**Referat vejledermøde #3**

**Dato: Tirsdag d. 29/03-2016**

**Tid: 14:15**

**Sted: Kathrines Kælder**

**Deltagere: Alle.**

**Ikke fremmødte: Stefan, Mikkel, Tonni og Martin**

**Dagsorden**

1. Valg af referant

* Anders B. Knudsen

1. Gennemgang af modtaget review materiale

* Computer software og styreboksen er ikke aktør. Alt hvad vi ikke laver skal være med: Bruger, lamper osv.
* Enheder skal specificeres.
* Kunden er ikke interesseret i at det er X.10. Bliver for detaljeret. Kommer først i næste fase.
* Den gruppe, som har lavet review af vores, har gjort det godt. Henning nævner samme ting.
* Snak omkring containers, og huske at oplysningerne skal beskrive i use cases.
* Talværdier omtales typisk i yderligere tekniske krav.
* Ikke funktionelle: I stedet for at skrive 60 bit/s, så er kunden mere interesseret i at vide hvor langt tid det tager.
* GUI. Det er godkendt blot at skrive antal knapper i nuværende fase.
* Alle linjer markedet med rødt i yderlige tekniske krav, skal fjernes.
* ”Skal kommunikerer via USB”. Godkendt at den står her. BR er tilfreds så længe at de ved at det er via USB, men er ligeglad med den tekniske del.
* Godkendt at MTBF er med %.
* Huske at ændrer accept-testen efter at kravspefikationen bliver ændret.
* Accept-testen. Husk at skrive specifikt hvad den forkerte/rigtige kode er.

1. Evt.
   1. Evt lav en matrix – Hvilken CPU er en del af hvilken use-case
      1. Brug den matrix til når der skal laves sekvensdiagram
   2. Der skal være en controller klasse for hver usecase der indgår I en given CPU.
      1. Hver controller klasse skal have ét sekvensdiagram.
      2. Brug sekvensdiagram til at lave ‘fully dressed.
   3. Ud over controller klasser skal der være utility som indeholder main. Den starter de andre controller klasser.
   4. Grænseflade til omverdenen – Ex. Mellem PC og styreboks vil der være forbindelse med USB – Der skal være en protocol
   5. Beskrivelse af protocol mellem enhed og styreboks etc.
   6. Boundary klasse indeholder funktioner der gør at man kan få informationer udefra.
      1. 1 boundary klasse per grænseflade
   7. Data der skal gemmes => Container klasse
   8. Når alle er bestemt kan de tegnes, og derefter laves sekvensdiagram.
   9. Sekvensdiagram kan bagefter bruges til at finde de metoder der skal være I de enkelte klasser.
   10. **!!!!!!!MEGET VIGTIGT MED ET KLASSEDIAGRAM PER CPU!!!!!!**