Mission-Cucumber Component Design

# Health Component

## Beschreibung

Managet Leben, Schaden, Heilung und Tod einer Unit.

## Eigenschaften

* Maxmimal Leben
* Aktuelles Leben

## Gefangene Events

* TakeDamage (read-only)
  + Wird von gegenerischer Attack Component geworfen
  + Event-Object enthält Grundschaden, der zugefügt werden soll
    - Grundschaden kann durch Abilities modifiziert werden (über ModifyIncomingDamage Event)
* GetHealing (read-only)
  + Wird von einer Ability geworfen
  + Event-Object enthält Grundheilung, die zugefügt werden soll
    - Grundschaden kann durch Abilities modifiziert werden (über ModifyIncomingHealing Event)

## Geworfene Events

* ModifyIncomingDamage
  + S.o.
* ModifyIncomingHealing
  + S.o.
* DamageTaken (read-only)
  + Enthält im Event-Object den tatsächlich genommenen Schaden und verbleibendes Leben
  + Nützlich für Schadensanzeige o.Ä.
* HealingReceived (read-only)
  + Enthält im Event-Object den tatsächlich erhaltene Heilung und verbleibendes Leben
  + Nützlich für Heilungsanzeige o.Ä.
* UnitDied (read-only)
  + Wenn aktuelles Leben unter 0 sinkt
  + Sollte Dispose aller Abhängigkeiten und der Unit selbst auslösen

# Sight Component

## Beschreibung

Prüft, ob BaseUnits in Sicht sind und verteilt diese Liste auf andere Componenten

## Eigenschaften

* Sichtweite
* Sichtradius

## Gefangene Events

## Geworfene Events

* UnitsInSight (read-only)
  + Wird nur geworfen, wenn min. eine BaseUnit im Sichtfeld sind
  + Eventobjekt enthält die Liste aller gesehenen BaseUnits
* ModifyUnitsInSight
  + Wird vor UnitsInSight geworfen, um z.B. Blind-Debuff die Liste der gesehenen Units leeren zu lassen
* IsVisble
  + Wird an alle BaseUnits geworfen, die im Sichtbereich sind
  + Bsp: Unsichtbare-Unit (Stealth) gibt false zurück

# Target Component

## Beschreibung

Wählt für Abilitie Components das passende Target aus. Nicht alle Abilitie Components brauchen eine Target Component (z.B. Attack Modiefier wie Critical Strike brauchen keine Target Component) Kommuniziert mit Ability Component über ein direktes Interface, damit jede Ability Component nur ihre eigenen Targets empfängt

## Eigenschaften

* Evtl. Spezial-Eigenschaften wie Reichweite o.Ä.

## Gefangene Events

* UnitsInSight (read-only)

## Geworfene Events

* NewTarget (read-only)
  + Echtes Event !!!
  + Wird beim Target Wechsel ausgelöst und gibt Ability Component ihr Target
  + Abitlity speichert Target bei Bedarf selbst ab und nutzt eigene Update Funktion, um etwas mit dem Target zu tun

# Ability Component

## Beschreibung

Component für alles (Attack, Movement, Critical-Strike, Evasion, Element-Schaden,…)

## Eigenschaften

* Bei Bedarf: Target Component

## Gefangene Events

* NewTarget (read-only)

## Geworfene Events

* Visuelle Darstelung (read-only)
  + Für die visuelle Darstellung, sobald die Ability etwas tut
  + Animation wird über Typ der Component ausgewählt, welcher im Event Object enthalten ist

# Buff Component

…