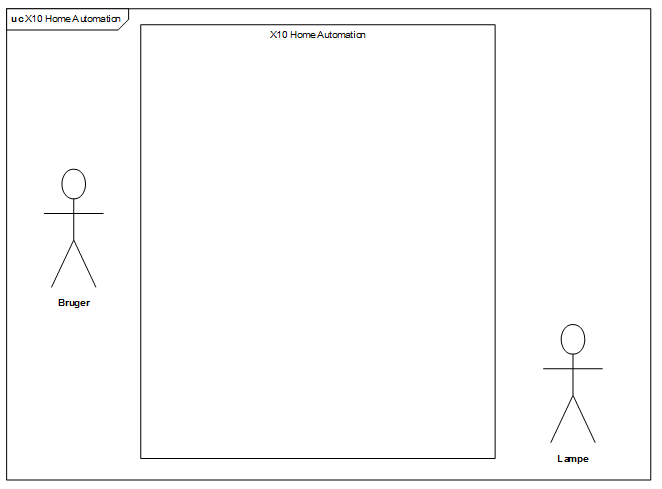
# Krav(TF)

Ud fra en analyse af systembeskrivelsen, samt projekt afgrænsning, er der i gruppen blevet udarbejdet en kravspecifikation indeholdende både funktionelle og ikke funktionelle krav. Med kravspecifikation er der udarbejdet 9 use cases, som beskriver brugerens interaktion med systemet.

## Aktør-kontekst

På Figur 1 ses vores aktør diagram, der viser de aktører der har indflydelse på systemet.



Figur - aktør kontekst diagram

## Aktør-Beskrivelser

### Bruger

Brugeren er den aktør der ønsker at benytte systemet. Brugeren har kendskab til koden til kodelåsen der kræves for konfiguration og betjening af systemet, og er den aktør der er ansvarlig for konfiguration af systemet.

### Lampe

Lampe er en sekundær aktør, der er forbundet til systemet der vil styre om lampen er tændt eller slukket.

## Funktionelle krav

De funktionelle krav beskriver systemets ønskede funktionalitet som en serie af usecases, der specificere brugerens interaktion med systemet ved hjælp af diagrammer.

På Figur 2 ses use case diagrammet, der viser hvilke aktører der interagerer med de enkle use cases.



Figur - Usecase diagram over aktørenes interaktion med de enkelte usecases

Der følger her en kort beskrivelse af de enkelte usecases, ønskes der en fully dressed version af usecasen kan denne findes i projektdokumentationen.

### Usecase 1: Opstart af System

Usecase 1 opstart af system, beskriver det første scenarie som brugeren møder ved anvendelse af systemet. Scenariet startes ved at brugeren starter det medfølgende PC software, hvorefter pc softwaren anmoder brugeren om at indtaste koden til systemet. Når brugeren har indtastet den korrekte kode, vil pc softwaren vise forsiden i den grafiske brugerflade.

Udvidelserne til denne usecase sikrer at der kun kan køre 1 instans af softwaren af gangen, og sikrer samtidig at en forkert indtastning af koden håndteres ved at vise en fejlmeddelelse.

### Usecase 2: Status Forespørgsel

Anvendes af brugeren til at få en oversigt over de tilsluttede enheders nuværende status. Dette gøres ved at brugeren trykker ”Opdatere Enhedsstatus”, hvorefter systemet vil vise en opdateret oversigt over status på alle tilsluttede enheder

### Usecase 3: Tilføjelse af enhed

Beskriver hvordan brugeren tilføjer en ny enhed til systemet. Dette gøres ved at brugeren tilslutter den nye enhed til lysnettet, hvorefter enheden oprettes i systemet ved hjælp af den grafiske brugerflade. I forbindelse med oprettelsen af enheden i systemet, vil der blive givet mulighed for at tildele enheden et rum, eller annullere.

Til at håndtere de forskellige indtastningsmuligheder er der en række udvidelser til hovedscenariet som håndtere de scenarier hvor brugeren ikke indtaster et rum, eller håndtering af at enhedsadressen kollidere med en eksisterende enhedsadresse.

### Usecase 4: Fjernelse af enhed

Specificere hvordan brugeren kan fjerne en eksisterende enhed fra systemet via brugergrænsefladen. For at kunne håndtere forskellige indtastningsfejl eller at brugeren ombestemmer sig, er der lavet to udvidelser der håndtere annullering fra brugerens side, samt at brugeren ikke har valgt en enhed der ønskes fjernet fra systemet.

### Usecase 5: Rediger Enhed

Rediger Enhed giver brugeren mulighed for at ændre enhedens adresse samt hvilket rum enheden er en del af i systemet.

Der er tilføjet en række udvidelser for at give brugeren mulighed for at annullere, samt at vælge kun at ændre rum eller enhedens ID.

### Usecase 6: Ændring af tidsplan

Ændring af tidsplan giver brugeren mulighed for at ændre en enheds tidsplan så en enhed kan indstilles til at tænde og slukke for en lampe på specifikke tidspunkter. Dette gøres ligeledes via den grafiske brugerflade, hvor brugeren får mulighed for at vælge ugedag samt tænd- og sluk tidspunkt for en allerede tilføjet enhed i systemet.

En række udvidelser håndtere ikke korrekt brugerinput samt andre mulige problemer i løbet af hovedscenariet, hvorved brugeren informeres om fejlen, og får mulighed for at fortsætte fra et tidligere stadie af hovedscenariet for at korrigere fejlen.

### Usecase 7: Kør simulering

Kør Simulering beskriver hvordan systemet kan tænde og slukke for lamper der er forbundet til enheder på de bruger angivne tidspunkter.

### Usecase 8: Fjernelse af tidsplan

Usecase 8 specificere hvordan brugeren kan fjerne en tidligere konfigureret tidsplan fra en enhed når denne ikke længe ønskes anvendt i forbindelse med usecase 7.

En kombination af hovedscenariet med en række udvidelser, giver muligheden for at håndtere forskellige fejl scenarier så disse kan håndteres på en brugbar måde, så brugeren ikke oplever at systemet går ned grundet fejl i brugerinput.

### Usecase 9: Tilføjelse af tidsplan

Usecase 9 giver brugeren mulighed for at oprette en ny tidsplan til en eksisterende oprettet enhed. Dette gøres på via den grafiske brugerflade.

Muligheden for forkert brugerinput eller forkert status på systemet håndteres ved hjælp af en række udvidelser der giver brugeren fejlmeddelelser der informere om problemet, og derefter går til det relevante punkt i hovedscenariet.

## Yderligere tekniske krav

De yderligere tekniske krav er udarbejdet i samarbejde i gruppen. Der er fra gruppens side specificeret tekniske krav specifikt for styreboksen og enheden, der definere stik typer, led farver mv.

Derudover er der udarbejdet en række ikke funktionelle krav der definere de kvalitetskrav som der stilles til systemet.

For yderligere oplysninger om de yderligere tekniske krav kan de forefindes i den vedlagte projektdokumentation.