# Usecase: Request Status ( skal deles op i flere mindre usecases)

(Denne usecase skal senere deles op i adskillige usecases, da den indeholder flere usecases)

## Aktører:

Primære: Master unit. (Kommunikerer med slave units, og styrer kommunikationen.)

Sekundære: Slave unit ( kommunikere med master omkring statusrequests)

Referencer: Ingen

Samtidige forekomster: ingen

Preconditions: System er online og klar til brug, counters til Request Status er initieret, bruger initierer statud request ved tryk på ”knap”.

Resultat: Masterenhed viser status over alle enheder på GUI

1. Master tjekker “master protokol”
   1. Master gennemløber protokol indtil den når til Request Status punkt.
2. Master tjekker register over kendte enheder.
   1. Extension 1: Alle enheder gennemløbet
   2. Master inkrementere counter for gennemløbne enheder
3. Sender status-request til den næste slave enhed.
   1. Master afventer svar retur fra enhed i ”svartid”
      1. Extension 2: Manglende svar fra slave unit
4. Slave modtager besked fra master
   1. Extension 3: checksum fra master passer ikke
   2. Slave Sender svar på request til master
5. Master modtager svar fra slave enhed
   1. Extension 2: checksum fra slave passer ikke
   2. Extension 2: Modtager Gensendelsesrequest
6. Master viser enhedsstatus på GUI
7. Gå til punkt 2

Extension 1: Alle enheder gennemløbet

1. Master checker om alle statusrequests er fejlede
   1. if true == melder fejl på kommunikationslinjen
2. Nulstiller alle counters i statusrequest forløbet
3. Master går til næste punkt i masterprotokollen

Extension 2: Fejl i kommunikationen mellem master og slave

1. inkrementere counter for antal kommunikationsforsøg til slave enheden
2. tjek counter > 3
   1. if (true) == melde fejl som enhedsstatus på enheden via GUI
   2. if (false) == Gensend Request.
3. gå til punkt 3.1

Extension 3: Checksum fra master passer ikke

1. Slave unit sender anmodning til master om gentagelse af besked.