# Anwendungsfallbeschreibungen

## Hardware hinzufügen:

* Der Nutzer füllt mit Hilfe der GUI- Felder die Attribute der zu beschreibenden Netzwerkkomponente/Hardware und fügt diese Netzwerkkomponente/Hardware durch Klick auf den entsprechenden Button der sichtbaren Tabelle hinzu.

## Hardware bearbeiten:

* Der Nutzer wählt aus der Liste eine schon bestehende Hardware aus und füllt mit Hilfe der GUI-Felder die Attribute der zu bearbeitenden Hardware und ändert diese Hardware durch Klick auf den entsprechenden Button in der Tabelle.

## Hardware löschen:

* Der Nutzer wählt aus der Liste eine schon bestehende Hardware aus und löscht diese Hardware durch Klick auf den entsprechenden Button aus der Tabelle.

## Hardware filtern:

* Der Nutzer filtert die dargestellten Daten mit Hilfe von zwei Comboboxen nach Raum und Gebäude und aktiviert den Filter über einen Button. Die Tabelle zeigt nun nur noch die zum Filter passenden Einträge.

## Hardware einlesen:

* Der Nutzer wählt über einen Dialog eine einzulesende Datei aus, die dann in der Tabelle dargestellt wird.

## Hardware speichern:

* Der Nutzer möchte die Tabelle in einer Datei abspeichern. Über einen Dialog wählt er den Speicherort und –Namen aus.

# Substantivmethode

## Hardware

* Das Objekt Hardware verwaltet unsere Netzwerkkomponente. Unter Hardware fallen noch weitere Dinge, somit leicht erweiterbar ohne die Klasse umbenennen zu müssen.

## Attribute

* Keine Klasse, sondern die Attribute der Klasse Hardware

## Feld (TextBox), Button, Combobox, Tabelle (ListView)

* Existiert bereits

## Raum/Gebäude

* Wenn Raum und Gebäude Attribute und/oder weitere Funktionalitäten haben sollen, dann würde sich eine Klasse hierfür anbieten. Dies lohnt in unserem Fall allerdings nicht.

## Dialog

* OpenFileDialog und SaveFileDialog, bereits implementiert