
- Abfangen von Eingabefehlern
- Integration in eine DIE
- Daten speichern (OSLC)

## ABGRENZUNGSKRITERIEN

## 2. PRODUKT-EINSATZ

Das Produkt dient der einfachen Erstellung von Durchführbarkeitsstudien. Es sollen Anforderungen mit Anforderungsattributen eingegeben werden können. Außerdem sollten Aufwandsschätzungen durchgeführt werden können, um damit Kosten, Personenaufwand und Dauer zu bestimmen.

Die Durchführbarkeitsstudie sollte als XML-Datei exportiert werden können.

Nachdem ein Projekt abgeschlossen ist, sollen die tatsächlichen Werte eingegeben werden können, um damit die Datenbankwerte zu vergleichen und diese gegebenenfalls anzupassen.

### ANWENDUNGSBEREICHE

Das Produkt wird in der Softwareentwicklung zur Aufwandseinschätzung eingesetzt.

### ZIELGRUPPEN

Das Produkt soll von Projektleitern eingesetzt werden, die bereits Erfahrung mit Aufwandseinschätzung gesammelt haben.

### BETRIEBSBEDINGUNGEN

Das Produkt soll auf Desktop-PCs mit aktueller Hard- und Software, in Büroumgebung lauffähig sein. Es wird hauptsächlich während der normalen Arbeitszeiten benutzt und muss nicht durchgängig funktionieren.

Für die Speicherung der tatsächlichen Werte wird eine Datenbank auf einem Server bereitgestellt.

## 3. PRODUKT-UMGEBUNG

| Umgebung | Spezifikationen  |
|----------|--|
| Software | Kompatible Betriebssysteme: <ul style="list-style-type: none"><li>▪ Windows XP oder neuer</li><li>▪ Linux 2 oder höher</li><li>▪ Mac OSX Version 10.6 oder höher</li></ul> Java JRE Version 6 oder höher |
| Hardware | x64 oder x86 Prozessor<br>Mindestens 1GB RAM<br>Grafikchip<br>Bildschirm<br>Tastatur<br>Maus<br>Netzwerkverbindung<br>Speichermedium mit mindestens 100 MB freiem Speicher                               |
| Orgware  | Datenbankanbindung   |

## 4. PRODUKT-FUNKTIONEN

/F10/ (LF10)

**Geschäftsprozess:** Eingabe der Anforderungen

**Ziel:** Die Anforderungen werden nach Eingabe durch den Benutzer in einer Liste gespeichert.

**Kategorie:** Primär

**Vorbedingung:** Es muss ein Projekt vorhanden sein.

**Nachbedingung Erfolg:** Die Anforderungen sind in der Liste gespeichert.

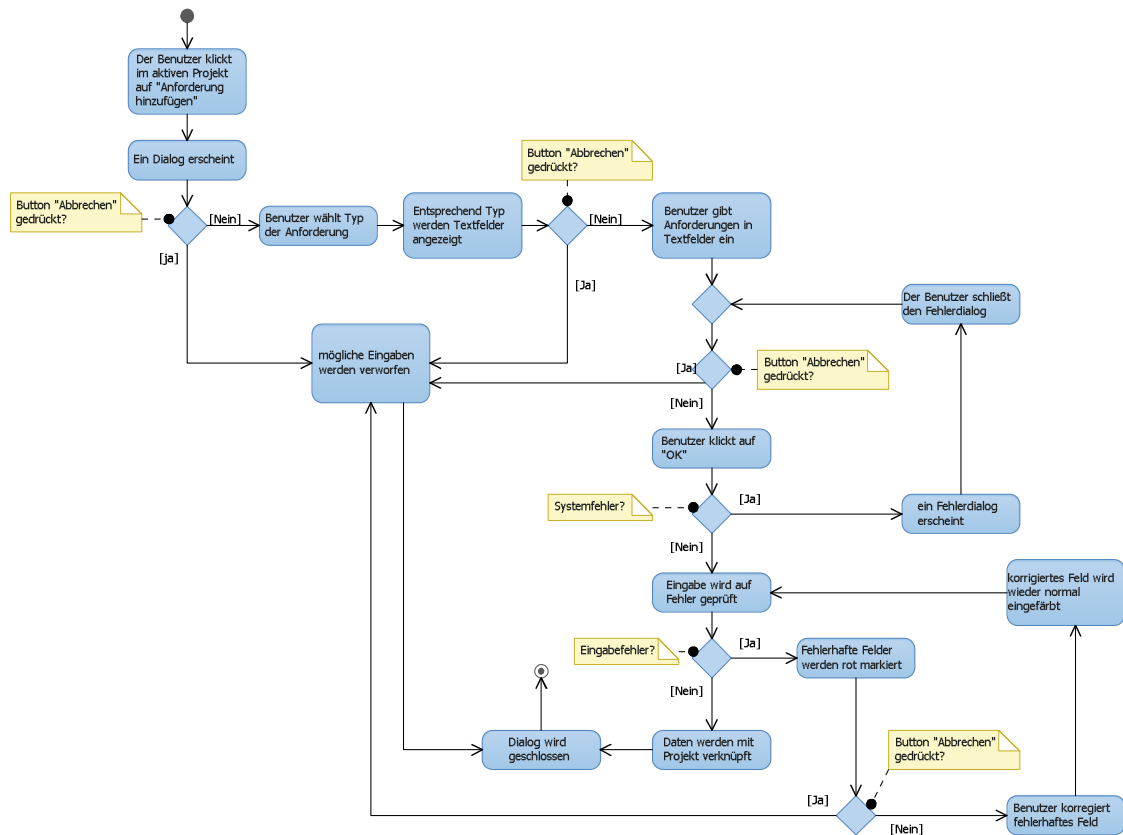
**Nachbedingung Fehlschlag:** Die Anforderungen werden nicht gespeichert und für Systemfehler eine Fehlermeldung angezeigt. Die Benutzereingabe bleibt dabei erhalten.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer klickt auf den Button, der eine neue Anforderung hinzufügt.

**Beschreibung:**

1. Ein Dialog zur Eingabe der Anforderung wird angezeigt.
2. Der Typ der Anforderung wird gewählt. Hierbei stehen qualitative, funktionale und Leistungsanforderungen, sowie Produktdaten zur Auswahl.
3. Der Benutzer gibt die Anforderung, entsprechend dem gewählten Typ, in ein Textfeld ein.
4. Der Benutzer kann die Eingabe bestätigen oder Abbrechen.
5. Bei Bestätigung werden die Daten mit dem Projekt verknüpft und Fehler gegebenenfalls abgefangen und angezeigt.
6. Bei Abbruch wird die Eingabe verworfen.
7. Nach Bestätigung oder Abbruch wird der Dialog ausgeblendet.



#### Alternativen:

1. Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.
2. Bei Eingabefehlern wird nach F70 verfahren.

**Erweiterungen:** Keine.

/F20/ (LF11)

**Geschäftsprozess:** Änderung von Anforderungen

**Ziel:** Eine Anforderung wird geändert.

**Kategorie:** Primär

**Vorbedingung:** Es muss mindestens eine Anforderung vorhanden sein.

**Nachbedingung Erfolg:** Die Änderungen der Anforderung werden übernommen.

**Nachbedingung Fehlschlag:** Die Änderungen werden verworfen.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer wählt mit der rechten Maustaste eine bestehende Anforderung aus einer Liste aus und drückt in einem dadurch geöffneten Kontextmenü den Button zur Bearbeitung der Anforderung.

**Beschreibung:**

1. Eingabefelder werden in einem Dialog mit den vorhandenen Daten angezeigt.
2. Der Benutzer ändert qualitative, funktionale oder Leistungsanforderungen, sowie ihre Eigenschaften.
3. Der Benutzer kann die Änderung bestätigen oder Abbrechen.
4. Bei Bestätigung werden vorgenommene Änderungen übernommen.
5. Bei Abbruch werden die Änderungen verworfen.
6. Der Dialog wird ausgeblendet

**Alternativen:**

1. Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.
2. Bei Eingabefehlern wird nach F70 verfahren.

**Erweiterungen:** Keine.

/F30/ (LF12)

**Geschäftsprozess:** Löschen von Anforderungen

**Ziel:** Eine Anforderung wird gelöscht.

**Kategorie:** Primär

**Vorbedingung:** Es muss mindestens eine Anforderung vorhanden sein.

**Nachbedingung Erfolg:** Die Anforderung wird aus der Liste im Programm gelöscht.

**Nachbedingung Fehlschlag:** Die Anforderung bleibt erhalten.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer wählt mit der rechten Maustaste eine bestehende Anforderung aus einer Liste aus und drückt in einem dadurch geöffneten Kontextmenü den Button zum Löschen der Anforderung.

**Beschreibung:**

1. Es wird ein Dialog zur Bestätigung mit den Buttons „Ja“, sowie „Nein“, angezeigt.
2. Bei Bestätigung durch „Ja“ wird die Anforderung aus der Liste im Programm gelöscht, die Daten bleiben auf der Festplatte erhalten.
3. Bei Auswahl von „Nein“ wird der Löschvorgang abgebrochen und die Anforderung bleibt erhalten.
4. Der Dialog wird ausgeblendet.

**Alternativen:** Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.

**Erweiterungen:** Keine.

/F40/ (LF40)

**Geschäftsprozess:** Eingabe der Anforderungsattribute.

**Ziel:** Der Benutzer gibt Faktoren zur Komplexität und den Kosten ein.

**Kategorie:** Primär

**Vorbedingung:** Es müssen Anforderungen vorhanden sein.

**Nachbedingung Erfolg:** Die im System gespeicherten Anforderungen werden durch Anforderungsattribute erweitert.

**Nachbedingung Fehlschlag:** Die Attribute werden nicht gespeichert und je nach Art des Fehlers eine passende Fehlermeldung angezeigt. Die Benutzereingabe bleibt dabei erhalten.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer wählt eine Anforderung aus und klickt auf einen Button, der Attribute zur Anforderung hinzufügt.

**Beschreibung:**

1. Der Benutzer wählt in einem Dialog aus, ob es sich um ein Kosten- oder Komplexitätsattribut handelt.
2. Abhängig von der Auswahl des Benutzers werden entsprechende Eingabefelder im Dialog angezeigt.
3. Der Benutzer gibt die Eigenschaften in die Eingabefelder ein.
4. Der Benutzer kann die Eingabe bestätigen oder Abbrechen.
5. Bei Bestätigung werden die Attribute zu der Anforderung hinzugefügt und Fehler gegebenenfalls abgefangen und angezeigt.
6. Bei Abbruch wird die Eingabe verworfen.
7. Die Eingabefelder werden ausgeblendet

**Alternativen:**

1. Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.
2. Bei Eingabefehlern wird nach F70 verfahren.

**Erweiterungen:** Keine.



## /F50/ (LF41)

**Geschäftsprozess:** Änderung der Anforderungsattribute.

**Ziel:** Der Benutzer ändert Faktoren zur Komplexität und den Kosten.

**Kategorie:** Primär

**Vorbedingung:** Anforderungsattribute müssen vorhanden sein.

**Nachbedingung Erfolg:** Die Anforderungsattribute werden geändert.

**Nachbedingung Fehlschlag:** Die Änderungen werden verworfen.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer wählt mit der rechten Maustaste ein bestehendes Anforderungsattribut aus einer Liste aus und drückt in einem dadurch geöffneten Kontextmenü den Button zur Bearbeitung des Attributes.

### **Beschreibung:**

1. Das Attribut mit seinen Daten wird angezeigt.
2. Der Benutzer ändert die Daten des Attributes.
3. Der Benutzer kann die Eingabe bestätigen oder Abbrechen.
4. Bei Bestätigung werden die Änderungen übernommen.
5. Bei Abbruch werden die Änderungen verworfen.
6. Die Eingabefelder werden ausgeblendet

### **Alternativen:**

1. Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.
2. Bei Eingabefehlern wird nach F70 verfahren.

**Erweiterungen:** Keine.

## /F60/ (LF42)

**Geschäftsprozess:** Löschen von Anforderungsattributen.

**Ziel:** Das in der Liste ausgewählte Attribut wird gelöscht.

**Kategorie:** Primär

**Vorbedingung:** Es muss mindestens ein Anforderungsattribut vorhanden sein.

**Nachbedingung Erfolg:** Das gewählte Attribut wird aus der Liste gelöscht.

**Nachbedingung Fehlschlag:** Das Attribut bleibt in der Liste erhalten.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer wählt mit der rechten Maustaste ein bestehendes Anforderungsattribut aus einer Liste aus und drückt in einem dadurch geöffneten Kontextmenü den Button zum Löschen des Attributes.

**Beschreibung:** Das Attribut wird aus der Liste gelöscht.

**Alternativen:** Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.

**Erweiterungen:** Keine.

/F70/ (LF100)

**Geschäftsprozess:** Abfangen von Eingabefehlern

**Ziel:** Eingabefehler sollen abgefangen und zur manuellen Korrektur angezeigt werden.

**Kategorie:** Primär

**Vorbedingung:** Es muss ein Eingabefehler vorliegen.

**Nachbedingung Erfolg:** Bei dem entsprechenden Eingabefeld wird der Fehler angezeigt.

**Nachbedingung Fehlschlag:** Der Fehler wird nicht abgefangen.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer macht eine fehlerhafte Eingabe und bestätigt die Eingabe.

**Beschreibung:**

1. Die Eingabe wird nicht übernommen.
2. Anhand des Datentyps, des Kontexts, sowie der Länge der Eingabe werden Fehler von der Software erkannt.
3. Die Eingabefelder, die Fehler enthalten werden rot eingefärbt und der Fehler wird, in Form eines roten Hinweistextes, unterhalb des Eingabefelds angezeigt. Die Eingabe bleibt dabei erhalten.
4. Der Benutzer kann die Fehler enthaltenden Felder durch eine erneute Eingabe korrigieren.
5. Bei Bestätigung der Eingabe werden korrigierte Eingabefelder wieder normal eingefärbt und der Hinweistext entfernt.

**Alternativen:** Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.

**Erweiterungen:** Keine.

/F80/ (LF130)

**Geschäftsprozess:** Erstellen von Projekten

**Ziel:** Ein Projekt mit den Eigenschaften Projektname, Autor, Firma, Produkteinsatz, Glossar, Lines-Of-Code und Zielbestimmung wird erstellt.

**Kategorie:** Primär

**Vorbedingung:** Das Programm muss gestartet sein.

**Nachbedingung Erfolg:** Das Projekt wird angelegt.

**Nachbedingung Fehlschlag:** Das Projekt wird nicht angelegt und eine Fehlermeldung wird angezeigt.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer drückt einen Button in der Menüleiste zum Erstellen eines neuen Projekts.

**Beschreibung:**

1. Ein Dialog zur Erstellung von Projekten mit Eingabefeldern für Projektname, Autor, Firma, sowie Buttons für Glossar, Zielbestimmung und Produkteinsatz, wird angezeigt.
2. Bei Klick auf den Button Glossar wird ein Dialog mit einer leeren Tabelle zur Eingabe des Glossars angezeigt.
3. Bei Bestätigung des Dialogs zum Glossar wird das Glossar gespeichert, bei Abbruch verworfen.
4. Bei Klick auf den Button Zielbestimmung wird ein Dialog mit einem Textfeld für die Eingabe der Zielbestimmung angezeigt.
5. Bei Bestätigung des Dialogs zur Zielbestimmung wird die Zielbestimmung gespeichert, bei Abbruch verworfen.
6. Bei Klick auf den Button Produkteinsatz wird ein Dialog mit einem Textfeld für die Eingabe des Produkteinsatzes angezeigt.
7. Bei Bestätigung des Dialogs zum Produkteinsatz wird der Produkteinsatz gespeichert, bei Abbruch verworfen.
8. Bei Bestätigung des Dialogs zur Projekterstellung wird das Projekt mit den Eigenschaften erstellt.
9. Bei Abbruch wird das Projekt verworfen.

**Alternativen:**

1. Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.
2. Bei Eingabefehlern wird nach F70 verfahren.

**Erweiterungen:** Keine.

/F90/ (LF131)

**Geschäftsprozess:** Bearbeiten von Projekten

**Ziel:** Die Eigenschaften Projektname, Autor, Firma, Produkteinsatz, Glossar, Lines-Of-Code und Zielbestimmung eines Projektes werden bearbeitet.

**Kategorie:** Primär

**Vorbedingung:** Es muss ein Projekt vorhanden sein.

**Nachbedingung Erfolg:** Die geänderten Eigenschaften werden gespeichert.

**Nachbedingung Fehlschlag:** Die Änderungen werden verworfen.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer wählt mit der rechten Maustaste ein bestehendes Projekt aus einer Liste aus und drückt in einem dadurch geöffneten Kontextmenü den Button zur Bearbeitung des Projekts.

**Beschreibung:**

1. Ein Dialog zur Bearbeitung von Projekten mit Eingabefeldern für Projektname, Autor, Firma, sowie Buttons für Glossar, Zielbestimmung und Produkteinsatz, wird angezeigt.
2. Bei Klick auf den Button Glossar wird ein Dialog mit einer Tabelle, die das bisherige Glossar enthält, zur Änderung des Glossars angezeigt.
3. Bei Bestätigung des Dialogs zum Glossar wird das Glossar gespeichert, bei Abbruch verworfen.
4. Bei Klick auf den Button Zielbestimmung wird ein Dialog mit einem Textfeld für die Änderung der Zielbestimmung angezeigt.
5. Bei Bestätigung des Dialogs zur Zielbestimmung wird die Zielbestimmung gespeichert, bei Abbruch verworfen.
6. Bei Klick auf den Button Produkteinsatz wird ein Dialog mit einem Textfeld für die Änderung des Produkteinsatzes angezeigt.
7. Bei Bestätigung des Dialogs zum Produkteinsatz wird der Produkteinsatz gespeichert, bei Abbruch verworfen.
8. Bei Bestätigung des Dialogs zur Änderung des Projekts werden die Änderungen übernommen.
9. Bei Abbruch werden die Änderungen verworfen.

**Alternativen:**

1. Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.
2. Bei Eingabefehlern wird nach F70 verfahren.

**Erweiterungen:** Keine.

## /F100/ (LF132)

**Geschäftsprozess:** Löschen von Projekten

**Ziel:** Das Projekt wird gelöscht.

**Kategorie:** Primär

**Vorbedingung:** Es muss ein Projekt vorhanden sein.

**Nachbedingung Erfolg:** Das Projekt ist gelöscht.

**Nachbedingung Fehlschlag:** Das Projekt bleibt erhalten.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer wählt mit der rechten Maustaste ein bestehendes Projekt aus einer Liste aus und drückt in einem dadurch geöffneten Kontextmenü den Button zur Löschung des Projekts.

**Beschreibung:**

1. Es wird ein Dialog zur Bestätigung mit den Buttons „Ja“, sowie „Nein“, angezeigt.
2. Bei Bestätigung durch „Ja“ wird das Projekt aus der Liste im Programm gelöscht, die Projektdateien bleiben auf der Festplatte erhalten.
3. Bei Auswahl von „Nein“ wird der Löschvorgang abgebrochen und das Projekt bleibt erhalten.

**Alternativen:** Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.

**Erweiterungen:** Keine.

## /F110/ (LF30)

**Geschäftsprozess:** Durchführen einer Aufwandsschätzung

**Ziel:** Der Aufwand eines vorhandenen Projekts wird nach COCOMO oder Function-Point berechnet.

**Kategorie:** Primär

**Vorbedingung:**

1. COCOMO: Es muss ein Projekt mit angegebenen Lines-Of-Code vorhanden sein.
2. Function-Point: Es muss ein Projekt mit Anforderungen und Anforderungsattributen vorhanden sein.

**Nachbedingung Erfolg:** Der berechnete Aufwand des Projekts wird zum Projekt hinzugefügt.

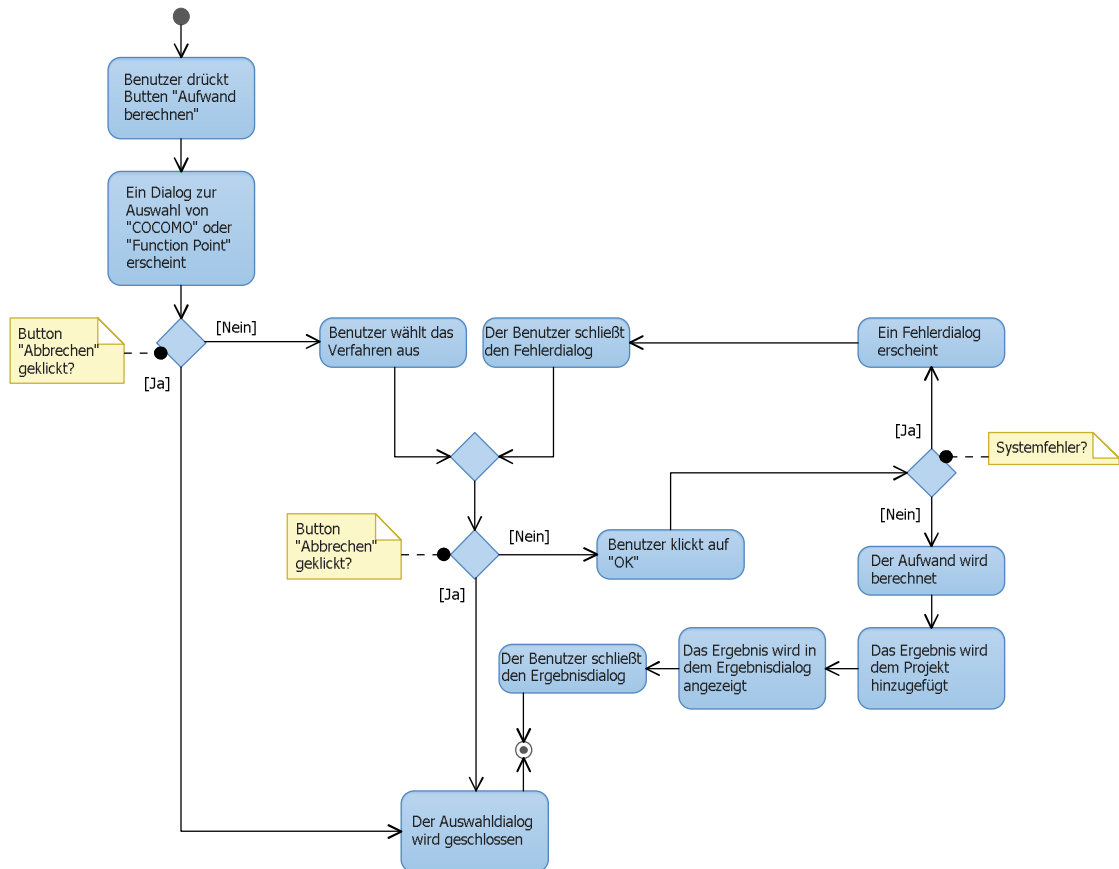
**Nachbedingung Fehlschlag:** Das Projekt bleibt erhalten.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer drückt den Button zur Berechnung des Aufwands in der Menüleiste.

**Beschreibung:**

1. Es wird ein Dialog zur Auswahl von COCOMO und/oder Function-Point angezeigt.
2. Der Benutzer wählt das Berechnungsverfahren und bestätigt die Wahl durch Klick auf den Button „OK“, oder bricht die Berechnung durch Klick auf den Button „Abbrechen“ ab.
3. Das Ergebnis wird berechnet.
4. Das Ergebnis wird zum Projekt hinzugefügt.



**Alternativen:** Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.

**Erweiterungen:** Keine.

/F120/(LF50)

**Geschäftsprozess:** Eingabe von tatsächlichen Werten zur Selbstoptimierung

**Ziel:** Der Benutzer gibt nach Abschluss eines Projekts die tatsächlichen Daten des Projekts ein.

**Kategorie:** Primär

**Vorbedingung:** Das Projekt muss im Programm vorhanden sein.

**Nachbedingung Erfolg:** Die eingegebenen Daten werden in einer Datenbank zur Selbstoptimierung gespeichert.

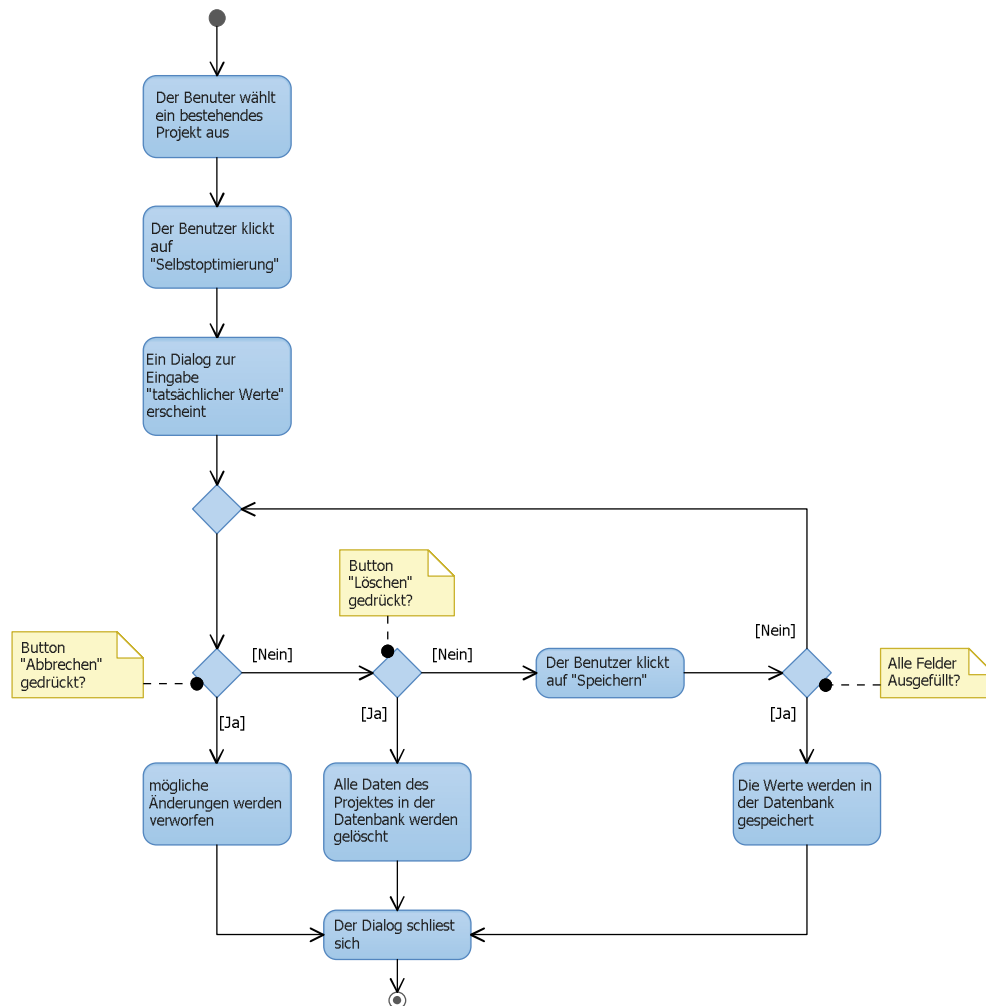
**Nachbedingung Fehlschlag:** Die eingegebenen Daten werden verworfen.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer wählt das Projekt in der Liste mit Rechtsklick aus und drückt den Button zur Selbstoptimierung im dadurch erscheinenden Kontextmenü.

**Beschreibung:**

1. Es wird ein Dialog zur Eingabe tatsächlicher Werte, des ausgewählten Projekts, angezeigt. Dazu gehören: Lines-Of-Code und Mann-Monate. Sind diese Werte bereits vorhanden, werden sie im Dialog angezeigt und können geändert werden.
2. Bei Klick auf „Speichern“ werden die vom Benutzer eingegebenen Daten in einer Datenbank gespeichert.
3. Bei Klick auf „Löschen“ werden die, zu dem Projekt gehörenden, Daten aus der Datenbank gelöscht.
4. Bei Klick auf „Abbrechen“ werden alle Änderungen verworfen.



#### Alternativen:

1. Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.
2. Bei Eingabefehlern wird nach F70 verfahren.

**Erweiterungen:** Keine.

/F130/(LF120)

**Geschäftsprozess:** Anpassung der Berechnung durch Datenbankwerte

**Ziel:** Die in F120 eingegeben Daten werden zur Selbstoptimierung der Aufwandsberechnung verwendet.

**Kategorie:** Primär

**Vorbedingung:** Die Datenbank zur Selbstoptimierung muss Werte von mindestens 20 abgeschlossenen Projekten enthalten.



**Nachbedingung Erfolg:** Die Berechnung wird angepasst.

**Nachbedingung Fehlschlag:** Die Berechnung bleibt unverändert.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer führt eine Aufwandsberechnung nach F110 durch.

**Beschreibung:** Die Berechnung wird, mithilfe der in der Datenbank enthaltenen Erfahrungswerte aus früheren Projekten, optimiert.

**Alternativen:** Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.

**Erweiterungen:** Keine.

/F140/ (LF90)

**Geschäftsprozess:** Export der Projektdaten im XML-Format

**Ziel:** Die Projektdaten werden im XML-Format exportiert und auf der Festplatte, an einem vom Benutzer auswählbaren Ort, gespeichert.

**Kategorie:** Primär

**Vorbedingung:** Es muss ein Projekt vorhanden sein.

**Nachbedingung Erfolg:** Das Projekt wird als XML-Datei am gewünschten Ort gespeichert.

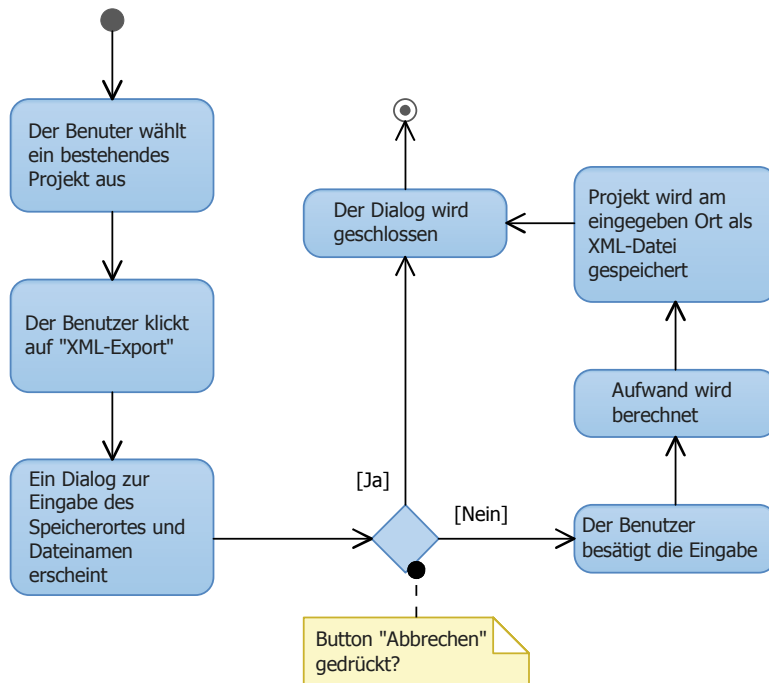
**Nachbedingung Fehlschlag:** Das Projekt wird nicht exportiert.

**Akteur:** Software, Benutzer

**Auslösendes Ereignis:** Der Benutzer wählt per linkem Mausklick ein Projekt in der Liste aus, wodurch das Projekt markiert wird. Um das ausgewählte Projekt zu exportieren, klickt er auf den Button „XML-Export“ in der Menüleiste.

**Beschreibung:**

1. Ein Dialog zur Auswahl des Ortes, sowie des Namens, der XML-Datei wird angezeigt.
2. Der Benutzer wählt einen Ort und Namen.
3. Bestätigt der Nutzer durch einen Klick auf „OK“, wird der Aufwand des Projekts neu berechnet und dem Projekt hinzugefügt. Das Projekt wird am angegebenen Ort als XML-Datei exportiert.
4. Bricht der Benutzer durch Klick auf „Abbruch“ die Eingabe ab, wird das Projekt nicht exportiert.



#### Alternativen:

1. Bei Systemfehlern nach Bestätigung wird eine Fehlermeldung angezeigt. Die Eingabe bleibt erhalten, wird aber nicht übernommen. Die Fehlermeldung kann vom Benutzer geschlossen werden.
2. Ist der Pfad schon vorhanden, wird ein Dialog zur Bestätigung angezeigt. Bei Bestätigung wird die existierende Datei überschrieben.
3. Bei Eingabefehlern wird nach F70 verfahren.

**Erweiterungen:** Keine.

## 5. PRODUKT-DATEN

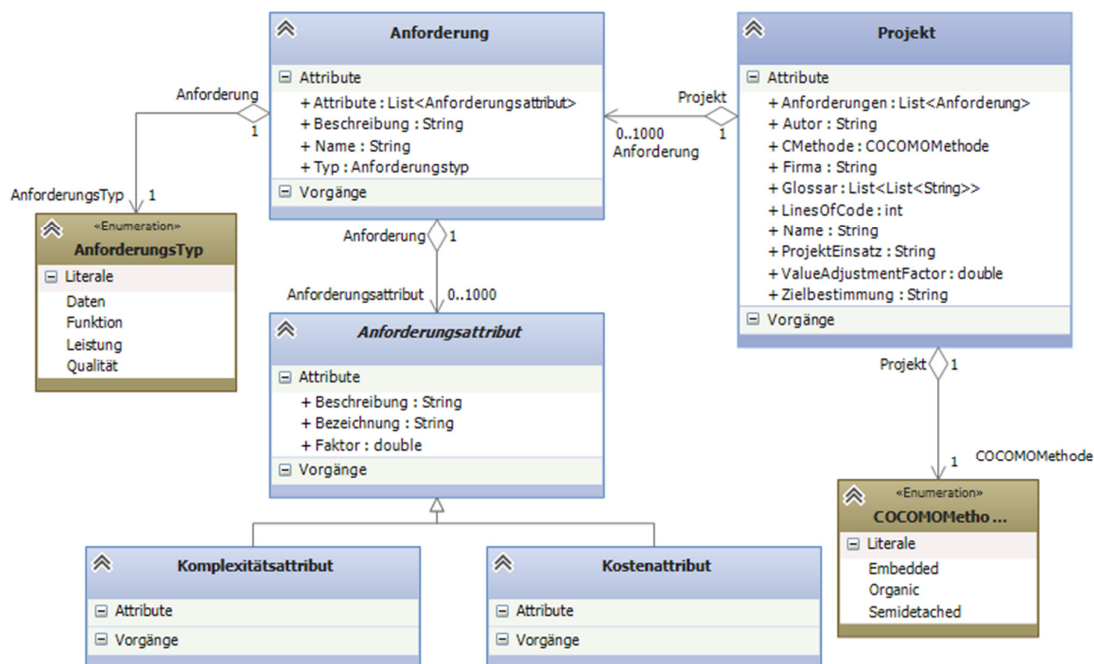


Abbildung 1: Klassendiagramm der gespeicherten Daten

### /D10/ (LD20) PROJEKTDATEN

| Attribut                     | Erklärung  |
|------------------------------|--|
| <b>Anforderungen</b>         | Liste der Anforderungen des Projekts                       |
| <b>Autor</b>                 | Name des Autors  |
| <b>CMethod</b>               | COCOMO-Methode   |
| <b>Firma</b>                 | Firmenname   |
| <b>Glossar</b>               | Das Glossar als zweidimensionale Liste                     |
| <b>Lines-Of-Code</b>         | Anzahl des Codezeilen                                      |
| <b>Name</b>                  | Name des Projekts  |
| <b>Projekteinsatz</b>        | Der Projekteinsatz als Text                                |
| <b>ValueAdjustmentFactor</b> | Der Value Adjustment Factor für die Function-Point Methode |
| <b>Zielbestimmung</b>        | Die Zielbestimmung des Projekts als Text                   |

Es sind bis zu 1000 Datensätze speicherbar.

### /D20/ (LD10) ANFORDERUNGSDATEN

| Attribut            | Erklärung   |
|---------------------|---|
| <b>Attribute</b>    | Liste der Attribute der Anforderung                                       |
| <b>Beschreibung</b> | Die Beschreibung der Anforderung als Text                                 |
| <b>Name</b>         | Der Name der Anforderung  |
| <b>Typ</b>          | Der Typ der Anforderung. Möglich: Daten, Funktion, Leistung oder Qualität |

Es sind bis zu 1000 Datensätze speicherbar.

## /D30/ (LD30) ANFORDERUNGSATTRIBUTE

| Attribut            | Erklärung  |
|---------------------|--|
| <b>Bezeichnung</b>  | Die Bezeichnung des Attributs als Text                 |
| <b>Beschreibung</b> | Die Beschreibung des Attributs als Text                |
| <b>Faktor</b>       | Faktor des Attributs für die Function-Point Berechnung |

Es sind bis zu 1000 Datensätze speicherbar.

## /D40/ DATENBANK ZUR SELBSTOPTIMIERUNG

In der Datenbank zur Selbstoptimierung werden die Projekte nach D10-D30 im XML-Format gespeichert und mit dem tatsächlichen Aufwand verknüpft.

Es sind bis zu 1000 Datensätze speicherbar.

## 6. PRODUKT-LEISTUNGEN

### /L10/ (LL10)

Die Software muss in 99% aller Fälle stabil bleiben. Dies bedeutet, sie darf nicht abstürzen.

### /L20/ (LL20)

Berechnungen dürfen höchstens 5 Sekunden dauern. (COCOMO/Function-Point)

### /L30/ (LL30)

L10 und L20 sollen auf aktuellen Systemen (Siehe Pflichtenheft: Kapitel 3) gewährleistet sein.

## 7. QUALITÄTS-ZIELBESTIMMUNG

| Produktqualität        |                       | Sehr gut | Gut | Normal | Irrelevant |
|------------------------|-----------------------|----------|-----|--------|------------|
| <b>Funktionalität</b>  | Angemessenheit        |          |     | X      |            |
|                        | Richtigkeit           |          | X   |        |            |
|                        | Interoperabilität     |          | X   |        |            |
|                        | Ordnungsmäßigkeit     |          | X   |        |            |
|                        | Sicherheit            |          |     | X      |            |
| <b>Zuverlässigkeit</b> | Reife                 |          | X   |        |            |
|                        | Fehlertoleranz        | X        |     |        |            |
|                        | Wiederherstellbarkeit |          | X   |        |            |
| <b>Benutzbarkeit</b>   | Verständlichkeit      | X        |     |        |            |
|                        | Erlernbarkeit         |          | X   |        |            |
|                        | Bedienbarkeit         | X        |     |        |            |
| <b>Effizienz</b>       | Zeitverhalten         |          |     | X      |            |
|                        | Verbraucherverhalten  |          | X   |        |            |

|                        |                   |   |   |   |  |
|------------------------|-------------------|---|---|---|--|
| <b>Änderbarkeit</b>    | Analysierbarkeit  |   | X |   |  |
|                        | Modifizierbarkeit |   |   | X |  |
|                        | Stabilität        | X |   |   |  |
|                        | Prüfbarkeit       |   | X |   |  |
| <b>Übertragbarkeit</b> | Anpassbarkeit     |   |   | X |  |
|                        | Installierbarkeit |   | X |   |  |
|                        | Konformität       |   | X |   |  |
|                        | Austauschbarkeit  |   |   | X |  |

## 8. TESTSZENARIEN

/T10/

Der Projektleiter Hans Wurst erstellt ein Projekt (F80) und gibt dabei folgende Attribute an:

| Attribut                     | Wert                     |
|------------------------------|--------------------------|
| <b>Autor</b>                 | Hans Wurst               |
| <b>CMethode</b>              | Organic                  |
| <b>Firma</b>                 | Hans' Wurstfabrik        |
| <b>Glossar</b>               | Beliebige Testwerte      |
| <b>Lines-Of-Code</b>         | 1337                     |
| <b>Name</b>                  | Leberwurst               |
| <b>Projekteinsatz</b>        | Wurstbrot                |
| <b>ValueAdjustmentFactor</b> | 1.2                      |
| <b>Zielbestimmung</b>        | Wurstbrot mit Leberwurst |

/T20/

Der Projektleiter Hans Wurst bearbeitet die Eigenschaften (F90) des nach T10 angelegten Projekts folgendermaßen:

| Attribut                     | Wert                  |
|------------------------------|-----------------------|
| <b>Autor</b>                 | Hans Bratwurst        |
| <b>CMethode</b>              | Semidetached          |
| <b>Firma</b>                 | Hans' Bratwurstfabrik |
| <b>Glossar</b>               | Beliebige Testwerte   |
| <b>Lines-Of-Code</b>         | 2668                  |
| <b>Name</b>                  | Bratwurst             |
| <b>Projekteinsatz</b>        | Würstchenbude         |
| <b>ValueAdjustmentFactor</b> | 1.3                   |
| <b>Zielbestimmung</b>        | Optimale Bratwurst    |

/T30/

Der Projektleiter Hans Wurst löscht das in T10 angelegte Projekt (F100).

/T40/

Der Projektleiter erstellt ein neues Projekt mit den Daten aus T10. Er fügt ihm folgende Anforderung hinzu (F10):

| Attribut     | Erklärung                                |
|--------------|--|
| Beschreibung | Der Geschmack der Wurst ist optimal sein |
| Name         | Geschmack                                |
| Typ          | Qualität                                 |

/T50/

Der Projektleiter ändert die Anforderung aus T40 folgendermaßen (F20):

| Attribut     | Erklärung               |
|--------------|-------------------------|
| Beschreibung | Die Wurst wird gegessen |
| Name         | Verzehr                 |
| Typ          | Funktion                |

/T60/

Der Projektleiter erstellt nach T10 ein Projekt und fügt ihm nach T40 eine Anforderung hinzu. Er löscht diese Anforderung (F30).

/T70/

Der Projektleiter verfährt nach T40. Dieser Anforderung fügt er folgendes Attribut hinzu (F40):

| Attribut     | Erklärung        |
|--------------|------------------|
| Bezeichnung  | Testbezeichnung  |
| Beschreibung | Testbeschreibung |
| Faktor       | 10               |

/T80/

Der Projektleiter verfährt nach T70 und bearbeitet das Attribut (F50):

| Attribut     | Erklärung |
|--------------|-----------|
| Bezeichnung  | Test      |
| Beschreibung | Test2     |
| Faktor       | 0.5       |

/T90/

Der Projektleiter verfährt nach T70 und löscht das Attribut (F60).

### /T100/

Der Projektleiter verfährt nach T10, T20, T40, T50, T70 und T80 und versucht jeweils fehlerhafte Werte einzugeben (F70).

### /T110/

Der Projektleiter verfährt nach T10, T40 und T70. Er führt drei Aufwandsschätzungen durch, eine nach COCOMO, eine nach Function-Point und eine nach beiden Methoden (F110).

### /T120/

Der Projektleiter verfährt nach T110, führt jedoch nur eine Berechnung durch, und gibt tatsächliche Werte zur Selbstoptimierung ein (F120).

### /T130/

1. Der Projektleiter legt nach T70 bei leerer Datenbank ein Projekt an und führt eine Aufwandsschätzung nach beiden Methoden durch. Er notiert die Ergebnisse.
2. Der Projektleiter legt nach T70 mehrere Projekte mit verschiedenen Daten an und gibt für jedes Projekt nach T120 tatsächliche Werte ein.
3. Der Projektleiter legt ein Projekt mit denselben Daten wie in 1. an und führt erneut eine Berechnung nach beiden Methoden durch.
4. Er vergleicht die Ergebnisse mit den notierten Werten. Sie sollten sich unterscheiden (F130).

### /T140/

Der Projektleiter legt nach T70 ein Projekt an und exportiert es als XML-Datei (F140).

## 9. BENUTZEROBERFLÄCHE

Die Benutzeroberfläche ist nicht Teil dieses Projekts.

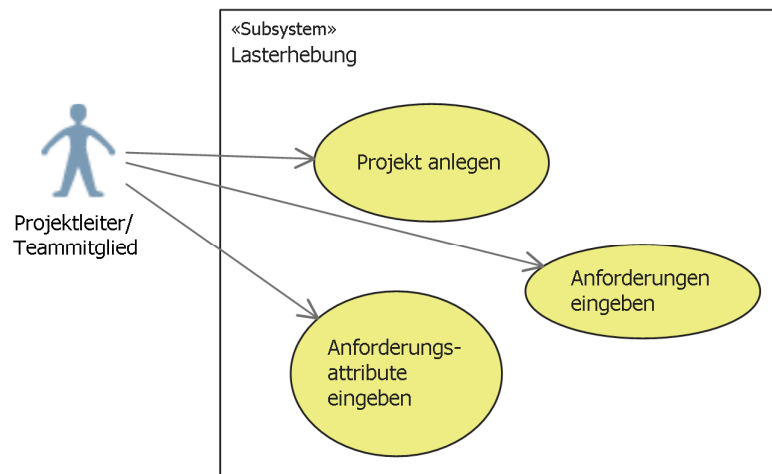
## 10. ENTWICKLUNGS-UMGEBUNG

| Umgebung | Spezifikationen  |
|----------|--|
| Software | Betriebssysteme: <ul style="list-style-type: none"><li>▪ Windows XP oder neuer</li><li>▪ Linux 2 oder höher</li><li>▪ Mac OSX Version 10.6 oder höher</li></ul> Java JDK 6 oder höher<br>Eclipse 4 |
| Hardware | x64 oder x86 Prozessor<br>Mindestens 2GB Hauptspeicher<br>Grafikchip<br>Bildschirm<br>Tastatur<br>Maus<br>Netzwerkverbindung   |
| Orgware  | Datenbankanbindung   |

## 11. ERGÄNZUNGEN

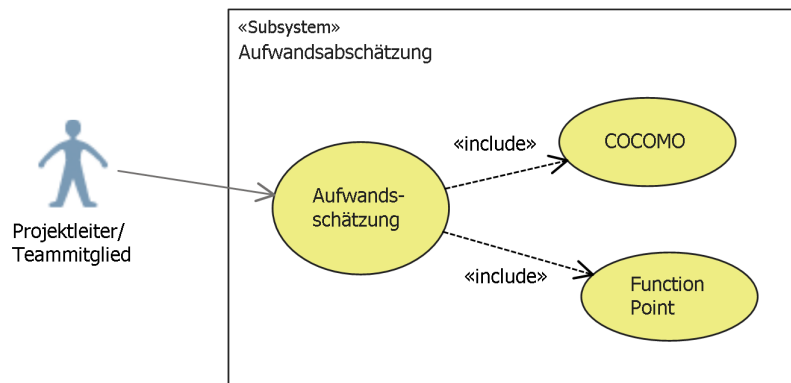
### USE-CASE-DIAGRAMME

#### Lasterhebung

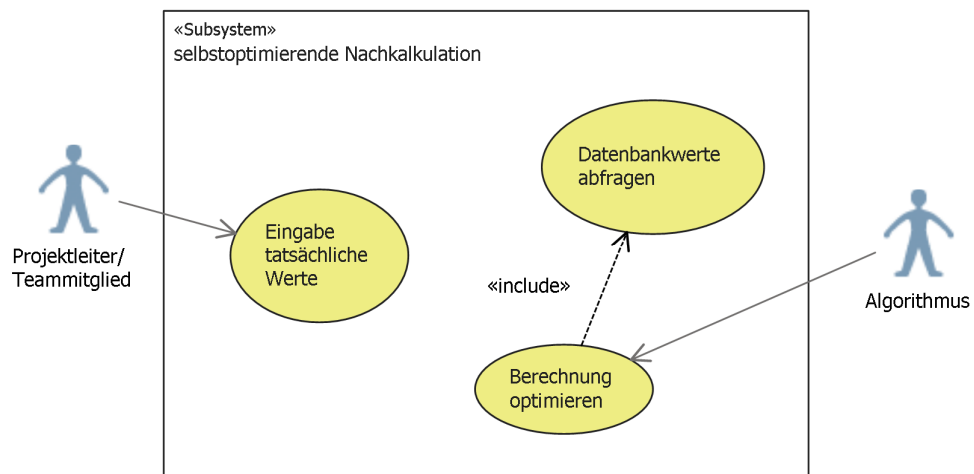




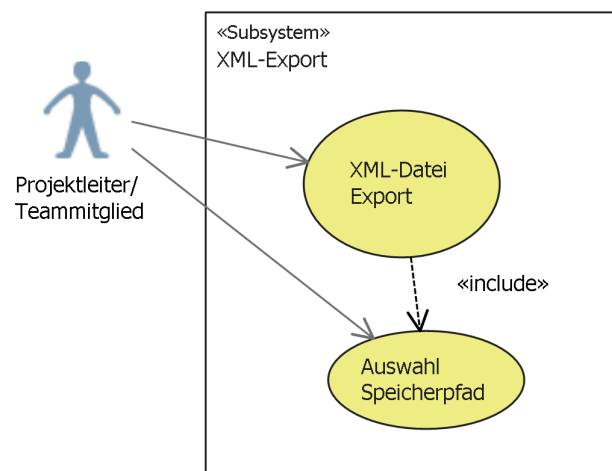
## Aufwandsabschätzung



## Selbstoptimierende Nachkalkulation



## XML-Export



## 13. GLOSSAR

| Begriff                     |                         | Erklärung  |
|-----------------------------|-------------------------|--|
| <b>Anforderung</b>          |                         | ein Erfordernis oder eine Erwartung, das oder die festgelegt, vorausgesetzt oder verpflichtend ist   |
| <b>Anforderungsattribut</b> |                         | Attribut einer Anforderung   |
| <b>Aufwand</b>              |                         | Aufwand eines Projekts, üblicherweise angegeben in Mannmonaten   |
| <b>Betriebssystem</b>       |                         | eine Sammlung von Computerprogrammen, die die Systemressourcen eines Computers wie Arbeitsspeicher, Festplatten, Ein- und Ausgabegeräte verwaltet und diese Anwendungsprogrammen zur Verfügung stellt          |
| <b>Button</b>               |                         | Button bezeichnet ein häufig verwendetes Element grafischer Benutzeroberflächen in einem Anwendungsprogramm. Die Schaltfläche ermöglicht dem Benutzer, eine dem Steuerelement zugeordnete Funktion auszulösen. |
| <b>COCOMO</b>               | Constructive Cost Model | Kostenmodell, das Aufwand mit Hilfe von Lines-Of-Code berechnet.   |
| <b>Datenbank</b>            |                         | System zur elektronischen Datenverwaltung  |
| <b>Desktop-PC</b>           |                         | ein Computer in einer Gehäuseform passend für den Einsatz als Arbeitsplatzrechner auf Schreibtischen   |
| <b>Dialog</b>               |                         | Fenster, in dem es möglich ist Informationen einzugeben oder dem Computer Befehle zu erteilen  |
| <b>Eclipse</b>              |                         | Plattformunabhängige Entwicklungsumgebung für Software   |
| <b>Eingabefehler</b>        |                         | Fehler, die durch die Eingabe des Benutzers entstehen  |
| <b>Fenster</b>              |                         | Ein Fenster ist ein (meist rechteckiger) Bestandteil einer grafischen Benutzerschnittstelle  |
| <b>Function-Point</b>       |                         | Kostenmodell, das Aufwand mit Hilfe von Anforderungen berechnet.   |
| <b>Grafikchip</b>           | Grafikprozessor         | dient zur Berechnung der Bildschirmausgabe auf Computern   |
| <b>Hardware</b>             |                         | Oberbegriff für die mechanische und elektronische Ausrüstung eines Systems   |
| <b>Hauptspeicher</b>        |                         | Bezeichnung für den Speicher, der die gerade auszuführenden Programme oder Programmteile und die benötigten Daten enthält  |
| <b>IDE</b>                  | Integrated Development  | Sammlung von Anwendungen für die   |

|                      | Environment<br>(Entwicklungsumgebung)    | Softwareentwicklung innerhalb einer Programmumgebung  |
|----------------------|--|---|
| <b>Kontextmenü</b>   |  | ein Kontextmenü ein Steuerelement, das dem Benutzer zu einem bestimmten Kontext verschiedene Aktionen zur Auswahl anbietet.   |
| <b>Liste</b>         |  | In der Programmierung gilt als 'Liste' eine verkettete Folge von Elementen eines gegebenen Datentyps  |
| <b>LOC</b>           | Lines-Of-Code                            | Zeilen Programmcode ohne Kommentare und Leerzeilen  |
| <b>Mannmonat</b>     |  | Einheit des Aufwands, Produkt aus benötigter Zeit und Personaleinheiten   |
| <b>Orgware</b>       |  | Rahmenbedingungen bei IT-Projekten, die weder zum Bereich Hardware noch Software zählen   |
| <b>OSLC</b>          | Open Services for Lifecycle Colaboration | Offene Spezifikation für Managementsoftware für Produktlebenszyklen   |
| <b>Plugin</b>        |  | Nachträglich installierbare Funktionserweiterung einer Software   |
| <b>Projektleiter</b> |  | ist für die operative Planung und Steuerung des Projektes verantwortlich  |
| <b>Prozessor</b>     |  | Maschine oder eine elektronische Schaltung, welche gemäß übergebener Befehle andere Maschinen oder elektrische Schaltungen steuert und dabei einen Prozess oder Algorithmus vorantreibt, was meist Datenverarbeitung beinhaltet |
| <b>Server</b>        |  | Computer, auf dem ein oder mehrere Programme, die mit dem Client kommunizieren um ihm Dienste zur Verfügung zu stellen  |
| <b>Software</b>      |  | Sammelbegriff für ausführbare Programme und die zugehörigen Daten   |
| <b>Systemfehler</b>  |  | Fehler, die ohne Zutun des Benutzers im System entstehen  |
| <b>X64</b>           |  | Erweiterung der IA-32-Architektur um die Fähigkeit, direkt mehr als 4 GiB Speicher zu adressieren und die 64-Bit-AMD64-Befehle auszuführen.   |
| <b>X86</b>           | IA-32-Architektur                        | Abkürzung einer Mikroprozessor-Architektur und der damit verbundenen Befehlssätze, welche unter anderem vom Chip-Hersteller Intel entwickelt werden.  |
| <b>XML</b>           | Extensible Markup Language               | Sprache zur hierarchischen Darstellung strukturierter Daten in Textform   |