**Architekturdokumentation**

<Mängel-Manager>

erstellt von

*Template Revision: 6.0 DE (Release Candidate)  
19. März 2012*

|  |  |
| --- | --- |
| We acknowledge that this document uses material from the arc 42 architecture  template, <http://www.arc42.de>. Created by Dr. Peter Hruschka & Dr. Gernot Starke. For additional contributors see arc42.de/about/contributors.html |  |

**Änderungsübersicht**

| **Version** | **Datum** | **Bearbeiter** | **Beschreibung** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Basisdokumente**

|  |  |
| --- | --- |
| **Dokument** | **Beschreibung** |
|  |  |
|  |  |

Inhaltsverzeichnis

1. Einführung und Ziele 7

1.1 Aufgabenstellung 8

1.2 Qualitätsziele 11

1.3 Stakeholder 11

2. Randbedingungen 13

2.1 Technische Randbedingungen 13

2.2 Organisatorische Randbedingungen 14

2.3 Konventionen 15

3. Kontextabgrenzung 15

3.1 Fachlicher Kontext 15

3.2 Technischer- oder Verteilungskontext 15

3.3 Externe Schnittstellen 16

4. Lösungsstrategie 18

5. Bausteinsicht 18

5.1 Ebene 1 18

5.2 Ebene 2 19

5.3 Ebene 3 22

6. Laufzeitsicht 22

6.1 Laufzeitszenario 1 22

6.2 Laufzeitszenario 2 22

6.3 ... 22

6.4 Laufzeitszenario n 22

7. Verteilungssicht 22

7.1 Infrastruktur Ebene 1 23

7.2 Infrastruktur Ebene 2 23

8. Konzepte 23

8.1 Fachliche Strukturen und Modelle 23

8.2 Typische Muster und Strukturen 23

8.3 Persistenz 23

8.4 Benutzungsoberfläche 23

8.5 Ergonomie 23

8.6 Ablaufsteuerung 24

8.7 Transaktionsbehandlung 24

8.8 Sessionbehandlung 24

8.9 Sicherheit 24

8.10 Kommunikation und Integration mit anderen IT-Systemen 24

8.11 Verteilung 24

8.12 Plausibilisierung und Validierung 24

8.13 Ausnahme-/Fehlerbehandlung 24

8.14 Management des Systems & Administrierbarkeit 24

8.15 Logging, Protokollierung, Tracing 24

8.16 Geschäftsregeln 24

8.17 Konfigurierbarkeit 24

8.18 Parallelisierung und Threading 24

8.19 Internationalisierung 25

8.20 Migration 25

8.21 Testbarkeit 25

8.22 Skalierung, Clustering 25

8.23 Hochverfügbarkeit 25

9. Entwurfsentscheidungen 25

9.1 Entwurfsentscheidung 1 25

9.2 Entwurfsentscheidung n 25

10. Qualitätsszenarien 25

10.1 Qualitätsbaum 25

10.2 Bewertungsszenarien 26

11. Risiken 26

12. Glossar 26

13. Requirements Dokumentation (2015\_FS\_INM21\_Requirements\_MangelManager.xls) 26

14. UseCase Dokumentation 27

14.1 UseCase001 27

14.2 UseCase002 33

14.3 UseCase003 37

14.4 UseCase004 40

14.5 UseCase005 45

14.6 UseCase006 48

50

14.7 UseCase007 51

14.8 UseCase008 53

15. Klassendiagramme 53

16. Deyploment-Infos 53

17. TDD und JUnit 53

18. Funktionale Test’s 53

19. DB – Dokumentation 54

20. Beiträge pro Projektmitglied (2015\_FS\_INM21\_Individuelles\_Portfolio\_Name\_Vorname.xlsx) 54

21. Source-Code von selber beschriebenem Code 54

22. Weitere Dokumentationen 54

# Einführung und Ziele

Die „W&W Architekten GmbH“ besteht seit über zehn Jahren und beschäftigt bereits mehr als 30 Mitarbeiter. Die Generalunternehmung (im weiteren Text GU) „W&W Architekten GmbH“ ist beim Bau von Ein- und Mehrfamilienhäusern tätig. Es werden Bauprojekte realisiert, wobei die GU die Zentralfigur ist. Die GU wird ausschliesslich vom Bauherrn informiert und diese wiederrum verständigt weitere Firmen (Subunternehmen) mit den noch zu erledigenden Arbeiten (z.B. Elektriker, Maler, Küchenbauer, Sanitär, Gartenbauer etc.).

###### IT Unterstützung

Momentan haben die GU Mitarbeiter, welche direkt mit der Bautätigkeit zu tun haben, eine eher spärliche IT Umgebung. Die Bauleiter beschweren sich, dass die Übersicht mit der Zunahme von Arbeitsvolumen nachlässt. Das Risiko, dass essentielle Informationen in Vergessenheit geraten steigt kontinuierlich. Aufgrund dieses Risikos muss nach einer Lösung gesucht werden, welche als Unterstützung der Bauleiter dienen soll um die anfallenden Kontrollen effizient und zuverlässig durchzuführen.

###### Ablauf

Wenn ein Bauleiter auf einer Baustelle eine Kontrolle durchführt, kontrolliert er, ob die erledigten Arbeiten korrekt ausgeführt wurden oder nicht. Sollte er dabei auf Fehler bei den ausgeführten Arbeiten stossen, werden diese als Mängel erfasst. Bei jedem Mangel wird erfasst,

• um was für einen Mangel es sich handelt (eine kurze Beschreibung),

• welches Subunternehmen für die Arbeit und somit für Behebung des Mangels zuständig ist,

• wann der Mangel festgestellt wurde (Datum) und

• bis wann der Mangel behoben werden sollte (Datum).

Die Angabe, bis wann der Mangel behoben werden muss, ist sehr wichtig, da andere Arbeiten unter Umständen erst nach der Behebung des Mangels ausgeführt werden können.

Die Erfassung von Mängel wird auf dem Papier gemacht und am Abend, wenn der Bauleiter ins Büro zurückkehrt, in eine Excel-Arbeitsmappe eingetragen (pro Projekt wird eine Excel-Arbeitsmappe angelegt). Anschliessend wird ein PDF generiert und um den Aufwand im Rahmen zu halten, an alle betroffene Subunternehmen gesendet. Die Rede ist von einem Mängelrapport (im weiteren Text MR). Das kommt bei einigen Subunternehmen nicht gut an. Sie halten es für unangebracht, dass andere Subunternehmen den Einblick in Ihre bemängelten Arbeiten bekommen. Die Bauleiter weigern sich aber weiterhin, für jedes betroffene Subunternehmen ein separates MR zu machen. Sie begründen es mit dem unverhältnismässigen Aufwand den sie betreiben müssten.

Wenn ein Subunternehmen (im weiteren Text SU) den MR bekommt, muss für die bemängelte Arbeit zuständige Person in jenem SU darüber informiert werden. Idealerweise per Telefon (in dringenden Fällen) oder am nächsten Morgen, wenn alle Mitarbeiter des SU wieder in der Werkstatt sind. Daraufhin kann der zuständige Mitarbeiter des SU reagieren:

• Es kann sein, dass es schlicht und einfach ein Fehler ist, der behoben wird.

• Es kann auch sein, dass es für diesen Fehler nachvollziehbare Gründe gibt und auf die Behebung des vermeintlichen Fehlers evtl. verzichtet werden kann.

• Es kann auch sein, dass der vermeintliche Fehler gar kein Fehler ist und die bemängelte Arbeit aus bestimmten Gründen explizit so ausgeführt wurde.

Wenn dies geklärt ist, wird der Bauleiter vom GU entsprechend informiert und je nach dem, was der Grund für den Mangel ist, kann sein, dass es eine weitere Klärungsrunde geben muss usw. Das SU sendet an den Bauleiter ein Email, in dem alles was nötig ist, angegeben wird. Beispielsweise könnte dies wie folgt aussehen:

• Der Mangel Nr. X wird wie von Ihnen gewünscht bis zum angegebenen Datum behoben.

• Der Mangel Nr. Y kann nicht so behoben werden, wie Sie es gewünscht haben. Dazu braucht es eine Abklärung vor Ort. Bitte geben Sie uns an, wann Sie wieder vor Ort sein werden.

• Der Mangel Z ist gar kein Mangel. Die Ausführung wurde so explizit von Ihrem Chef-Architekt (z.B. Hr. Max Mustermann) gewünscht. Bitte klären Sie es mit Herrn Mustermann ab.

Wenn ein Mangel definitiv behoben ist, wird der Bauleiter benachrichtigt, damit er es noch abschliessend kontrollieren kann. Nach der abschliessenden Kontrolle wird der Mangel von Bauleiter als erledigt markiert, solange die Behebung zufriedenstellend war. Sollte das nicht der Fall sein, wendet sich der Bauleiter erneut an das zuständige SU usw. Dieser Vorgang kann bei Bedarf mehrfach wiederholt werden. Am Schluss müssen alle Mängel behoben und in der entsprechenden Excel-Arbeitsmappe als solche markiert sein.

Die Kommunikation in diesem Kontext ist etwas umständlich und bis ein Fall geklärt wird, kann unter Umständen lange dauern, da mehrere "Runden" benötigt werden. Um das Problem zu entschärfen, soll eine Applikation entwickelt werden, welche das Management von Baumängeln in einer einfacheren Art und Weise ermöglichen soll.

## Aufgabenstellung

Nachfolgend werden die Anforderungen an die Applikation so gut wie zurzeit möglich beschrieben. Mit weiteren Ergänzungen muss gerechnet werden, Verbesserungsvorschläge sind willkommen.

**Verwaltung von Stammdaten**

Die zu erstellende Applikation muss mindestens folgendes ermöglichen (Erweiterungen sind erlaubt):

1. die Verwaltung aller nötigen Daten von Subunternehmen, welche für die "W&W Architekten GmbH" im Auftragsverhältnis Arbeiten ausführen.
2. die Verwaltung aller nötigen Daten von Bauherren, für welche ein Bau ausgeführt bzw. ein Projekt realisiert wird. Dabei ist zu berücksichtigen, dass in Rolle des Bauherrn sowohl eine Person als auch eine Firma auftreten kann.
3. die Verwaltung aller Bauprojekte, welche von der GU realisiert werden. Für jedes Bauprojekt muss es schnell ersichtlich sein:

* um was für ein Projekt es sich handelt (Einfamilienhaus, Mehrfamilienhaus, Wohnung, Garage etc.),
* um was für eine Arbeit es sich handelt (Neu- oder Umbau, Renovation, Teil-Renovation),
* in welcher Zeitperiode das Projekt realisiert werden soll (Start / Ende),
* an welcher Adresse sich das neugebaute / umgebaute Objekt befindet,
* wer der Bauherr ist,
* welcher Bauleiter für das Projekt bzw. Bauobjekt seitens GU zu welcher Zeit zuständig ist,
* welche Subunternehmen in dem Projekt involviert sind und
* wer zu welcher Zeit die Ansprechperson für das jeweilige Subunternehmen in den einzelnen Projekten ist.

Die obige Aufzählung ist nicht abschliessend und kann bzw. soll im Bedarfsfall ergänzt werden.

###### Verwaltung von Mängeldaten

Die zu erstellende Applikation muss dem Bauleiter ermöglichen, die während einer Kontrolle bzw. einer Besichtigung der Baustelle gefundenen Mängel vor Ort zu erfassen und die erfassten Daten direkt auf dem Server der GU zu speichern. Das heisst, dass die Daten zum Server übertragen und da von einer dafür zuständigen Komponente in Empfang genommen und weiter verarbeitet werden.

###### Benutzerschnittstelle (User Interface)

Die Applikation muss drei Benutzerschnittstellen zur Verfügung stellen:

1. Die Benutzerschnittstelle für BackOffice (BO-UI) wird im Intranet von administrativen Mitarbeitern benutzt und stellt einem Benutzer alle Funktionalitäten zur Verfügung, welche für die Verwaltung von Daten (Stammdaten und Mängeldaten) benötigt werden.
2. Die Schnittstelle für Bauleiter (BL-UI) wird im Extranet von Bauleiter benutzt. Sie soll einem Bauleiter den Zugriff auf alle für ihn relevanten Daten zu einem beliebigen Zeitpunkt ermöglichen. Des Weiteren soll sie auch die Erfassung von während einer Kontrolle gefundenen Mängel vor Ort ermöglichen. Alle Bauleiter werden mit Notebooks ober Tabletts ausgerüstet. Ausserdem wird dafür gesorgt, dass alle Bauleiter immer online sind und auf die IT Infrastruktur der GU (Website der GU, Server etc.) jederzeit zugreifen können.
3. Die Subunternehmen-Schnittstelle (SU-UI) soll den Zugriff auf die für ein Subunternehmen relevanten Daten ermöglichen. Dazu gehören die eigenen Stammdaten und die für das Subunter-nehmen bestimmten Mängeldaten.

###### Zugang und Datenablage

Der Zugriff auf die Applikation und Daten muss in allen Fällen durch ein entsprechendes Anmelde-Verfahren geschützt werden. Die Applikation wird einerseits von der GU Mitarbeiter (BackOffice und Bauleiter) und andererseits von den Mitarbeiter von Subunternehmen benutzt. Dabei muss auch sichergestellt werden, dass die unterschiedlichen Benutzer nur auf jene Daten zugreifen können, für die sie berechtigt sind. Die Benutzerschnittstelle für Subunternehmen (SU-UI) wird, sobald verfügbar, im Downloadbereich der GU allen Subunternehmen zum Downloaden zur Verfügung gestellt.

Die Verwaltung der anfallenden Daten muss mit einem zuverlässigen DBMS (z.B. PostgreSQL, MySQL, Oracle-DB, SQL-Server etc.) realisiert werden. Dabei muss sichergestellt werden, dass der verwendete DBMS problemlos und ohne zusätzliche Aufwände ausgetauscht werden kann (Bindung an ein bestimmtes DBMS ist nicht erlaubt).

###### Diverses

Nachfolgend werden noch weitere Anforderungen aufgeführt:

* Für ein Projekt (Objekt) ist seitens GU i.d.R. ein Bauleiter zuständig. Es kann aber sein, dass ein Projekt vorübergehend von einem anderen Bauleiter betreut wird (z.B. Ferienabwesenheit). Es muss aber auch möglich sein, dass ein Projekt von einem anderen Bauleiter vollständig übernommen wird (z.B. wenn ein Bauleiter die GU verlässt oder andere Projekte übernimmt). Sie müssen beim Bauleiter also eine Art "Buch führen", wann welcher Bauleiter welche Mängel erfasst hat und zuständig war.
* Ein Subunternehmen kann in mehreren Projekten Aufträge übernehmen. Für jedes Projekt, in dem das Subunternehmen mitarbeitet, muss eine Ansprechperson seitens Subunternehmen bestimmt werden. Je nach Grösse des Subunternehmens und Umfang der auszuführenden Arbeiten kann eine Ansprechperson für alle Projekte oder auch unterschiedliche Ansprechpersonen für unterschiedliche Projekte bestimmt werden.
* Ein Bauleiter der GU darf nur jene Daten verwalten können, die sich auf ihm zugeordnete Projekte beziehen, das heisst, von ihm geleitet werden. Die Einsicht in Projekte, die von andern Bauleiter geleitet werden, darf ein Bauleiter nicht haben.
* Die Zuweisung, welcher Bauleiter welches Projekt leitet, wird durch die Administration des GU vorgenommen (Sachbearbeiter / Administrator).
* Seitens Subunternehmen ist es auch so, dass eine Ansprechperson nur jene Projektdaten sehen darf, für die sie zuständig ist. Die Zuweisung einer Ansprechperson einem bestimmten Projekt wird durch den Administrator (Sachbearbeiter) des Subunternehmens vorgenommen.

## Qualitätsziele

|  |  |
| --- | --- |
| Qualitätsmerkmal | Ziel |
| Änderbarkeit | Es muss sichergestellt werden, dass der verwendete DBMS (Datenbankmanagementsystem) problemlos und ohne zusätzliche Aufwände ausgetauscht werden kann. |
| Benutzbarkeit | Es ist einfach gestaltet und einfach zu bedienen. |
| Effizienz | Das Programm soll zu jeder Zeit, ungeachtet des Projektumfangs, schnell laufen. |
| Sicherheit | Die Projektdaten sollen abgeschirmt auf einem Server gespeichert sein, welcher gut geschützt ist. |
| Zuverlässigkeit | Der Mängel-Manager muss auf jede mögliche Fehlerquelle korrekt reagieren. |
| Betreibbarkeit | Der Mängel-Manager soll kostengünstig und ohne viel Aufwand zu betreiben sein. |

## Stakeholder

|  |  |  |
| --- | --- | --- |
| Beteilligte | Rolle | Beschreibung |
| Generalunternehmen | BackOffice | Das BackOffice des GU ist sozusagen die Zentralstelle wenn es um den Mängel-Manager geht dementsprechend verfügt Sie auch über alle Funktionalitäten des Mängel-Managers. Sie kann Stammdaten und Mangeldaten hinzufügen, bearbeiten oder auch löschen. Mit Stammdaten sind aktuelle Projekte, Login Daten (Generalunternehmen Bauleiter, Subunternehmen BackOffice) gemeint. |
| Bauleiter | Der Bauleiter ist für Projekte zuständig. Er führt Kontrollen durch und erfasst allfällige Mängel mit einem Notebook oder Tablet. Die Bauleiter können zu jeder Zeit und an jedem Ort die gefundenen Mängel per Mängel-Manager erfassen. |
| Subunternehmen | BackOffice | Das BackOffice des SU kann seine eigenen Stammdaten ändern und die Mängel die der betreffen SU zugewiesen sind einsehen und bearbeiten. |
| Ansprechperson | Wenn bei einer Kontrolle ein Mangel von einem Bauleiter erfasst wird, wird es dem betreffendem SU zugeteilt. Das SU bestimmt eine Ansprechperson welche für diesen Mangel zuständig ist. Die Ansprechperson kann also demnach nur diejenigen Mängel sehen und bearbeiten die ihm zugeteilt worden sind. |

# Randbedingungen

## Technische Randbedingungen

|  |  |
| --- | --- |
| **Hardware-Vorgaben** | |
| Angemessene Hardwareausstattung | Der Mängel-Manager muss auf einem, zum Release Zeitpunkt, marktüblichen Notebook und Desktop-PC betrieben werden können. |
| Speicher | Erwähnenswert? |
| Netzwerk | Erwähnenswert? |
| Firewall | Erwähnenswert? |
| Software-Vorgaben | |
| *Siehe Punkt 1.1 „Aufgabenstellung“ für Software-Vorgaben* | |
| **Vorgaben des Systembetriebs** | |
| Kompatibel mit Windows 8 & 8.1 | Die Software muss voll funktionsfähig mit Windows 8 und 8.1 sein. |
| Datenbanksysteme | Der Mängel-Manager benutzt das Datenbankmanagementsystem (DBMS) PostgreSQL, sollte aber ohne grossen Aufwand auf andere DBMS konfiguriert werden. |
| Systembetrieb | Batch oder Onlinebetrieb? |
| Verfügbarkeit der Server | Die Verfügbarkeit sollte mindesten bei der heutigen allgemein normalen Rate von 99% liegen. |
| **Programmiervorgaben** | |
| Programmierung in Java | Die Lösung sollte in Java implementiert werden. Es wird mit der aktuellsten Java Version entwickelt. (Java SE 8) |
| Bibliotheken, Frameworks und Komponenten | Sollen bestimmte „Software-Fertigteile“ eingesetzt werden? |
| Middleware | ? |

## Organisatorische Randbedingungen

|  |  |
| --- | --- |
| **Organisation und Struktur** | |
| Organisationsstruktur beim Arbeitgeber | Die Projektaufgabe wurde anhand eines PDF-Dokuments, welche alle nötigen Informationen zur Implementierung und Dokumentationen beinhaltet, gestellt.  Die Ansprechpersonen sind klar definiert und werden sich während der Projektdauer voraussichtlich nicht ändern. |
| Eigenentwicklung | Das Projekt wird zu 100% vom unten genannten Team entwickelt unter der Zuhilfenahme von einigen Java Libraries. |
| Entwicklung als Produkt | Angeben ob Schulprojekt oder nicht? |
| **Ressourcen (Budget, Zeit, Personal)** | |
| Zeitplan | *Siehe Punkt 23. „Weitere Dokumentationen“* |
| Release | Der Abgabetermin des Mängel-Managers ist auf Mittwoch den 20.5.2015, 12.00 Uhr datiert. |
| Team | Zum Team gehören 6 Studenten der Hochschule Luzern – Departement Wirtschaft, Studiengang Wirtschaftsinformatik, 2. Semester Vollzeit:  Luca Kündig (Projektleiter) Sandro Ritz (Projektleiter Stv.) Cihan Demir Mike Iten Mike Monticoli |
| Präsentation | Erwähnenswert? |
| **Organisatorische Standards** | |
| Vorgehensmodell | Die Modellierung der Software wird mit Hilfe von „use case diagram“ in der Modelierungssprache „Unified Modeling Language“ (UML) durchgeführt. Zusätzlich wird ein „Entity-Relationship-Modell“ (ERM) der zukünftigen Datenbank erstellt. |
| Entwicklungswerkzeuge | Vorgaben bezüglich der Entwicklungswerkzeuge  (etwa: CASE-Tool, Datenbank, Integrierte  Entwicklungsumgebung, Kommunikationssoftware,  Middleware, Transaktionsmonitor).  Enterprise Architect  Eclipse  Dokumentation – Word Online (OneDrive) |
| Abnahme- und Freigabeprozesse | Mittwoch meeting? |
| Mockup |  |
| Konfigurations- und  Versionsverwaltung | Das ganze Projekt wird als „public repositorie“ von „GitHub“ gehostet womit direkt auch „Git“ für die Versionsverwaltung zum Einsatz kommt. |
| Testwerkzeuge und  Prozesse | Vorgaben? Junit? |
| **Juristische Faktoren** | |
| Datenschutz | Das System kann nach Belieben konfiguriert werden, im Normalfall haben die Unternehmen nur auf Daten Zugriff die für Ihren Geschäftsfall auch relevant ist. |

## Konventionen

|  |  |
| --- | --- |
| **Konventionen** | |
| Architekturdokumentation | Dokumentierung des ganzen Softwareprozesses unter Benützung des arc42-Template-v60. |
| Implementierungsrichtlinien | Sich an die Grundlegenden Java Coding Conventions von Oracle halten. |

# Kontextabgrenzung

|  |  |  |
| --- | --- | --- |
| Schnittstelle | Daten | Technologie |
| Java Application | Quellcode | Via Webservice |
| PostgreSQL | Objekte | RMI |

## Fachlicher Kontext



# Lösungsstrategie

Der Mängel-Manager wurde aufgrund der Punkte Plattformunabhängigkeit, Sicherheit und Flexibilität in der Programmiersprache Java geschrieben und weist verschiedene Schichten wie Business, Model, Persister, Webservice, RMI auf. Für weitere Informationen siehe Punkt 7. „Verteilungssicht“.

Für die Grafische Benutzeroberfläche (GUI) wurde das aktuelle JavaFX 8 UI-Toolkit ausgewählt. Das Design wurde mithilfe der Layout-Visualisierungs-Software „Scene Builder welche von Oracle als Open Source Software daherkommt, verwirklicht. Auf Swing wurde bewusst verzichtet, da das neue JavaFX 8 moderner daherkommt, simpel zu bedienen ist und sich rasant weiterentwickelt.

Die Softwarelösung wird eine relationale Datenbank voraussetzen, in unserem Fall wird es das PostgreSQL sein. Das DBMS sollte leicht austauschbar sein weswegen wir auf JDBC (Java Database Connectivity) setzen um einen Austausch des Datenbankmanagementsystems schnell und einfach vornehmen zu können.

Es wird versucht vier verschiedene Sichten, also je eine Sicht für einen Stakeholder, zu realisieren. *Siehe Punkt 5. „Bausteinsicht“ für weitere detaillierte Informationen zu den verschiedenen Views.*

# Bausteinsicht

## Ebene 1

Die folgende Abbildung zeigt die Hauptbausteine unseres Systems und deren Abhängigkeiten.

<hier Überblicksdiagramm einfügen>

Erläuterung zu Struktur und Abhängigkeiten der Ebene 1:

### Bausteinname 1 (BlackBox-Beschreibung)

<Black Box Template des Bausteins hier einfügen>

### Bausteinname 2 (Black Box-Beschreibung)

<Black Box Template des Bausteins hier einfügen>

### ...

<Black Box Template des Bausteins hier einfügen>

### Bausteinname n (Black Box-Beschreibung)

<Black Box Template des Bausteins hier einfügen>

### Offene Punkte

## Ebene 2

### Bausteinname 1 (Whitebox-Beschreibung)

< Hier Überblicksdiagramm für Innenleben von Baustein 1 einfügen>

#### Bausteinname 1.1 (BlackBox Beschreibung)

Struktur gemäß Black-Box- Template:

1. Zweck / Verantwortlichkeit:
2. Schnittstelle(n):
3. Erfüllte Anforderungen:
4. Variabilität:
5. Leistungsmerkmale:
6. Ablageort / Datei:
7. Sonstige Verwaltungsinformation:
8. Offene Punkte:

#### Bausteinname 1.2 (Black Box Beschreibung)

Struktur gemäß Black-Box- Template:

1. Zweck / Verantwortlichkeit:
2. Schnittstelle(n):
3. Erfüllte Anforderungen:
4. Variabilität:
5. Leistungsmerkmale:
6. Ablageort / Datei:
7. Sonstige Verwaltungsinformation:
8. Offene Punkte:

#### ...

#### Bausteinname 1.n (Black Box Beschreibung)

Struktur gemäß Black-Box- Template:

1. Zweck / Verantwortlichkeit:
2. Schnittstelle(n):
3. Erfüllte Anforderungen:
4. Variabilität:
5. Leistungsmerkmale:
6. Ablageort / Datei:
7. Sonstige Verwaltungsinformation:
8. Offene Punkte:

#### Beschreibung der Beziehungen

#### Offene Punke

### Bausteinname 2 (Whitebox-Beschreibung)

#### Überblicksdiagramm 2 Ebene 2

1. zeigt das Innenleben des Bausteines in Diagrammform mit den lokalen Bausteinen 1 - n, sowie deren Zusammenhänge und Abhängigkeiten.
2. beschreibt wichtige Begründungen, die zu dieser Struktur führen
3. verweist evtl. auf verworfene Alternativen (mit der Begründung, warum es verworfen wurde

#### Bausteinname 2.1 (BlackBox Beschreibung)

Struktur gemäß Black-Box- Template:

1. Zweck / Verantwortlichkeit:
2. Schnittstelle(n):
3. Erfüllte Anforderungen:
4. Variabilität:
5. Leistungsmerkmale:
6. Ablageort / Datei:
7. Sonstige Verwaltungsinformation:
8. Offene Punkte:

#### Bausteinname 2.2 (Black Box Beschreibung)

Struktur gemäß Black-Box- Template:

1. Zweck / Verantwortlichkeit:
2. Schnittstelle(n):
3. Erfüllte Anforderungen:
4. Variabilität:
5. Leistungsmerkmale:
6. Ablageort / Datei:
7. Sonstige Verwaltungsinformation:
8. Offene Punkte:

#### ...

#### Bausteinname 2.n (Black Box Beschreibung)

Struktur gemäß Black-Box- Template:

1. Zweck / Verantwortlichkeit:
2. Schnittstelle(n):
3. Erfüllte Anforderungen:
4. Variabilität:
5. Leistungsmerkmale:
6. Ablageort / Datei:
7. Sonstige Verwaltungsinformation:
8. Offene Punkte:

#### Beschreibung der Beziehungen

#### Offene Punkte

### Bausteinename 3 (Whitebox-Beschreibung)

#### Überblicksdiagramm 3 Ebene 2

1. zeigt das Innenleben des Bausteines in Diagrammform mit den lokalen Bausteinen 1 - n, sowie deren Zusammenhänge und Abhängigkeiten.
2. beschreibt wichtige Begründungen, die zu dieser Struktur führen
3. verweist evtl. auf verworfene Alternativen (mit der Begründung, warum es verworfen wurde

#### Bausteinname 3.1 (BlackBox Beschreibung)

Struktur gemäß Black-Box- Template:

1. Zweck / Verantwortlichkeit:
2. Schnittstelle(n):
3. Erfüllte Anforderungen:
4. Variabilität:
5. Leistungsmerkmale:
6. Ablageort / Datei:
7. Sonstige Verwaltungsinformation:
8. Offene Punkte:

#### Bausteinname 3.2 (Black Box Beschreibung)

Struktur gemäß Black-Box- Template:

1. Zweck / Verantwortlichkeit:
2. Schnittstelle(n):
3. Erfüllte Anforderungen:
4. Variabilität:
5. Leistungsmerkmale:
6. Ablageort / Datei:
7. Sonstige Verwaltungsinformation:
8. Offene Punkte:

#### ...

#### Bausteinname 3.n (Black Box Beschreibung)

Struktur gemäß Black-Box- Template:

1. Zweck / Verantwortlichkeit:
2. Schnittstelle(n):
3. Erfüllte Anforderungen:
4. Variabilität:
5. Leistungsmerkmale:
6. Ablageort / Datei:
7. Sonstige Verwaltungsinformation:
8. Offene Punkte:

#### Beschreibung der Beziehungen

#### Offene Punkte

## Ebene 3

# Laufzeitsicht

## Laufzeitszenario 1

## Laufzeitszenario 2

## ...

## Laufzeitszenario n

# Verteilungssicht

## Infrastruktur Ebene 1

### Verteilungsdiagramm Ebene 1

### Prozessor 1

### Prozessor 2

### ...

### Prozessor n

### Kanal 1

### Kanal 2

### ...

### Kanal m

### Offene Punkte

## Infrastruktur Ebene 2

# Konzepte

## Fachliche Strukturen und Modelle

## Typische Muster Strukturen

## Typische Abläufe

Die Abbildung X zeigt das Erfassen eines Mangels visualisiert auf:



## Persistenz

Um die Persistenz sicherzustellen müssen verschiedene Bedingungen wie erfolgreicher Login, stabile Konnektivität und reibungslose Kommunikation zwischen Client bis zu der letzten Schicht der Architektur (Datenbank) des Mängel-Manager erfüllt sein.

Der Client muss z.B. bis zur Erstellung eines Mangels mit der Datenbank verbunden sein um diese Informationen auch erfolgreich persistieren zu können. Alles was nicht mit der Datenbank synchronisiert worden ist aufgrund Unvollständigkeit befindet sich im Cache Speicher des Clients und ist somit flüchtig, was bei einem Systemabsturz verloren wäre.

Die Unmengen an Daten welche die Datenbank beherbergt, werden bei der Persistenz schlussendlich, für alle erfolgreiche Verbindung, durch das Massenspeicher der Server zur Verfügung gestellt.

## Benutzungsoberfläche

## Ergonomie

## Ablaufsteuerung

## Transaktionsbehandlung

## Sessionbehandlung

## Sicherheit

Um sich überhaupt mit der Datenbank verbinden zu können wird neben der Client Software ein Login vorausgesetzt, welcher zuerst vom BackOffice erstellt werden muss. Wenn eine erfolgreiche Authentifikation stattgefunden hat wird als aller erstes geprüft welche Rechte der Benutzer hat und dementsprechend werden nur Items angezeigt die für diesen Benutzer relevant sind.

Momentan gibt es zwei Benutzergruppen, das BackOffice GU und das BackOffice SU mit folgenden Rechten:

* **Die BackOffice GU:**

Kann neue BackOffice GU wie auch BackOffice SU Benutzer erstellen. Kann auf alle Informationen zugreifen, Stammdaten wie auch Mängeldaten hinzufügen bearbeiten oder löschen. Im Grunde genommen ist der BackOffice GU Benutzer eine Art Administrator.

* **Die BackOffice SU:**

Kann nur neue BackOffice SU erstellen (dient anstelle der Ansprechspartner SU). Kann nur auf Informationen zugreifen welche für die BackOffice SU bestimmt sind. Kann nur die eigenen Stammdaten bearbeiten. Kann nur die Mängel bearbeiten die Ihr zugeteilt sind.

Es war auch eine Benutzergruppe für den Bauleiter und die Ansprechperson geplant, jedoch mussten wird den Funktionsumfang, aufgrund einer Verringerung der Projektmitglieder, kürzen um den Zeitplan auch einhalten zu können.

## Kommunikation und Integration mit anderen IT-Systemen

## Verteilung

## Plausibilisierung und Validierung

## Ausnahme-/Fehlerbehandlung

## Management des Systems & Administrierbarkeit

Der Mängel-Manager wird auf einem IT-System zentral in der Generalunternehmung der „W&W Architekten GmbH“ betrieben. Die Stakeholder werden bezüglich spezieller Eingriffs- oder Konfigurationsmöglichkeiten bei Bedarf, eingeschult.

## Logging, Protokollierung, Tracing

Die Anwendungsmeldungen werden hauptsächlich mithilfe des „log4j“ Frameworks, welcher von der Apache Software Fundation bereitgestellt wird, geloggt.

## Geschäftsregeln

## Konfigurierbarkeit

## Parallelisierung und Threading

## Internationalisierung

Die „W&W Architekten GmbH“ ist ein in der Schweiz tätiges Unternehmen. Es wird grundsätzlich mit schweizer Standard Einstellungen gearbeitet, weswegen wir auf Anpassungen betreffend länderspezifischer Merkmale verzichten können.

## Migration

Da die „W&W Architekten GmbH“ kein Altsystem benutzt hat gibt es wenig zu migrieren. Es müssen Stammdaten und eventuell offene, schon bekannte Mängel erfasst werden.

Die die bereits bekannten, offenen Mängel schon im Excel Format bereit stehen, können sie einfach mithilfe des GUI per Copy & Paste übertragen werden. Es macht wenig Sinn für eine Aufgabe die nur einmal auftritt und die eine überschaubare Grösse hat Scripts zu schreiben welche einen grösseren Programmieraufwand hat im Vergleich zur manuellen Erledigung der Aufgabe.

## Testbarkeit

*Siehe Kapitel 18. „Funktionale Test‘s“*

## Skalierung, Clustering

## Hochverfügbarkeit

Eine Verfügbarkeit von 99% ist heutzutage ein Standard bei qualitativ hochwertigen EDV-Geräten und wird somit vorausgesetzt. Eine Hochverfügbarkeit von 99.99% würde der Verfügbarkeitsklasse 4 entsprechen welche von uns auch angestrebt wird und mithilfe des USV-Systems auf ziemlich realistisch scheint. Die Verfügbarkeitsklasse 4 erlaubt es pro Monat 4:23 Minuten und pro Jahr 52:36 Minuten offline zu sein.

# Entwurfsentscheidungen

## Entwurfsentscheidung 1

### Fragestellung

### Rahmenbedingungen

### Annahmen

### Betrachtete Alternativen

### Entscheidung

## Entwurfsentscheidung n

# Qualitätsszenarien

## Qualitätsbaum



## Bewertungsszenarien

Nutzungsszenario (Qualitätsziel Zuverlässigkeit):

Dem Bauleiter wird vor Ort mitgeteilt, dass einige Ressourcen bezüglich des Baus fehlen. Entweder muss er den Projektplan ändern oder er muss sich beim Subunternehmen melden.

Dazu erstellt er über sein Tablet erstmal einen Mangel um später noch einmal darauf hingewiesen zu werden.

Nutzungsszenario (Qualitätsziel Benutzbarkeit):

Ein Bauleiter muss sich neu mit dem System befassen und startet zum ersten mal die Menüansicht. Er sieht alles auf einen Blick und wenn er die Menüwahl links ändert, bleibt alles einheitlich bis auf den mittleren Bereich welcher sich dynamisch verändert. Somit bleibt alles übersichtlich und einfach zu bedienen.

# Risiken

Das Risikomanagement umfasst sämtliche Maßnahmen zur systematischen Erkennung, Analyse, Bewertung, Überwachung und Kontrolle von Risiken. Hier werden die uns wichtigs-ten Risiken aufgezählt.

**Risiko 1:**

Datenverlust von Projektdaten

Eintrittswarscheinlichkeit:

Die Eintrittswahrscheinlichkeit ist klein.

Schadenshöhe:

Je nach Verlust grösser oder kleiner

Massnahmen:

Jeden Tag wird ein Backup der Kundendaten durchgeführt. Datenbank ist hinter einer USV, so ist auch bei Stromausfall die Datenbank weiterhin online.

**Risiko 2:**

Angriff von aussen auf den Server

Eintrittswahrscheinlichkeit:

Die Eintrittswahrscheinlichkeit ist klein.

Schadenshöhe:

Grosser Verlust

Massnahmen:

Der Server soll gut geschützt sein und zu jederzeit mit dem aktuellsten Firewall ausgestattet sein.

# Glossar

|  |  |
| --- | --- |
| Begriff | Beschreibung |
| Graphical user interface | Auf Deutsch auch grafische Benutzeroberfläche genannt. Dient dazu eine Applikation mittels Symbole oder Steuerelemente bedienbar zu machen um Aktionen einfach mit der Maus und der Tastatur ausführen zu können. |
| Datenbankmanagementsystem (DBMS) | Ein Datenbankmanagementsystem ist eine Software, welche an erster Stelle, eine Datenbank zur verfügen stellt, die verschiedene Nutzern-Anfragen oder Anfragen von anderen Programmen entgegennehmen und bearbeiten kann. |
| Entity-Relationship-Modell (ERM) | Auf Deutsch etwa Gegenstand-Beziehung-Modell, welches primär für den Entwurf der Datenbank dient, bevor sie per SQL (Structured Query Language) implementiert wird. |
| Mockup | Im Deutschen auch unter Maquette bekannt. Kann als Instrument bezeichnet werden um Entwürfe oder Vorformen einer Website oder eines App’s zur Konzeption zu erstellen. |
| Middleware | Eine Middleware ist eine sogenannte Zwischenanwendung die zwischen zwei verschiedenen Schichten eines Programms vermittelt was somit auch die Komplexität und Infrastruktur gegenüber der Aussenwelt verborgen hält. |
| Coding conventions | Im Deutschen auch unter dem Begriff Programmierstil bekannt, welcher in der Programmierung vorgibt nach welchen Regeln der Quellcode erstellt werden muss. |
| Use cases, use case diagram | Das use case diagram ist ein Instrument welches für die Erstellung von verschiedenen Szenarien dient, die einem Stakeholder (Akteur) möglich sind, zur Erreichung eines bestimmten technischen Zieles. |
| Unified Modeling Language (UML) | Die Unified Modeling Language ist eine grafische Modellierungssprache die grundsätzlich zur Visualisierung von Diagrammen benutzt wird. |
| Framework | Unter einem Framework versteht man im Bereich der Software-Engineering ein Rahmenwerk, welches Komponenten und Libraries zur Verfügung stellt womit sie dem Programmierer beim Erstellen einer Applikation unter die Arme greift. |
| Java Database Connectivity (JDBC) | Die Java Database Connectivity ist eine universelle Java Datenbankschnittstelle, die eine Schnittstelle zu Datenbanken verschiedener Hersteller bietet. |

*Die Beschreibungen der Begriffe oben sind kurzgefasste, vereinfachte Formen der Beschreibungen die jeweils unter http://www.itwissen.info oder http://de.wikipedia.org zu finden sind.*

# Requirements Dokumentation (2015\_FS\_INM21\_Requirements\_MangelManager.xls)

|  |  |
| --- | --- |
| Nr. | Anforderung |
| 1 | Datenverwaltung Back-Office für Subunternehmen |
| 2 | Datenverwaltung des Unternehmens |
| 3 | Datenverwaltung für alle Bauprojekte des Unternehmens |
| 4 | Projektobjekt muss ersichtlich sein (Mehrfamilienhaus, Wohnung usw.) |
| 5 | Projekttyp klar ersichtlich (Neubau, Renovation…) |
| 6 | Projekte haben Start und Enddatum |
| 7 | Adresse des Projekts ist ersichtlich |
| 8 | Bauherr des Projekts ist ersichtlich |
| 9 | Arbeiten/Mängel aller Bauleiter eines Projekts sind zeitlich aufgeführt |
| 10 | Alle involvierten Subunternehmen sind ersichtlich |
| 11 | Die Ansprechpersonen der Subunternehmen sind zeitlich aufgeführt |
| 12 | Mängel können vor Ort durch den Bauleiter erfasst werden |
| 13 | Mängeldaten werden direkt auf dem Server der GU gespeichert |
| 14 | Benutzerschnittstelle für Administration |
| 15 | Benutzerschnittstelle für Bauleiter |
| 16 | Benutzerschnittstelle für Subunternehmen |
| 17 | Sicheres Login auf allen Benutzerschnittstellen |
| 18 | Datenzugriff für die jeweiligen Benutzer einschränken |
| 19 | Subunternehmenssoftware wird Online zum Download zur Verfügung gestellt |
| 20 | Zuverlässiges Datenbankmanagementsystem wird zum Verwalten der Daten verwendet |
| 21 | Software darf nicht Datenbank abhängig sein |
| 22 | Projekte können durch mehrere Bauleiter betreut werden |
| 23 | Hauptbauleiter des Projekts ist ersichtlich und kann ausgetauscht werden |
| 24 | Ein Subunternehmen kann in mehreren Projekten tätig sein |
| 25 | Projekte können durch mehrere, müssen aber durch mindestens eine Ansprechperson des SU betreut werden. |
| 26 | Bauleiter dürfen nur eigenen Projekte einsehen und bearbeiten |
| 27 | Bauleiter eines Projekts werden durch die Administration des GU festgelegt |
| 28 | Ansprechpersonen eines SU dürfen nur eigene Projekte einsehen und bearbeiten |
| 29 | Stammdaten müssen durch Administrative Mitarbeiter Initial erfasst werden |
| 30 | Sobald Mängel erfasst wurden sind diese auch für das SU ersichtlich |
| 31 | Mängel können durch den SU Mitarbeiter quittiert werden |
| 32 | Mängel können mehrere Meldungen beinhalten, die durch GU oder SU erfasst werden können |
| 33 | Meldungen von Mängeln stehen dem Bauleiter sofort zur Verfügung |
| 34 | Mängelbehebung muss durch SU explizit angegeben werden. |
| 35 | GU Bauleiter kontrolliert Mängel und bestätigt falls Behebung in Ordnung |
| 36 | Sollten die Mängel nicht behoben sein kann der Bauleiter die Mängel erneut dem SU schicken |
| 37 | Meldungen sind nach dem Speichern unveränderbar |
| 38 | Alle UIs müssen die Mängeldaten nach diversen Kriterien filtern können |
| 39 | Mängeldaten müssen in Listenform angezeigt und ausgedruckt werden können. |
| 40 | Benutzerverwaltung durch GU-BO |
| 41 | Adminuser für GU-BO wird initial erstellt |
| 42 | Adminuser für SU-BO wird bei Erfassung von SU im GU-BO erstellt |

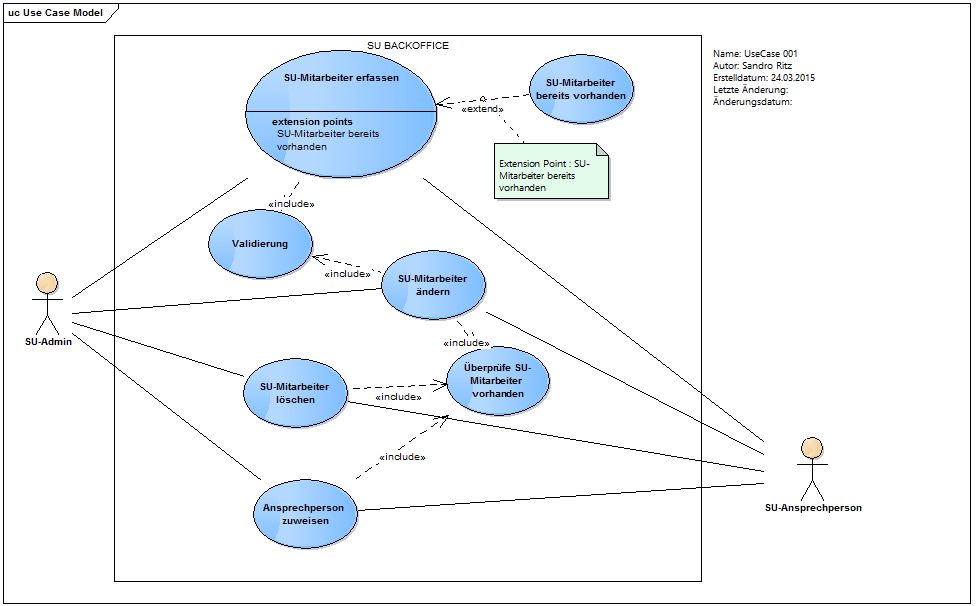
# UseCase Dokumentation

## UseCase001

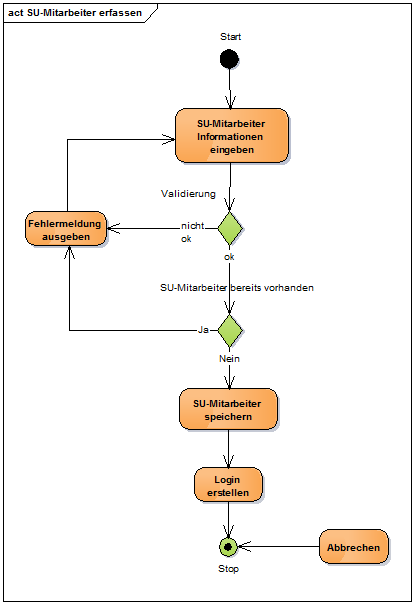
### UseCase001 Beschreibung

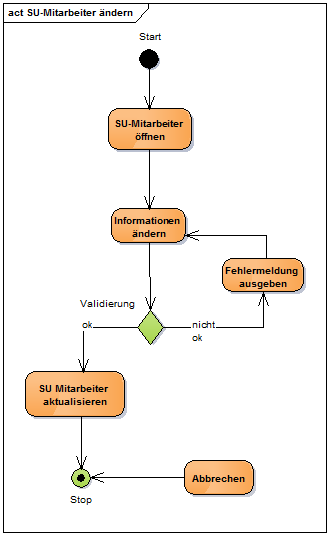
|  |  |
| --- | --- |
| UseCase001 | SU-Backoffice Datenverwaltung |
| Autor | Sandro Ritz |
| Ziel | Der BackOffice-Mitarbeiter der Subunternehmen kann Ansprechpersonen in der Datenbank sowohl erfassen, löschen, ändern, lesen und einem Projekt zuweisen. |
| Kategorie | Primär, ist essenziell für die Funktionalität des Mängelmanagers |
| Vorbereitungen | Ein SU-Ansprechperson ist noch nicht im System erfasst |
| Nachbedingungen Erfolg | Die SU-Ansprechperson ist mit ID, Namen, Vornamen, Rolle, Direkt-Telefon und Login Daten im Mängelmanager erfasst |
| Nachbedingungen Fehlschlag | Die SU-Ansprechperson ist nicht oder fehlerhaft (Mit falschen oder unvollständigen Informationen) im Mängelmanager erfasst |
| Akteure | SU-Admin, SU-Ansprechperson |
| Szenario 01 | |
| Auslösendes Ereignis | Neuer SU-Mitarbeiter tritt in ein Unternehmen ein. |
| Beschreibung | 1. BackOffice Mitarbeiter überprüft ob der Mitarbeiter bereits im System ist 2. BackOffice Mitarbeiter erfasst den Mitarbeiter mit ID, Namen, Vornamen, Rolle, Direkt-Telefon und Login Daten im Mangelmanager 3. Mitarbeiter klickt auf Save und speichert so den neuen Mitarbeiter und Benutzer 4. Die Benutzerdaten werden in der Datenbank gespeichert |
| Szenario 02 | |
| Auslösendes Ereignis | SU-Mitarbeiter verlässt das Unternehmen |
| Beschreibung | 1. BackOffice Mitarbeiter überprüft ob der SU-Mitarbeiter im System vorhanden ist. 2. BackOffice Mitarbeiter deaktiviert SU-Mitarbeiter inkl. Login. Der SU-Mitarbeiter wird nicht aus der Datenbank gelöscht |
| Szenario 03 | |
| Auslösendes Ereignis | SU-Mitarbeiter Information ändern. z.B. Telefonnummer wird geändert. |
| Beschreibung | 1. BackOffice Mitarbeiter überprüft ob der SU-Mitarbeiter im System vorhanden ist. 2. SU-Mitarbeiter Datensatz wird geöffnet. 3. Daten werden geändert. 4. Datensatz wird mittels Klick auf „Speichern“ abgespeichert und in der Datenbank aktualisiert. |
| Szenario 04 | |
| Auslösendes Ereignis | Einem Projekt wird eine Ansprechperson zugewiesen. |
| Beschreibung | 1. BackOffice Mitarbeiter öffnet in der Projektverwaltung ein ausgewähltes Projekt. 2. Falls keine Ansprechperson zugeteilt ist, kann eine Ansprechperson ausgewählt und gesetzt werden. 3. Falls bereits eine Ansprechperson zugeteilt ist, kann diese geändert werden. Mit einem Klick auf Ansprechperson ändern öffnet sich eine Eingabemaske. 4. Neue Ansprechperson wird ausgewählt. 5. Mit „OK“ bestätigt man die Änderung. 6. Daten werden in der Datenbank aktualisiert. |

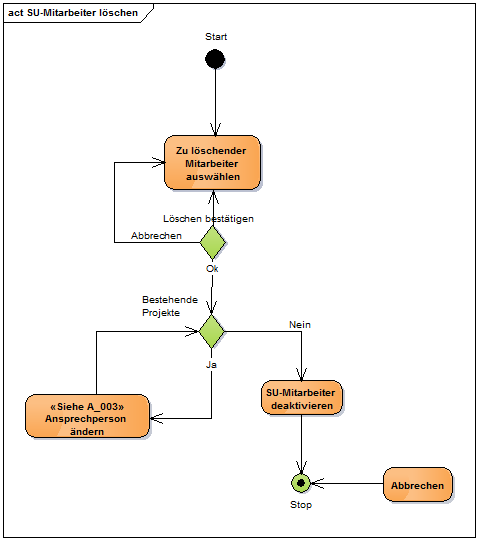
### UseCase001 Visualisierung

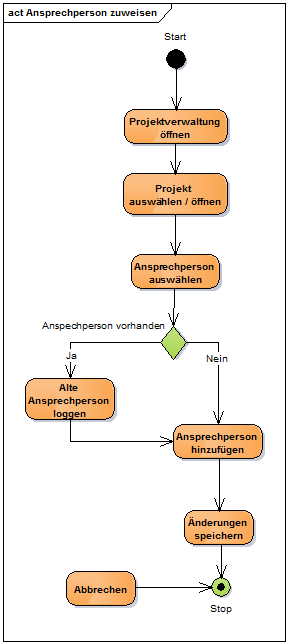


### UseCase001 Aktivitätsdiagramm







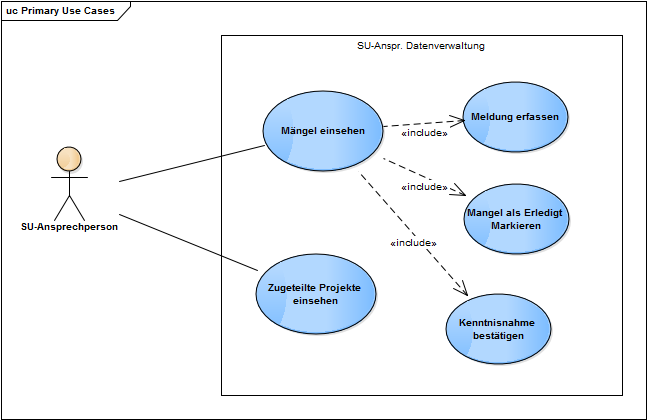


## UseCase002

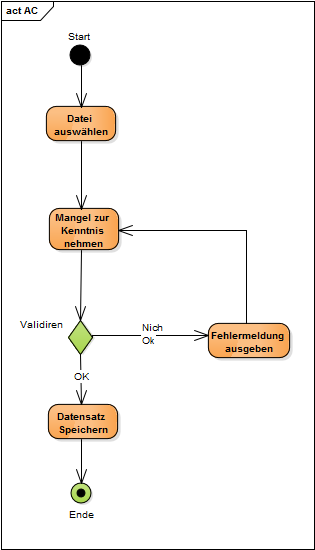
### UseCase002 Beschreibung

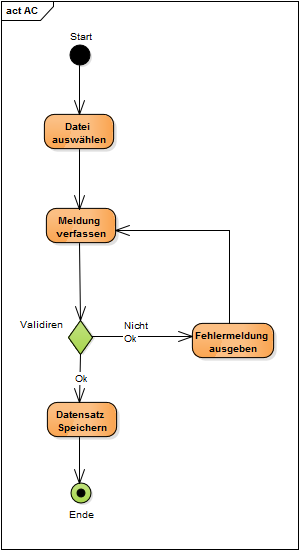
|  |  |  |
| --- | --- | --- |
| UseCase002 | SU-Ansprechperson Datenverwaltung | |
| Autor | Max von Gelhorn | |
| Ziel | Die Ansprechperson des Subunternehmens kann für die Ihm Zugeteilte Projekte die Mängel welche seine Firma betreffen einsehen, bestätigen und Kommentieren. | |
| Kategorie | Primär, Ist essenziell für die Funktionalität des Mängelmanagers | |
| Vorbedingungen | Es sind vom Projektleiter Mängel erfasst worden welche den SU-Anspr. Betreffen. Und welche der SU-Anspr. Noch nicht zur Kenntnis genommen hat. | |
| Nachbedingungen Erfolg | Der SU-Anspr. Hat die Ihm Betreffenden Mängel abgearbeitet.  Und in der Datenbank sind die Dazugehörigen Daten dem Entsprechen abgeändert worden. | |
| Nachbedingungen Fehlschlag | Der SU-Anspr. Konnte keine Verbindung zum Server herstellen und konnte deshalb die abgearbeiteten Daten nicht Speichern. | |
| Akteure | GU-Bauleiter, SU-Ansprechperson | |
| Szenario 01 SU-Ansprechpeson kann Mangel-Kenntnisnahme bestätigen SU-UI | |
| Auslösendes Ereignis | Der GU-Bauleiter hat einen neuen Mangel erfasst. |
| Beschreibung | 1. Der SU-Anspr. Loggt sich im Tool ein. 2. Der SU-Anspr. sieht dass es neue Mängel gibt. 3. Wenn er die Meldung gelesen hat, kann er bestätigen dass er sie gelesen hat. |
| Alternativen | Der SU-Anspr. Liehst den Mängel aber bestätigt nicht dass er Ihn gelesen hat. |
| Szenario 02 SU-Ansprechperson kann zugeteilte Mängel/Projekte in SU-UI einsehen | |
| Auslösendes Ereignis | Der SU-Anspr. Will sehen was für Mängel/Projekte er beheben muss. |
| Beschreibung | 1. Der SU-Anspr. Loggt sich im Tool ein. 2. Der SU-Anspr. sieht eine Liste mit allen Mängel/Projekte welche ihn betreffen. |
| Alternativen | Der SU-Anspr. Kann die Mängel/Projekte nicht einsehen da er keine Verbindung zum Server hat. |
| Szenario 03 SU-Ansprechperson kann Meldung zu Mangel verfassen in SU-UI | |
| Auslösendes Ereignis | Der SU-Anspr. Will eine Meldung zum Mängel schreiben, die der GU-Bauleiter einsehen kann. |
| Beschreibung | 1. Der SU-Anspr. Loggt sich im Tool ein. 2. Der SU-Anspr. Verfasst zu einem Mangel eine Meldung und Speichert diese. |
| Alternativen | Der SU-Anspr. Kann die Mängel nicht einsehen da er keine Verbindung zum Server hat. |
| Szenario 04 SU-Ansprechperson kann Mangel als erledigt markieren in SU-UI | |
| Auslösendes Ereignis | Der SU-Anspr. Hat einen Mangel behoben und möchte diesen als erledigt markieren. |
| Beschreibung | 1. Der SU-Anspr. Loggt sich im Tool ein. 2. Der SU-Anspr. Setzt den Status von einem Mangel auf „Erledigt“ |
| Alternativen | Der SU-Anspr. Kann den Status nicht ändern da er keine Verbindung zum Server hat. |

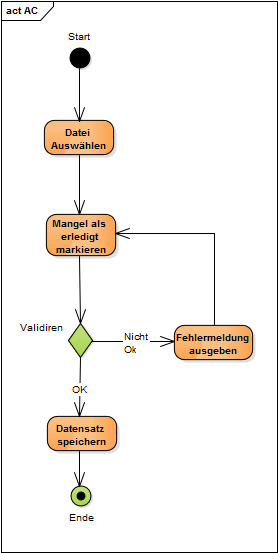
### UseCase002 Visualisierung



### UseCase002 Aktivitätsdiagramm





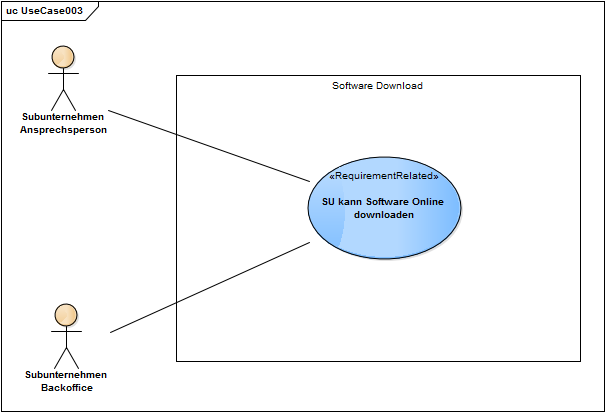


## UseCase003

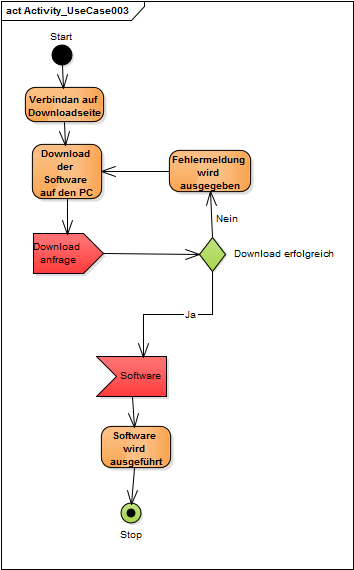
### UseCase003 Beschreibung

|  |  |
| --- | --- |
| UseCase003 | SU kann Software Online Downloaden |
| Autor | Luca Kündig |
| Ziel | Mitarbeiter eines Subunternehmens können die Mängelmanager Software Downloaden |
| Kategorie | Sekundär, da kein direkten Einfluss auf die Funktionalität |
| Vorbedingungen | Der Mitarbeiter des Subunternehmens hat die Software noch nicht auf seinem PC |
| Nachbedingungen Erfolg | Der Mitarbeiter des Subunternehmens kann die Mängelmanager Software auf seinem PC verwenden. |
| Nachbedingungen Fehlschlag | Der Mitarbeiter des Subunternehmens kann die Mängelmanager Software nicht downloaden, oder der Download ist fehlerhaft. |
| Akteure | Subunternehmen Ansprechperson, Subunternehmen Backoffice |
| Auslösendes Ereignis | Neues Subunternehmen wird erfasst und beteiligt sich an einem Projekt |
| Beschreibung | 1. Der Subunternehmen Mitarbeiter verbindet sich auf die Download Seite 2. Der Subunternehmen Mitarbeiter lädt sich die Datei auf seinen PC. 3. Download ist erfolgreich abgeschlossen 4. Der Subunternehmen Mitarbeiter führt die Software aus |
| Erweiterung |  |
| Alternativen | 1a Es gibt keine Download Seite: Die Datei wird via externes Medium (CD, Stick,…) Verbreitet. |

### UseCase003 Visualisierung



### UseCase003 Aktivitätsdiagramm

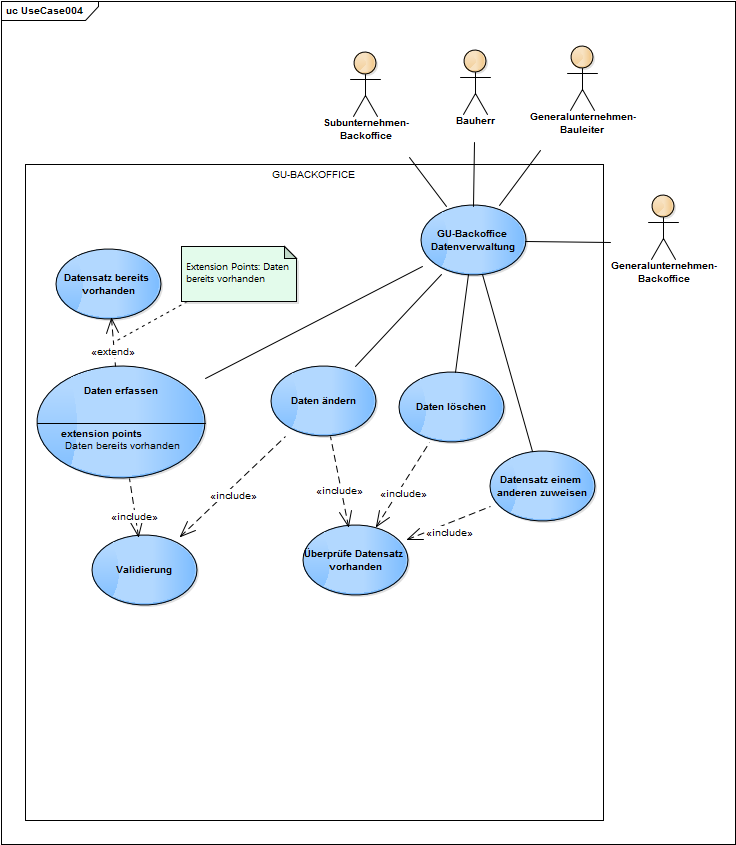


## UseCase004

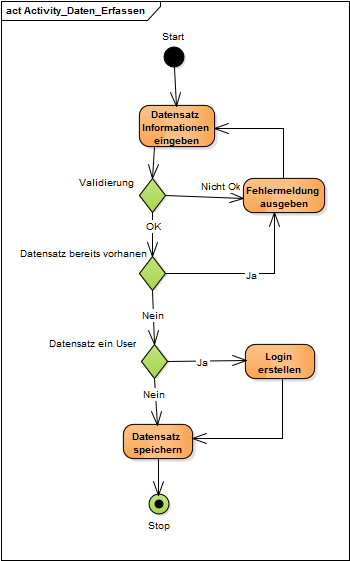
### UseCase004 Beschreibung

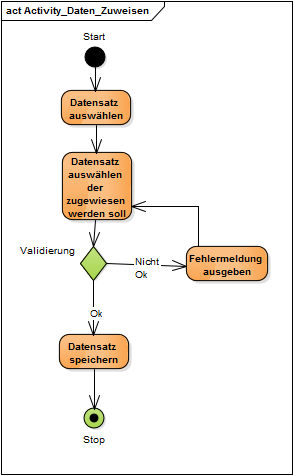
|  |  |
| --- | --- |
| UseCase004 | GU-BackOffice Datenverwaltung |
| Autor | Luca Kündig |
| Ziel | Der BackOffice-Mitarbeiter der Unternehmung kann Einträge in der Datenbank sowohl erfassen, löschen, ändern und lesen |
| Kategorie | Primär, Ist essenziell für die Funktionalität des Mängelmanagers |
| Vorbedingungen | Daten sind nicht vorhanden die erfasst werden müssen und die Daten die geändert, gelöscht oder gelesen werden müssen sind im System vorhanden |
| Nachbedingungen Erfolg | Daten sind im System erfasst, geändert oder gelöscht worden |
| Nachbedingungen Fehlschlag | Die Daten sind fehlerhaft erfasst, geändert, nicht gelöscht |
| Akteure | GU-BackOffice, GU-Bauleiter, SU-BackOffice, Bauherr |
| Szenario 01 | |
| Auslösendes Ereignis | Neuer Bauleiter tritt in die Unternehmung ein |
| Beschreibung | 1. BackOffice Mitarbeiter überprüft ob der Bauleiter bereits im System ist 2. BackOffice Mitarbeiter öffnet Maske zum Erfassen eines neuen Bauleiters 3. BackOffice Mitarbeiter erfasst den Bauleiter mit ID, Name, Vorname, Adresse, Rolle, Login-Daten im Mängelmanager 4. BackOffice Mitarbeiter klickt auf Save und speichert so den neuen Benutzer 5. Die Benutzerdaten werden in der Datenbank gespeichert |
| Erweiterung | 1a Benutzer existiert bereits: Nichts ist zu tun |
| Alternativen | 2a Der neue Bauleiter erfasst seinen User selber |
| Szenario 02 | |
| Auslösendes Ereignis | Bauherr ändert seine Adresse |
| Beschreibung | 1. BackOffice Mitarbeiter überprüft ob der Bauherr bereits im System ist 2. BackOffice Mitarbeiter öffnet den Datensatz des SU-BackOffice Mitarbeiter 3. BackOffice Mitarbeiter ersetzt die alte Adresse des Bauherren im Mängelmanager 4. BackOffice Mitarbeiter klickt auf Save und speichert so die neue Adresse 5. Die Adresse des Bauherren wird in der Datenbank gespeichert |
| Erweiterung | 1a Bauherr existiert noch nicht: Bauherr wird via Szenario 01 erfasst |
| Alternativen | - |
| Szenario 03 | |
| Auslösendes Ereignis | Bauherr beendet Zusammenarbeit mit GU, das Projekt wird gelöscht |
| Beschreibung | 1. BackOffice Mitarbeiter überprüft ob das Projekt im System vorhanden ist 2. BackOffice Mitarbeiter öffnet den Datensatz des Projekts 3. BackOffice Mitarbeiter klickt auf den delete Button im Mängelmanager 4. BackOffice Mitarbeiter klickt auf bestätigen Button im Mängelmanager 5. Die Projektdaten werden aus der Datenbank gelöscht |
| Alternativen | Status des Projekt wird auf abgebrochen geändert |

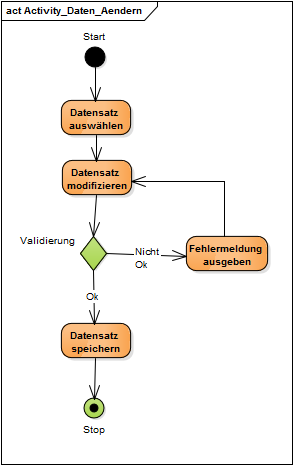
### UseCase004 Visualisierung

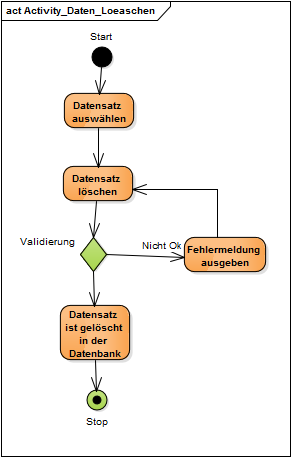


### UseCase004 Aktivitätsdiagramm







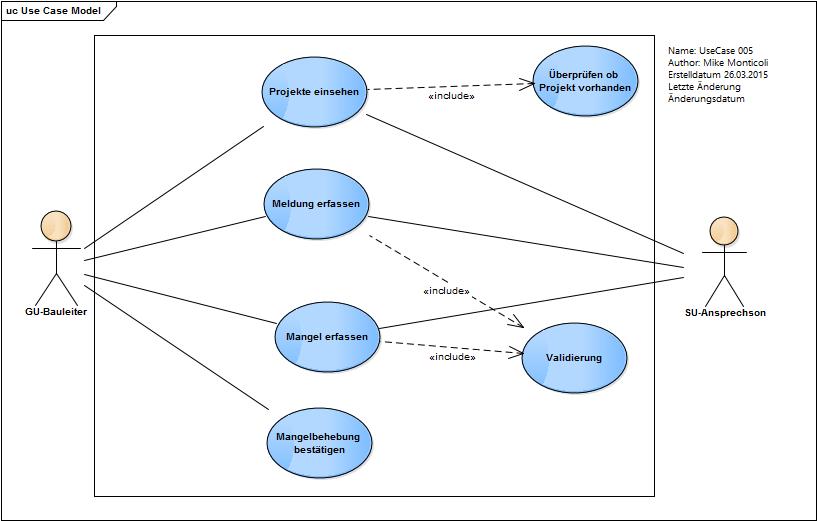


## UseCase005

### UseCase005 Beschreibung

|  |  |
| --- | --- |
| UseCase005 | GU-Bauleiter Datenverwaltung |
| **Autor** | Mike Monticoli |
| **Ziel** | GU-Bauleiter kann Mängel und Meldungen erfassen. Er kann alle Meldungen und Mängel zu seinem Projekt einsehen. Er kann die Mängelbehebung bestätigen. Er sieht all seine Projekte im GU-UI. |
| **Kategorie** | Primär, ist essenziell für die Funktionalität des Mängelmanagers |
| **Vorbedingungen** | Ein Mangel / Eine Meldung ist noch nicht erfasst oder muss bearbeitet werden |
| **Nachbedingungen Erfolg** | Der GU-Bauleiter konnte die Mängel und Meldungen erfassen beziehungsweise die gewünschte Mängelbehebung bestätigen. |
| **Nachbedingungen Fehlschlag** | Die Mängel und Meldungen wurden fehlerhaft erfasst oder die Mängelbehebung konnte nicht bestätigt werden. |
| **Akteure** | GU-Bauleiter, SU-Ansprechperson |
| **Szenario 1** | |
| **Auslösendes Ereignis** | GU-Bauleiter will neuen Mangel erfassen. |
| **Beschreibung** | GU-Bauleiter öffnet Projekt in GU-UI.  GU-Bauleiter erfasst neuen Mangel im GU-UI.  GU-Bauleiter füllt die Daten zum Mangel aus.  GU-Bauleiter bestätigt, dass die Daten korrekt erfasst wurden. |
| **Szenario 2** | |
| **Auslösendes Ereignis** | GU-Bauleiter will eine neue Meldung erfassen. |
| **Beschreibung** | GU-Bauleiter erfasst neue Meldung im GU-UI.  GU-Bauleiter füllt die Daten zur Meldung aus.  GU-Bauleiter bestätigt, dass die Daten korrekt erfasst wurden. |
| **Szenario 3** | |
| **Auslösendes Ereignis** | GU-Bauleiter will Meldung oder Mangel einsehen |
| **Beschreibung** | GU-Bauleiter öffnet das betroffene Projekt im GU-UI.  GU-Bauleiter öffnet die gewünschte Meldung beziehungsweise den gewünschten Mangel |
| **Szenario 4** | |
| **Auslösendes Ereignis** | GU-Bauleiter will die Behebung eines Mangels bestätigen. |
| **Beschreibung** | GU-Bauleiter öffnet das betroffene Projekt im GU-UI.  GU-Bauleiter öffnet den behobenen Mangel  GU-Bauleiter bestätigt die Mangelbehebung |
| **Szenario 5** | |
| **Auslösendes Ereignis** | GU-Bauleiter will seine Projekt im GU-UI einsehen |
| **Beschreibung** | GU-Bauleiter startet das GU-UI.  GU-Bauleiter öffnet das gewünschte Projekt.  GU-Bauleiter sieht die gewünschten Daten ein. |
| **Erweiterung** | Fehlermeldung bei:  Mangel ist bereits erfasst.  Mangel ist bereits behoben aber noch nicht entfernt.  Meldung ist bereits erfasst.  Meldung ist bereits gelöscht.  Projekt ist nicht vorhanden. |
| **Alternativen** | SU erfasst Mängel und kümmert sich um die Behebung dieser. |

### UseCase005 Visualisierung



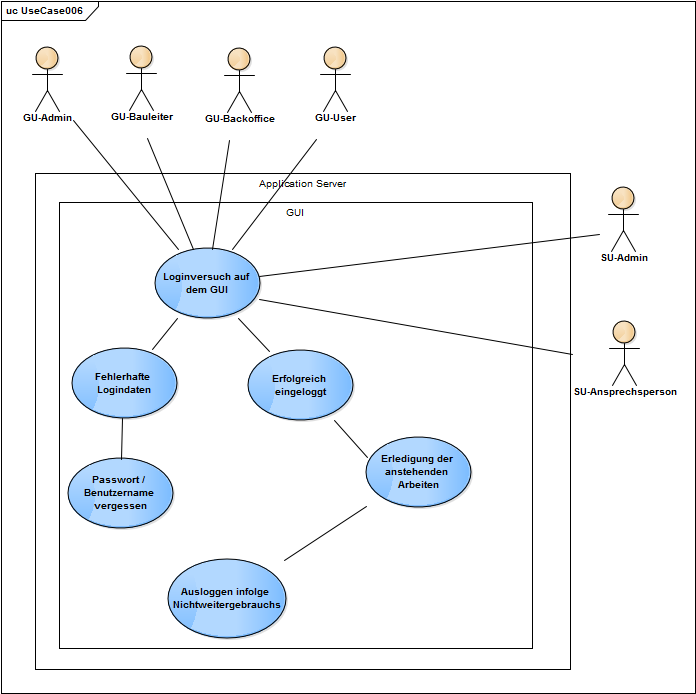
### UseCase005 Aktivitätsdiagramm

## UseCase006

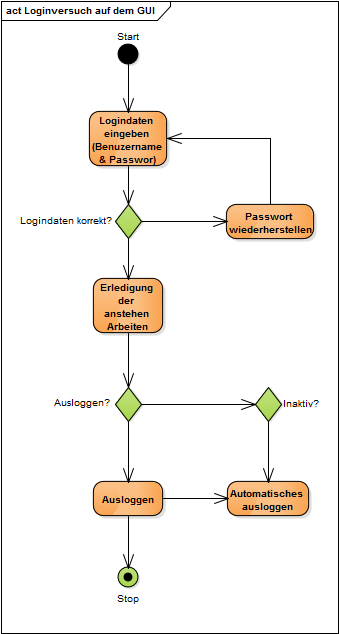
### UseCase006 Beschreibung

|  |  |  |
| --- | --- | --- |
| UseCase006 | User logt sich auf UI ein | |
| Autor | Cihan Demir | |
| Ziel | Beliebige User können sich anhand eines graphical user interface’s einloggen. |
| Kategorie | Primär |
| Vorbedingung | User muss gültige Logindaten (Username, Password) aufweisen. Zusätzlich muss eine aktive Verbindung zwischen UI und Applikationsserver bestehen. |
| Nachbedingungen Erfolg | Jener User kann anhand den gegebenen Rechten in die Datenbank einsehen und kann die betreffenden Daten nach Bedarf und Möglichkeit verwalten. |
| Nachbedingungen Fehlschlag | Zugang zur Applikation / Datenbank wurde verweigert. |
| Akteure | GU-Admin, GU-Bauleiter, GU-Backoffice, GU-User, SU-Admin, SU-Ansprechperson |
| Auslösendes Ereignis | Beliebiger User muss sich aufgrund eines Geschäftsfalles mit der Applikation / Datenbank verbinden. |
| Szenario 1 |  |
| Beschreibung | 1. Der User loggt sich anhand des user interface’s ein. 2. Der User erledigt die anstehenden Arbeiten. 3. Der User loggt sich nach erfolgreicher Erledigung der Arbeiten oder infolge Nichtweitergebrauchs, aus. |
| Szenario 2 |  |
| Auslösendes Ereignis | Beliebiger User muss sich aufgrund eines Geschäftsfalles mit der Applikation / Datenbank verbinden. |
| Beschreibung | 1. Der User versucht sich anhand des user interface’s einzuloggen. 2. Der User kann sich nicht einloggen, da fehlerhafte Logindaten vorliegen. 3. Der User wählt den Link: „Passwort vergessen“ 4. Der User gibt entweder seine E-Mail-Adresse oder den Benutzernamen ein. 5. Der User wählt den Link in seiner E-Mail um sein Passwort zurückzusetzen. 6. Der User loggt sich anhand des user interface’s ein. 7. Der User ändert sein Passwort. 8. Der User erledigt die anstehenden Arbeiten. 9. Der User loggt sich nach erfolgreicher Erledigung der Arbeiten oder infolge Nichtweitergebrauchs, aus. |
| Erweiterung | Der User wird automatisch nach einer gewissen Zeit angesichts Inaktivität ausgeloggt. |

### UseCase006 Visualisierung



### UseCase006 Aktivitätsdiagramm

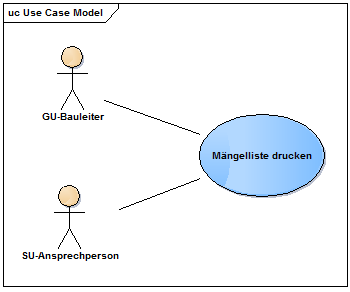


## UseCase007

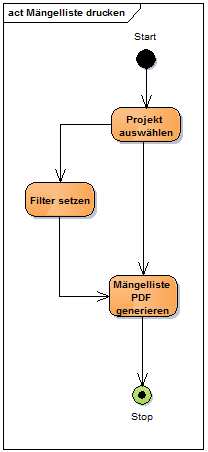
### UseCase007 Beschreibung

|  |  |  |
| --- | --- | --- |
| UseCase007 | Mängelliste ausdrucken | |
| Autor | Sandro Ritz | |
| Ziel | Der Ansprechperson der Subunternehmen kann von seinen Projekten die Mängelliste ausdrucken. |
| Kategorie | Sekundär, der Mangelmanager funktioniert auch ohne entsprechende Funktion. |
| Vorbedingungen Erfolg | Es müssen offene Mängel vorhanden sein. |
| Nachbedingungen Fehlschlag | PDF mit Mängel wird generiert. |
| Akteure | SU-Admin, SU-Ansprechperson |
| Auslösendes Ereignis | SU-Ansprechperson möchte eine Liste mit allen offenen Mängeln ausdrucken. |
| Beschreibung | 1. Die Ansprechperson öffnet ein ausgewähltes Projekt. 2. Durch die Filterfunktion kann die Ansprechperson die gewünschten Mängel anzeigen lassen. 3. Ein Klick auf „Drucken“ generiert ihm eine PDF Datei mit den vorhin ausgewählten gefilterten Mängeln. |
| Erweiterung | 1a Auswählbar ob ein PDF geöffnet oder direkt gedruckt werden soll. |

### UseCase007 Visualisierung



### UseCase007 Aktivitätsdiagramm

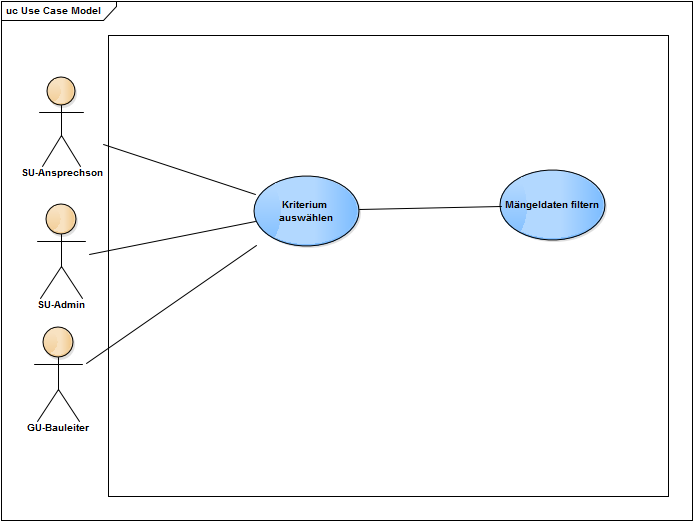


## UseCase008

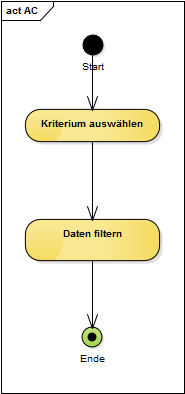
### UseCase008 Beschreibung

|  |  |
| --- | --- |
| UseCase008 | Mängeldaten filtern |
| Autor | Mike Iten |
| Ziel | Die UIs müssen die Mängeldaten nach diversen Kriterien filtern können |
| Kategorie | Sekundär, der Mangelmanager funktioniert auch ohne entsprechende Funktion. |
| Vorbedingungen | Es müssen Daten vorhanden sein. |
| Nachbedingungen Erfolg | Die Daten werden korrekt nach dem gewählten Kriterium gefiltert. |
| Nachbedingungen Fehlschlag | Die Daten werden falsch gefiltert. z.B. werden bereits erledigte Mängel angezeigt bei einer Suche mit nicht erledigten Mängel. |
| Akteure | GU-Bauleiter, SU-Admin, SU-Ansprechperson |
| Auslösendes Ereignis | Die Mängeldaten müssen nach irgendeinem Kriterium gefiltert angezeigt werden. |
| Beschreibung | Sinnvolle Kriterien müssen eingebaut werden um z.B. alle Mängeldaten für ein Projekt nach Fälligkeitsdatum sortiert anzuzeigen oder z.B. alle Mängel für ein Projekt anzuzeigen, die noch nicht erledigt wurden. |

### UseCase008 Visualisierung



### UseCase008 Aktivitätsdiagramm



# Klassendiagramme

# Deyploment-Infos

1. Starten Sie die Datei „RMI-Server.jar“ auf dem Server.
2. Kopieren Sie die „Datei MangelManager.war“ in das webapps Verzeichnis der Tomcat Installation.
3. Stellen Sie sicher, dass Ihr Tomcat lib-Verzeichnis die JAX-WS library enthält.
4. Starten Sie die Datei „startup.bat“ im Tomcat Verzeichnis.
5. Je nachdem ob Sie intern oder extern auf den Mangel-Manager zugreifen möchten öffnen Sie im ersteren Fall die Datei „Client-Inter.jar“ und im letzteren Fall die Datei „Client-Extern.jar“.

# TDD und JUnit

# Funktionale Test’s

|  |  |
| --- | --- |
| Test ID | 1 |
| Beschreibung | Logininformationen wurden nicht korrekt angegeben |
| Resultat | Fehlermeldung und erneuter Loginversuch. |

|  |  |
| --- | --- |
| Test ID | 2 |
| Beschreibung | Ein Mangel wurde erfasst. |
| Resultat | Der Mangel wird für das jeweilige Projekt angezeigt und der Bauleiter kann diese korrekt einsehen. |

|  |  |
| --- | --- |
| Test ID | 3 |
| Beschreibung | Ein Mangel wurde abgearbeitet. |
| Resultat | Der Bauleiter kann den Mangel als erledigt kennzeichnen. |

|  |  |
| --- | --- |
| Test ID | 4 |
| Beschreibung | Ein neues Projekt wird hinzugefügt |
| Resultat | Jeder kann das neue Projekt begutachten, der Bauleiter aber mit erhöhter Berechtigung. |

|  |  |
| --- | --- |
| Test ID | 5 |
| Beschreibung | Dem Projekt wird ein Subunternehmen zugeteilt |
| Resultat | Das Subunternehmen und zusätzlich die Ansprechperson wird hinzugefügt. |

|  |  |
| --- | --- |
| Test ID | 6 |
| Beschreibung | Ein Mitarbeiter kann die Mängel nicht bearbeiten |
| Resultat | Nur der Bauleiter hat die Berechtigung die Mängel zu bearbeiten oder als erledigt zu kennzeichnen. |

|  |  |
| --- | --- |
| Test ID | 7 |
| Beschreibung | Ein neues Projekt wurde erfasst |
| Resultat | Das Projektobjekt, der Projekttyp und das Datum ist klar ersichtlich, sowie die Adresse des Projekts und der Bauherr wurden angegeben. |

|  |  |
| --- | --- |
| Test ID | 8 |
| Beschreibung | Der Bauleiter will alle Mängel anschauen |
| Resultat | Alle Mängel und Arbeiten des Bauleiters sind zeitlich gegliedert und aufgeführt. |

|  |  |
| --- | --- |
| Test ID | 9 |
| Beschreibung | Der Bauleiter will das Subunternehmen kontaktieren |
| Resultat | Die involvierten Subunternehmen sind klar ersichtlich und die Ansprechspersonen sind zeitlich aufgeführt. |

|  |  |
| --- | --- |
| Test ID | 10 |
| Beschreibung | Der Bauleiter will das Subunternehmen kontaktieren |
| Resultat | Die involvierten Subunternehmen sind klar ersichtlich und die Ansprechspersonen sind zeitlich aufgeführt. |

|  |  |
| --- | --- |
| Test ID | 11 |
| Beschreibung | Der Bauleiter des Projektes muss ausfinding gemacht werden |
| Resultat | Es kann auch möglich sein, dass ein Projekt durch mehrere Bauleiter betreut wird. Jedoch ist der Hauptbauleiter ersichtlich und dessen Adresse aufgeführt. |

|  |  |
| --- | --- |
| Test ID | 12 |
| Beschreibung | Der Bauleiter möchte das Projekt eines Kollegen einsehen |
| Resultat | Dies ist nicht möglich, da jeder Bauleiter nur sein eigenes Projekt einsehen kann. |

|  |  |
| --- | --- |
| Test ID | 13 |
| Beschreibung | Das Subunternehmen begutachtet das Projekt |
| Resultat | Jegliche Mängel sind aufgeführt und können auch über das Subunternehmen quittiert werden. |

|  |  |
| --- | --- |
| Test ID | 14 |
| Beschreibung | Die Meldungen werden eingesehen |
| Resultat | Jeder Mangel kann mehrere Meldungen beinhalten, die durch das Generalunternehmen oder Subunternehmen erfasst werden können, sie sind für den Bauleiter immer sofort zur Verfügung. |

|  |  |
| --- | --- |
| Test ID | 15 |
| Beschreibung | Der Mangel wird vom Subunternehmen abgearbeitet |
| Resultat | Der Bauleiter kann den Mangel einsehen und bestätigen, dass dieser abgearbeitet wurde und kann demnach den Mangel als erledigt kennzeichnen. Ist dies nicht der Fall, dann kann er den Mangel erneut an das Subunternehmen schicken. |

|  |  |
| --- | --- |
| Test ID | 16 |
| Beschreibung | Die Meldung soll verändert werden |
| Resultat | Niemand kann die Meldung nach dem Speichern verändern. |

|  |  |
| --- | --- |
| Test ID | 17 |
| Beschreibung | Die Mängel sollen nach einem Kriterium gefiltert werden |
| Resultat | Der Bauleiter möchte alle Mängel bezüglich der Hauswand aufgeführt haben. |

|  |  |
| --- | --- |
| Test ID | 18 |
| Beschreibung | Der Bauleiter möchte alle Mängel in Papierform haben |
| Resultat | Der Bauleiter kann alle Mängel in Listenform anzeigen lassen und diese dann ausdrucken. |

# DB – Dokumentation

# Beiträge pro Projektmitglied (2015\_FS\_INM21\_Individuelles\_Portfolio\_Name\_Vorname.xlsx)

# Source-Code von selber beschriebenem Code

# Weitere Dokumentationen