# Inhalt:

01 - Erstellung des fertigen Entwurfes

02 - **Abänderung des Entwurfes aufgrund einer Änderung des Arbeitsauftrages**

**03 -** Fertigstellung der GUI sowie der Klasse CSV ReadWrite

04 - Verbinden der Logik mit der GUI

Ergebnisprotokolle

Netzwerkverwaltungstool

Brian Koduan, davis fröse, marco d‘addona

**Ergebnis Protokoll Nr. 01:**

„Erstellung des fertigen Entwurfes“

Aktueller Stand:

Alle benötigten Klassen wurden erstellt. Somit existiert nun ein fertiger Entwurf

Es existieren nun folgende Klassen incl. Assoziationen und Vererbungen:  
Gebäude, Raum, Komponente, Switch, Router, PC, Laptop, JSONReadWrite, Main, MainForm.

Testfälle:

Es können Objekte von allen Klassen generiert werden, somit sind alle Klassen funktionsfähig.  
Tester: alle Gruppenmitglieder.

Weitere Vorgehensweise:

* Erstellung des grafischen User Interfaces (GUI)
* Implementierung der Methoden Read und Write in der Klasse JSONReadWrite zum Auslesen und Schreiben (befüllen) einer JSON Datei.
* Logik mit der GUI verbinden.

**Ergebnis Protokoll Nr.02**

**„Abänderung des Entwurfes aufgrund einer Änderung des Arbeitsauftrages“**

Aktueller Stand:

Wir haben erfahren, dass der JSON Dateiformat nun noch nicht für den Austausch von Daten zwischen den Computersystemen verwendet werden soll.

Aufgrund dessen mussten wir uns in XML einarbeiten.

Da wir nur sehr wenige Kenntnisse in XML haben, haben wir aufgrund der knappen Zeit für CSV als Datentyp um entschieden. Dieser Datentyp war auch ursprünglich für diesen Zweck angedacht.

Es wurde die Klasse CSVReadWrite erstellt sowie eingearbeitet, wie man CSV Dateien mit Java liest und schreibt.

Dies hat sehr viel Zeit gekostet, jedoch sind wir nicht viel weiter seit Ergebnis Protokoll Nr. 01

Testfälle:

Es kann ein Objekt der Klasse CSVReadWrite erstellt werden. Diese Klasse ist funktionsfähig, hat jedoch noch keine Methoden.

Weitere Vorgehensweise:

* Erstellung des grafischen User Interfaces (GUI)
* Implementierung der Methoden Read und Write in der Klasse CSVReadWrite zum Auslesen und Schreiben (befüllen) einer CSV Datei.
* Logik mit der GUI verbinden.

**Ergebnis Protokoll Nr. 03**

„Fertigstellung der GUI sowie der Klasse CSVReadWrite“

Aktueller Stand:

Die GUI wurde fertiggestellt sowie die Methoden der Klasse CSVReadWrite implementiert.

Man kann nun CSV Dateien erstellen mit hard-gecodeten Strings.

Ebenso wurde die GUI fertiggestellt.

Testfälle:

Es wurde erfolgreich eine CSV Datei generiert mit hard-gecodeten Strings.

Probleme:

Leider hatten wir vergessen, die Klasse „JSONReadWrite“ zu löschen, welche eine Bilbiothek benötigt welche nicht mehr installiert war.

Es hat uns viel Zeit gekostet, bis wir das Problem gelöst hatten.