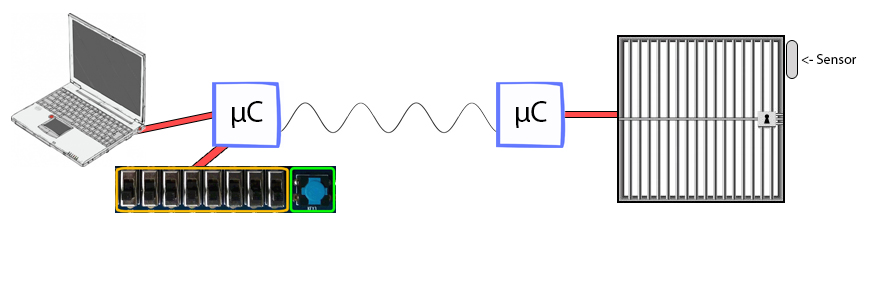
Version 2.0 - 28-04-2014

# Kravspecifikation

”Prison System” er designet til at styre dørene i et fængsel. Systemet skal kunne åbne og lukke alle døre, samt bruge en sensor til at se om dørene er åbne eller lukkede. Det vil sige at der skal være 2-vejs-kommunikation mellem styringscomputeren og dørene. I dette projekt vil der ikke være en rigtig dør, da omfanget af projektet er stort nok i forvejen. I stedet vil der blive simuleret en dør, fx med dioder. På figur 1 ses en illustration af system designet.



Figur : Overordnet system design

Brugeren (fængselsvagtens) vil først blive mødt af et GUI på styringscomputeren (figur 3). Her anmodes brugeren om at logge på systemet ved at indstille nogle switches på DE2 kittet. Efter login skifter GUI til en oversigt over alle døre i systemet (figur 2), hvor man kan styre hver enkelt dør, samt styre alle på en gang.

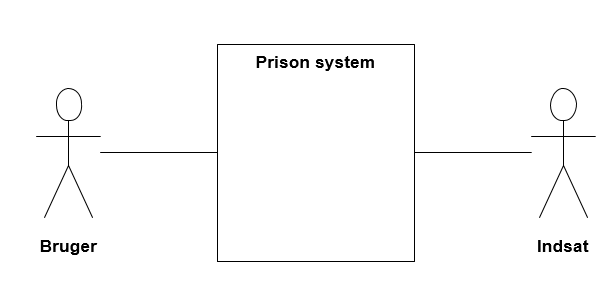
Når styringscomputeren ikke udsender kommandoer til dørene, tjekker den dørene en af gangen efter ændring i deres status, så den kan slå alarm, hvis der er åbnet en dør, uden at der er sendt en ”åben” kommando.



Figur 2: Oversigt over døre i systemet (GUI)

Figur : System login (GUI)

## Aktør-kontekst diagram

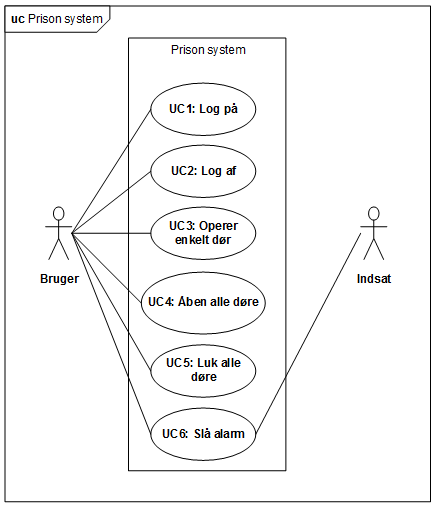


Figur 4 - Aktør-kontekst diagram

|  |  |
| --- | --- |
| **Navn på aktør** | Bruger |
| **Aktørens rolle** | Brugeren er den der anvender systemet. Det er til ham at systemets GUI er designet.  Brugeren er den primære aktør i systemet. |
| **Alternativ reference** | Fængselsvagt |
| **Type:** | Primær |

|  |  |
| --- | --- |
| **Navn på aktør** | Indsat |
| **Aktørens rolle** | Den indsatte er den person som systemet er bygget op omkring. Han er grunden til at systemet tjener en funktion. |
| **Type:** | Sekundær |

## Funktionelle krav



Figur 5 - Use case diagram

| **Navn** | **UC1: Login** |
| --- | --- |
| Mål | At brugeren bliver logget på systemet og er klar til at betjene det. |
| Initiator | Bruger |
| Aktører og stakeholders | Primær aktør: Bruger |
| Antal samtidige hændelser | 1 |
| Forudsætning | At systemet er startet. |
| Resultat | Brugeren er logget på systemet |
| Hoved senarie | **1.** Brugeren åbner programmet.  **2.** Brugeren trykker på ”Login”.  **3.** Brugeren bliver bedt om at indtaste den korrekte kode på det tilhørende ”DE2 Board” og derefter trykke på godkend knappen på boardet.  **4.** Et ”godkendt” vindue vises på computerskærmen, og brugeren bliver bedt om at nulstille koden på DE2 boardet, og trykke på godkend knappen igen, for at fortsætte  **5.** Brugeren er nu logget på, og betjeningsmodulet bliver tilgængeligt. |

|  |  |
| --- | --- |
| **Navn** | **UC2: Log out** |
| Mål | At brugeren bliver logget af systemet så det ikke kan betjenes uautoriseret. |
| Initiator | Bruger |
| Aktører og stakeholders | Primær aktør: Bruger |
| Referencer | UC1: Login |
| Antal samtidige hændelser | 1 |
| Forudsætning | At brugeren er logget på systemet |
| Resultat | Brugeren er logget af og befinder sig på startskærmen. |
| Hoved senarie | **1.** Brugeren trykker på ”Log out”. **2.** Systemet låses (systemets UI viser startskærmen). |

|  |  |
| --- | --- |
| **Navn** | **UC3: Toggle Door** |
| Mål | At brugeren kan betjene (åbne eller lukke) en dør |
| Initiator | Bruger |
| Aktører og stakeholders | Primær aktør: Bruger |
| Referencer | UC1: Login |
| Antal samtidige hændelser | 1 |
| Forudsætning | Brugeren er logget på |
| Resultat | Døren åbner eller lukker |
| Hoved senarie | **1.** Bruger trykker på den ønskede dørs ”toggle” knap.  **2.** UI ændrer den viste status på døren:   * Ved åbnet dør: ”Closing” * Ved lukket dør: ”Opening”   **3.** Efter 4 sekunder sættes dørens status:   * Ved ”Closing” sættes forventet status til 0. * Ved ”Opening” sættes forventet status til 1. |

|  |  |
| --- | --- |
| **Navn** | **UC4: Open all doors** |
| Mål | At brugeren kan åbne alle døre |
| Initiator | Bruger |
| Aktører og stakeholders | Primær aktør: Bruger |
| Referencer | UC1: Login |
| Antal samtidige hændelser | 1 |
| Forudsætning | Brugeren er logget på |
| Resultat | Alle døre er åbnet |
| Hoved senarie | **1.** Bruger trykker på ”Open all doors”.  **2.** Alle lukkede døre skifter status til ”Opening”  **3.** Efter 4 sekunder sættes dørenes forventede status til 1. |

|  |  |
| --- | --- |
| **Navn** | **UC5: Close all doors** |
| Mål | At brugeren kan lukke alle døre |
| Initiator | Bruger |
| Aktører og stakeholders | Primær aktør: Bruger |
| Referencer | UC1: Login |
| Antal samtidige hændelser | 1 |
| Forudsætning | Brugeren er logget på |
| Resultat | Alle åbne dører lukker |
| Hoved senarie | **1.** Bruger trykker på ”Close all doors”.  **2.** Alle åbnede døre skifter status til ”Closing”  **3.** Efter 4 sekunder sættes dørenes forventede status til 0. |

| **Navn** | **UC6: Get Status** |
| --- | --- |
| Mål | At brugeren bliver orienteret ved uautoriseret adgangsforsøg |
| Initiator | Ingen (Loop, når inaktiv) |
| Aktører og stakeholders | Primær aktør: Bruger |
| Referencer | UC1: Login |
| Antal samtidige hændelser | 1 |
| Forudsætning | Brugeren er logget på, og systemet modtager eller sender ikke. |
| Resultat | UI har opdateret status tilsvarende til hver dør |
| Hoved senarie | **1**. ”Get Status” klassen på computeren, sender en 16 bit datapakke til Masteren, som indeholder kommandoen ”1111”.  **2.** Datapakken sendes via X.10 kommunikation til Slaven.  **3.** Masteren skifter mode fra sende til lytte.  **4.** Denne kommando fortæller Slaven, at den skal sende sin status tilbage.  **5.** Slaven sender en 16 bit datapakke til Masteren med sin status.  **6.** Masteren skifter mode fra lytte til sende.  **7.** Masteren sender status og adressen på døren via seriel RS232 til computeren.  **8.** UI sammenligner den forventede status med den aktuelle status.  [Undtagelse 1a]  **9.** Den aktuelle status stemmer overens med den forventede status  **10.** Status opdateres på UI til hhv. ”Closed” og ”Open”. |
| Undtagelse | **Undtagelse 1a:**  **1.** Den aktuelle status afviger fra den forventede status  **2.** Brugeren alarmeres med en besked på UI |

## Ikke funktionelle krav

MoSCoW:

M – Must have this

* Skal virke alle døgnets timer. Kun downtime ved skemalagt årlig vedligeholdelse.
* Skal kunne virke på et 18V AC system
* Skal kunne overføre med minimum 20 bit/s på det 18V el-net
* Skal kunne håndtere 2 enheder
* Skal kunne lukke/åbne en fængselsdør på under 4 sekunder.

S – Should have this if possible

* Bør ikke tage mere end 20 sekunder for computeren at tjekke status på alle dørene

C – Could have this if it does not affect anything

* Kunne håndtere 256 enheder
* Berørings interaktiv brugerflade (Touch screen)

W – Won’t have this time, but WOULD like in the future

* Fjernbetjening til kontrol af dørene
* Ikke gøre brug af STK500, men derimod selvbygget print.

## Begrænsninger

* Systemet anvender 18V AC i stedet for 230V AC, af sikkerhedsmæssige årsager.
* Døren er ikke en del af det færdige produkt, grundet tidsbegrænsning på projektet.
* Softwaren til systemet fungerer kun i Windows.

## Bruger interface

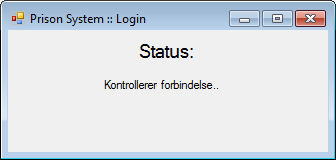
Brugeren bliver mødt af skærmen på figur 6, indtil der er etableret forbindelse med STK500. Derefter bliver der afventet den rigtige kode kombination på DE2 boardet (figur 9). Den orange markering skal den rette kombination indtastes, og derefter skal der trykkes på den blå knap.   
Efterfølgende vises en oversigt over alle døre tilknyttet systemet, hvor brugeren kan operer dørene (figur 8).



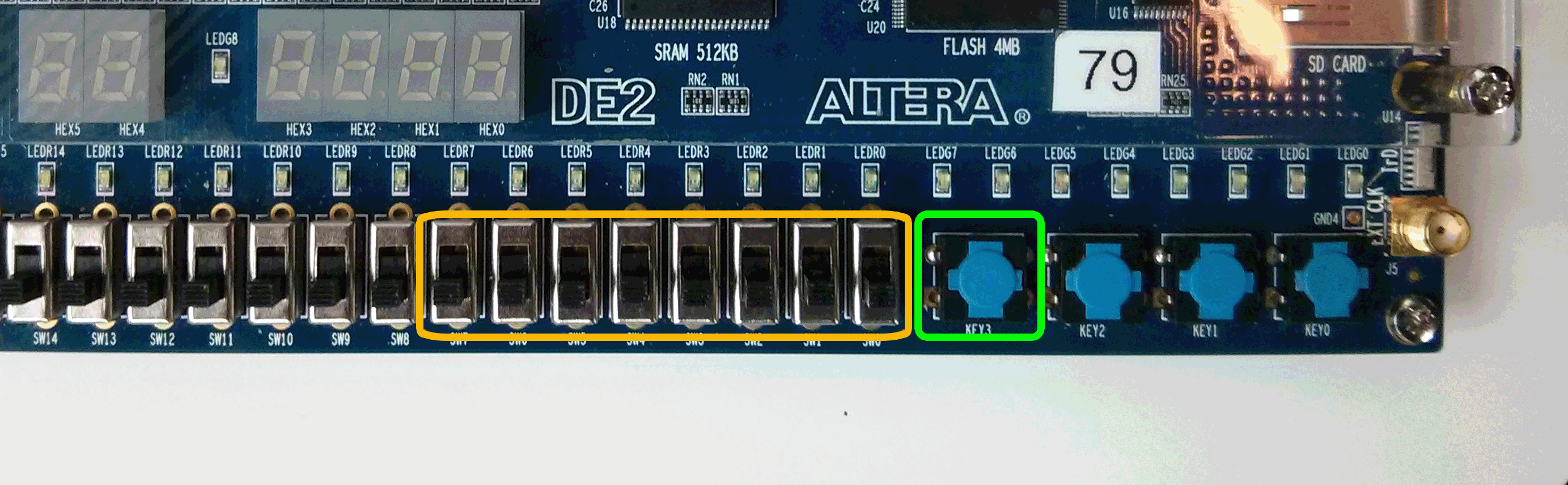
Figur 8



Figur 7



Figur 6



Figur 9