1. **Tegn et aktør – kontekst diagram for blokken Slusekontrol i Figur 3. Husk at aktører kan være fysiske genstande, som systemet interagerer med. Beskriv kort aktørerne til slusekontrollen.**

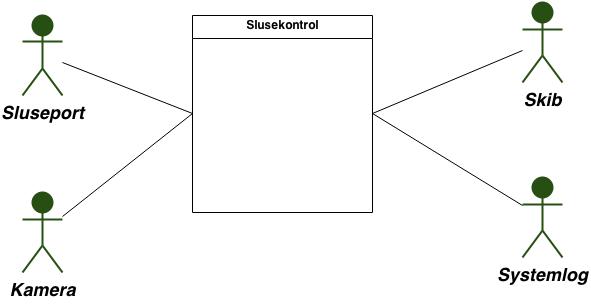


Figure - Aktør diagram fora Slusekontrol

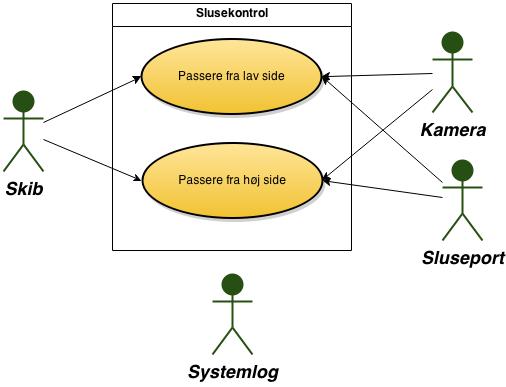
Sluseport 🡪 Modtager information fra slusekontrol om porten skal være åben eller lukket.

Kamera 🡪 Kontrollerer for skib på vandet og sender information videre til slusekontrolsystemet

Systemlog 🡪 Beskriver tidsangivet operationer og systemfejl

1. **En oplagt use cose for funktionaliteten af blokken Slusekontrol er den, hvor et skib passerer slusen fra den side, hvor vandstanden er lavest (kaldet lav side). Tegn et use case diagram med use casene:**

**”Passere fra lav side” og ”Passere fra høj side”.**



1. **Skriv den fuldstændigt (fully dressed) use case for ”Passere fra lav side”. Medtag bl.a detektering af skib ved ankomst til, tilstedeværelse i og afgang fra slusen, styring af sluseporte og –ventiler, skibets ind- og udsejling. Medtag undtagelsen, hvor der samtidig detekteres ankomst af skibe ved både lav og høj side. I dette tilfælde prioteres use casen ”Passere fra høj side” først.**

|  |
| --- |
| **Scenario for use case 1:** Passere fra lav side  **Navn:** Passere fra lav side  **Use case ID:** 1  **Samtlige forekomster:** 1  **Primær aktør:** Skib: Ønsker passage  **Sekundær aktør:** Kamera: Detektere skib  Sluseport: Modtagere information fra slusekontrol(åbner/lukker)  **Off Stage aktør:** Systemlog: Modtager information fra system, men påvirker dog ikke.  **Initialisering:** Skibet detekteres af kamera  **Frekvens:** Hver gang et skib detekteres startes processen.  **Forudsætninger:** Use case 1 gennemføres  **Resultat:** Skibet ender ved høj side  **Hovedforløb:**   1. Kamera detekterer skib   [Extension 1.a: Kamera kan ikke detektere skib]  [Extension 1.b: Kamera detekterer skib fra høj vand]   1. Sluseport 1 åbnes 2. Sluseport 1 lukkes 3. Ventil fra høj vand åbnes og derved fyldes sluse 4. Sluseport 2 åbnes 5. Skibet sejler videre ved høj vand   **Undtagelser:**  Extension:  1.a: Hovedforløbet startes forfra hvis skib ikke detekteres  1.b: Skib fra høj vand har forsejlsret. Så derfor gennemføres dens passage først. |

1. **Specificer 8 ikke-funktionelle krav for slusesystemet, med brug af FURPS+ (se slides omhandlende ”System specification”). Find selv på krav, der som minimum omhandler kategorierne: pålidelighed (reliability herunder MTBF), sikkerhed, performance og support. Proriter kravene med brug af MoSCoW metoden. Formuler krav med: Formuler krav med: skal (must), bør (should) og kunne (could)**

**Slusesystemet:**

|  |  |  |
| --- | --- | --- |
| Krav | FURPS+ | MoSCoW |
| Min. en process i timen | **P**erformance (Capacity & Efficiency Constraunts) | Skal (must) |
| Ni succesfulde processer før en fejl | **R**eliability (MTBF, Availabity) | Bør (should) |
| Let vedligeholdelse | **S**upportability (Installability, Compatibility, Maintainability) | Skal (must) |
| Effektiv åbning af sluser | **P**erformance (Response time, Start-up time) | Bør (should) |
| Sikkerhed – portene skal min. holde til det dobbelte vandtryk | **R**eliability | Skal (must) |
| Konstruktionens holdbarhed på min. 50 år | **R**eliability | Bør (should) |
| Forholdsvis stort – skal kunne håndtere store skibe/flere små | **U**sability (Accessibility) | Kunne (could) |
| Billigt at producere | **+** (Economic constraints) | Kunne (could) |

1. **Skriv en specifikation af to test cases til accepttesten, der hver især teste ét af følgende to scenarier i use casen specificeret i spørgsmål 2.**

**🡪 Hovedscenariet**

**🡪 Extension: Samtidig ankomst af skibe ved både lav og høj side**

**Brug nedenstående skabelon:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use case under test | |  | | | |
| Scenarie | |  | | | |
| Prækondition | |  | | | |
|  | | | | | |
| Step | Handling | | Forventet observation/resultat | Faktisk observation/resultat | Vurdering(OK/FAIL) |
| 1 |  | |  |  |  |
| 2 |  | |  |  |  |
| 3 |  | |  |  |  |