

# Zwischenbericht Gruppe 2

## **Projektauftrag:**

Erstellung einer Softwarelösung für den Kunden, zur Verwaltung von Configuration-Items (CI).

## **Gruppenmitglieder:**

- Daniel Reimann
- Dihpol Gianna
- Dimitri Wagner
- Johanna Gröll
- Laura Bieberle
- Marcel Mahlfeld
- Max Schneider
- Pascal Jung
- Til Wohlmuth

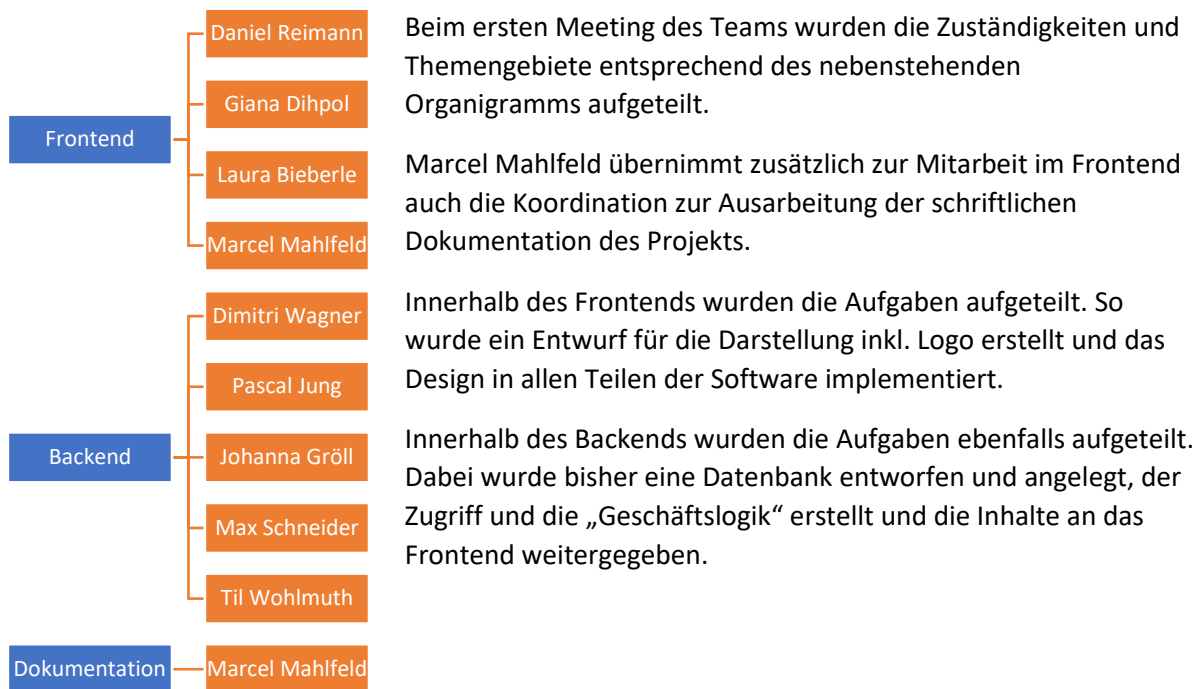
## **Anforderungen an die Softwarelösung:**

- Erstellen, bearbeiten und löschen von CI-Typen und anlegen von CI-Records durch den Benutzer ermöglichen.
- Benutzerfreundliche Oberfläche zum einfachen und schnellen Erstellen.
- Feste Zuordnung von Attributen zu CI-Typen, die beim Erstellen von CI-Records zwingend befüllt werden müssen.
- Optional sollen weitere, individuelle Attribute einem CI-Record hinzugefügt werden können.
- Anzeigen von allen CI-Typen und den Records in der Benutzeroberfläche.
- Filterkriterien und oder eine Suchfunktion sollen implementiert werden.

## **Struktur der Software:**

- Auf Grund von Vorkenntnissen im Bereich Softwareentwicklung mit Java, wurde die Entscheidung getroffen, das Backend mit Java zu programmieren. Zur Vereinfachung wurde sich für das Framework Spring entschieden, da dieses den Zugriff auf die Datenbank und das Darstellen der Inhalte auf HTML-Seiten erleichtert. Für die Darstellung der Daten soll webbasiert abgewickelt werden. Dabei wird das Frontend mit HTML und CSS entwickelt und anschließend vom Backend, unter Zuhilfenahme von Thymeleaf, mit den entsprechenden Daten befüllt. Thymeleaf ist eine Engine, die das Darstellen von Daten auf HTML-Seiten erleichtert und sehr gut mit Spring kompatibel ist.
- Für die persistente Datenhaltung fiel die Entscheidung auf eine relationale Datenbank. Dies hat den Vorteil, dass die Komponenten der Software (CI-Typen, CI-Records und die entsprechenden Attribute) sehr einfach in Relation gesetzt werden können. Außerdem erleichtert es den schnellen und gezielten Zugriff auf einzelne Instanzen. Aktuell ist geplant, eine H2 Datenbank einzusetzen, da diese in die Software implementiert werden kann und keine separate Erstellung und Verwaltung einer Datenbank nötig ist.

### Projektorganigramm:



### Projektorganisation:

- Weekly mit allen Projektbeteiligten zur Abstimmung und Besprechung (was wurde geschafft, was steht als nächstes an)
- Dokumentation der besprochenen Punkte in einem Protokoll, das für alle zur Verfügung steht.
- Individuelle Meetings innerhalb der „Sub-Teams“ Frontend und Backend.
- Agiler Ansatz wurde verfolgt, um Probleme und Unklarheiten bereits vor dem Entstehen zu diskutieren.

### Projektphasen:



Aktuell befinden wir uns in der Erstellung und beginnen in den nächsten Tagen mit der Zusammenführung von Front und Backend.

Danach werden alle Funktionen getestet und ggf. noch kleine Anpassungen vorgenommen.

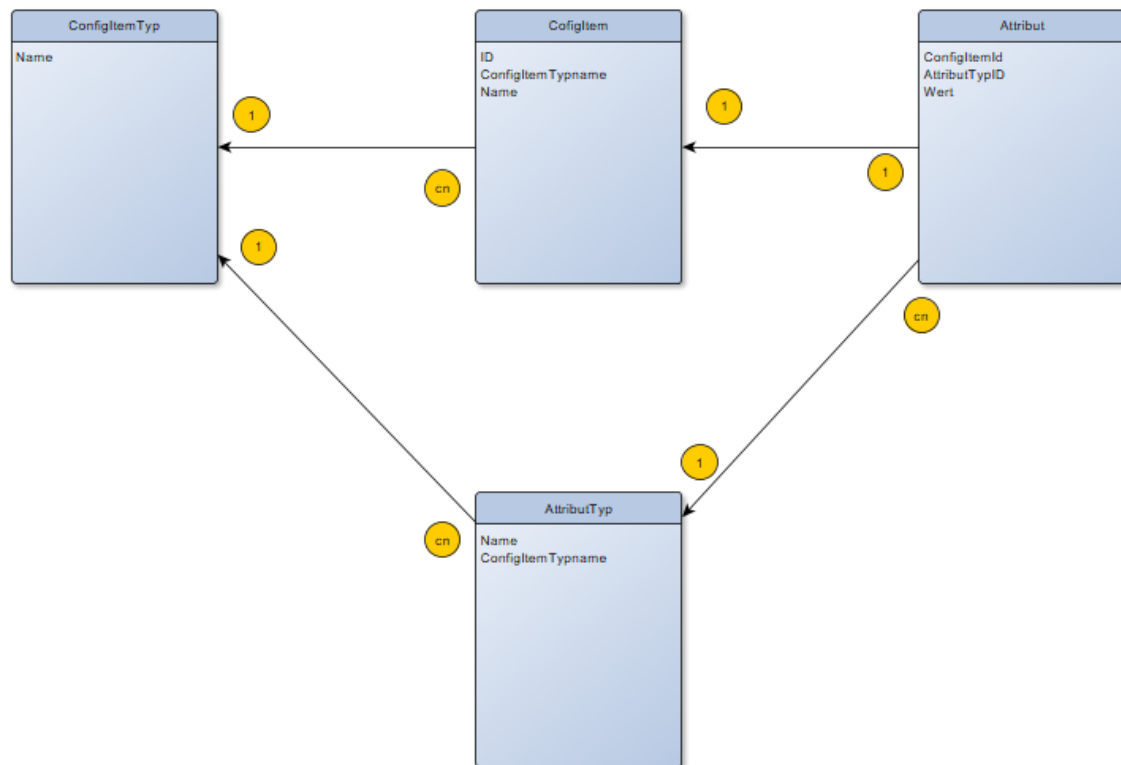
Die Dokumentation wird parallele zum gesamten Projektverlauf erstellt.

## Anhang:

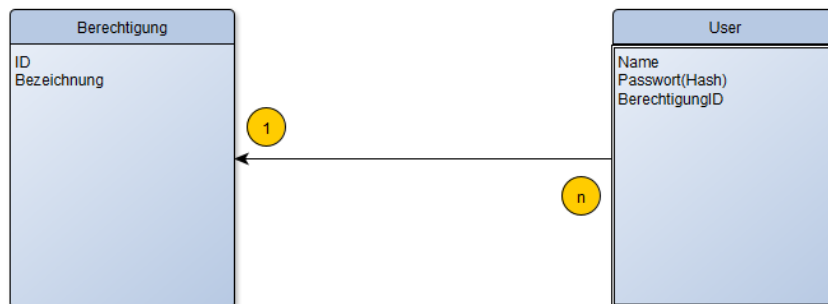
### Anhang1: Protokoll der wöchentlichen Besprechungen

| Protokoll Fallstudie zero 1 one |  |  |            |                  |      | 17.09.2020            |
|---------------------------------|--|--|------------|------------------|------|-----------------------|
| ID                              | Arbeitspaket                             | Resolution   | Due Date   | Status           | Type | Assigned to           |
| 1                               | Anforderungen                            | <ul style="list-style-type: none"> <li>- Konfigurations Management System CMS</li> <li>- CI-Type Jeden beliebigen erstellen ?</li> <li>- Attribute Name,Version,Seriennummer, Hersteller, Standort</li> <li>- optional beziehungen zwischen CI-Typen</li> <li>- Balken-,Kreisdigramm (Übersichtdiagramme)</li> <li>- Ein-/Ausloggen mit Springframework</li> </ul> <p>Configuration Item Records<br/>Instanzen von CI-Typen / tatsächliche CIs</p>   | 14.09.2020 | work in progress |      | all                   |
| 2                               | Regeltermin                              | - Donnerstag nach der Uni 10-20 min  | 14.09.2020 | -                |      | all                   |
| 3                               | Infos                                    | <p>Maven</p> <p>Registrierung neuer User /Rechteverwaltung</p> <p>Benutzergruppen = Berechtigungen</p> <p>Kommentar in Code !!!</p> <p>Java (Html,css,SQL),Js</p> <p>- IdoIT tool anschauen! <a href="https://www.i-doit.com/i-doit/">https://www.i-doit.com/i-doit/</a></p> <p><a href="https://www.i-doit.com/support/starter-guide/">https://www.i-doit.com/support/starter-guide/</a></p> <p>-Maven</p> <p><a href="https://draeger-it.blog/maven-einrichten-von-apache-maven/?cn-reloaded=1">https://draeger-it.blog/maven-einrichten-von-apache-maven/?cn-reloaded=1</a></p> <p>Maven: Einrichten von Apache Maven</p> <p>In diesem Tutorial möchte kurz erläutern wie man Apache Maven einrichtet. Dieses Tutorial wird benötigt da ich in anderen Tutorials für die Entwicklung Apac</p> | 14.09.2020 | work in progress |      | all                   |
| 4                               | Funktionalität                           | <p>-Configuration Item Records inkl. Attribute anzeigen, erstellen, verändern, löschen...</p> <p>-Optional: Beziehungen zwischen Configuration Item Records</p> <p>- Einfache und schnelle Erstellung</p> <p>- Configuration Item Typen inkl. Attribute anlegen, löschen ...</p> <p>Was passiert mit CI Records eines bestimmter CI Typs, wenn der CI Typ gelöscht wird?</p> <p>Anzahl an CI Records je CI Typ</p> <p>Optional: Grafiken, z. B.</p> <ul style="list-style-type: none"> <li>-Balkendiagramm für Vergleiche zwischen unterschiedlichen CI-Typs</li> <li>-Kreisdigramm für Anteil eines CI-Typs an der Summe aller CI Records</li> </ul>  | 14.09.2020 | work in progress |      | all                   |
| 5                               | Funktionalität Login                     | <p>Benutzername und Passwort abfragen</p> <ul style="list-style-type: none"> <li>- Zugang nach erfolgreicher Authentifizierung</li> <li>-Berechtigungen in der Software entsprechend des Benutzers</li> <li>-Beenden der Sitzung durch Logout (Tab,Browser und Logoutbotten</li> <li>- reChaptcher</li> </ul>  | 24.09.2020 | work in progress |      | all                   |
| 6                               | Funktionalität Benutzer/Rechteverwaltung | <p>Benutzer anlegen:</p> <ul style="list-style-type: none"> <li>- Benutzernamen und initiales Passwort vergeben</li> <li>- Rechte vergeben</li> <li>Benutzer löschen</li> <li>Benutzer verwalten</li> <li>- Rechte ändern</li> <li>- Ggf. Benutzernamen, Passwort... ändern</li> <li>Welche Benutzergruppen bestehen?</li> <li>Welcher Benutzergruppen haben welche Berechtigungen?</li> <li>- als .exe</li> </ul>   | 24.09.2020 | work in progress |      | all                   |
| 8                               | Projektprotokoll                         | <p>Umfang: 2,5 ECTS-LP entspricht ca. 5 Textseiten (ohne Abbildungen, Tabellen, Verzeichnisse, Anhänge, ...) * Anzahl der Gruppenmitglieder = ca. 45 Textseiten (bei 9 Gruppenmitgliedern)</p> <p>Marcel mit Unterstützung aller</p>   | 24.09.2020 | work in progress |      | all                   |
| 9                               | Frontend                                 | <p>Ginna,Laura,Daniel,Marcel</p> <ul style="list-style-type: none"> <li>- Vorteile zwischen HTML und Java Gui auf Erfahrung</li> <li>- Termin für Frontend Montag 21.09.2020</li> <li>- <a href="https://www.youtube.com/watch?v=F7krXlSpwNA&amp;t=1s&amp;ab_channel=CodingMarket">https://www.youtube.com/watch?v=F7krXlSpwNA&amp;t=1s&amp;ab_channel=CodingMarket</a></li> <li>- <a href="https://www.youtube.com/watch?v=URf1YcfigTU&amp;ab_channel=CodingMarket">https://www.youtube.com/watch?v=URf1YcfigTU&amp;ab_channel=CodingMarket</a></li> </ul> <p>Seiten : Profil ; CI-Typen; CI-Record</p>   | 24.09.2020 | work in progress |      | GI,LA,DR,MC           |
| 10                              | Backend                                  | <p>Johanna, Pascal,Til,Dimi,Max</p> <ul style="list-style-type: none"> <li>- Termin wird mit Pascal , Max Johana nach Uni stattfinden</li> </ul>   | 24.09.2020 | work in progress |      | JOH,PCJ,TIL, DIMI,MAX |

## Anhang 2: Entity-Relationship-Modell für die Datenbank, zur Ablage der CI-Typen/-Records



## Anhang 3: Entity-Relationship-Modell für die Datenbank, zur Ablage von Benutzerdaten



### Anhang 3: Übersicht der Struktur der Software und der Inhalte der jeweiligen Seite

#### 1. Login

Benutzername (Textfeld)

Passwort (Passwortfeld)

Button „Login“

(Dokumentation)

#### 2. Dashboard

Navigation Bar

Suchleiste(filter)

- **Ci-Typen**

- Typ A

- Record 1
    - Record 2
    - Record n
    -

- Typ B

- Record 1
    - Record 2
    - Record n

- Typ C

- Typ N

#### 3. Ci-Typ anlegen

Name (Textfeld)

Mit Benachrichtigung, dass Eintrag erfolgreich angelegt wurde und Auszug der Table

Attributtyp vordefinieren Name (Textfeld)

#### 4. Ci-Record anlegen

Ci-Typ (Drop-Down mit allen Ci-Typen)

Ci-Record-Name (Textfeld)

Alle vordefinierten Attributtypen Wert (Textfeld)

**(Unterseiten:) Wenn benötigt**

##### **I. Attributtyp erstellen**

Name (Textfeld)

#### 5. Profil

Passwort ändern (mit Angabe des alten PW)

Altes Passwort (Passwortfeld)

Neues Passwort (Passwortfeld)

Neues Passwort (Passwortfeld)

Bestätigen (Button)

Benutzer anlegen (Wenn Adminrechte vorhanden)

Benutzername (Textfeld)

Passwort (Passwortfeld)

Bestätigen (Button)

Benutzer löschen (Wenn Adminrechte vorhanden)

Auswahlfeld (Drop Down mit allen Benutzern die nicht Adminrechte haben)

Bestätigen (Button)

## **6. Hilfe**

„Leitfaden“ Dokumentation **statisch**

Lets start **statisch**

Impressum Copyright **statisch**