

AARHUS SCHOOL OF ENGINEERING

KRAVSPECIFIKATION

Indholdsfortegnelse

Indholdsfortegnelse	2
Kapitel 1 Kravspecifikation	5
1.1 Versionshistorik	5
1.2 Formål	5
1.3 Systembeskrivelse	5
1.4 Aktører	6
1.5 MoSCoW analyse	7
1.6 Ikke-funktionelle krav	8
1.7 Funktionelle krav	10
1.8 Domænemodel	14

Ordliste

Forkortelse	Forklaring
EPJ	Elektronisk Patient Journal
MTBF	Mean time between failure
MTTR	Mean time to restore
RPi	Raspberry Pi

Kravspecifikation

1

1.1 Versionshistorik

Version	Dato	Ansvarlig	Rettelser
1.0	6/9-16	KDK	Første version af kravspecifikation.
1.1	7/9-16	KDK	Tilføjet diagrammer, tabeller og skabeloner til use cases.
1.2	8/9-16	JB	Lavet første version af use case 1 og 2. Revideret Moscow ift. møde 7/9. Skrevet udkast til ikke-funktionelle krav.
1.3	12/9-16	KDK	Udkast til systembeskrivelse.
1.4	15/9-16	JB	Opdatering af systembeskrivelse.
1.5	19/9-16	JB+KDK	Opdatering af use cases, use case diagram og systembeskrivelse.
1.6	20/9-16	JB	Lavet use case 3.

Tabel 1.1: Versionsstyring af kravspecifikation.

1.2 Formål

En kravspecifikation er kontrakten mellem virksomhed og kunde. Heri er systemets funktionelle og ikke-funktionelle krav specificeret detaljeret.

Formålet med dette dokument er at specificere kravene for det antimikrobielle kateter.

1.3 Systembeskrivelse

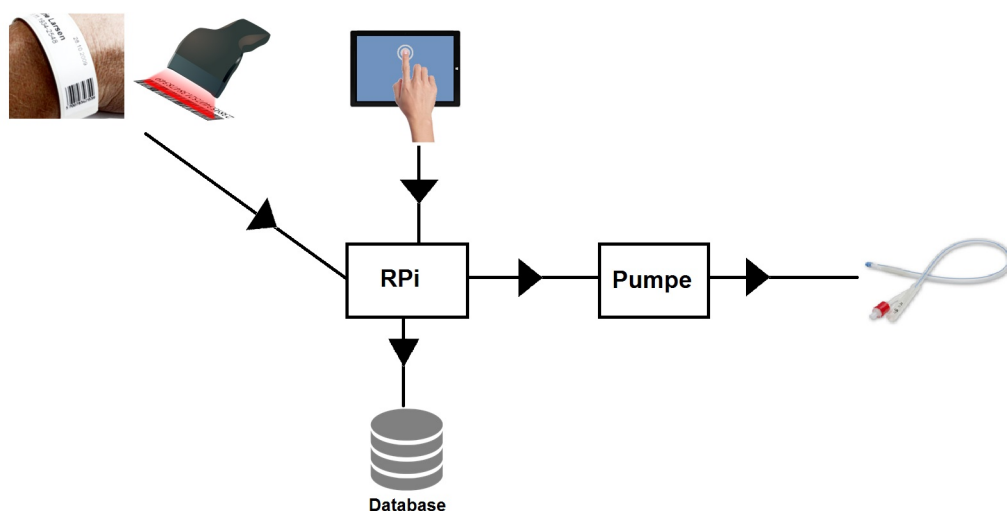
Det antimikrobielle kateter består af Raspberry Pi (RPi), display, strekodescanner, pumpe og selve katetret. RPi er en minicomputer, hvorpå programmet er installeret. Til RPi er tilkoblet en touchskærm med brugerinterface, som håndterer operatørens input. Data omkring hver skylning vil blive logget i en database internt på RPi.

Det er tiltænkt, at der er én pumpe til flere patienter, således at pumpen kan afkobles et kateter og dermed anvendes til skylning af flere patienter. Hermed kan operatøren foretage skylningen under stuegang, hvor alle patienter på afdelingen tilses. Med denne opsætning vil systemet kunne anvendes på en vilkårlig afdeling.

Systemet fungerer således, at operatøren kigger på systemets interface før stuegang. Her vises en dagsorden for hvilke patienter, der skal skylles den pågældende dag. Derefter

finder operatøren disse patienter og scanner deres tilhørende stregkode, hvori patientens unikke data findes. Dette skal foregå hver gang, der er stuegang. På denne måde kan systemet differentiere mellem de forskellige patienter med urinvejskateter. Ved scanning af en patient vil brugerinterfacet vise patientens CPR-nummer, tidspunkt for sidste skylning og informere omkring næste skylning.

Når operatøren har scannet stregkoden for en patient, hvor systemet anbefaler kateterskylning, vil en start-knap være mulig at klikke på. Hvis skylning af katetret ikke er påkrævet, er start-knappen utilgængelig. Ved tryk på start-knappen igangsættes pumpen, og katetret vil blive skyllet med klorhexidin og efterfølgende med isotonisk saltvand. Efter hver skylning logges tidspunktet for skylningen af den pågældende patient og en nedtælling til næste skylning startes. Når skylningen er afsluttet afmonteres pumpen og stuegangen fortsættes, så de næste patienter kan få en antimikrobiel skylning.



Figur 1.1: Systembeskrivelse

1.4 Aktører

Dette afsnit beskriver de aktører som påvirker systemet eller påvirkes af systemet. De er inddelt i primære og sekundære aktører, hvor de primære aktører påvirker systemet og sekundære aktører bliver påvirket af systemet.

1.4.1 Aktørbeskrivelse

Aktørnavn	Type	Beskrivelse
Operatør	Primær	Operatøren er den aktør, der ønsker at igangsætte en skylning. Operatøren er en sygeplejerske.

Tabel 1.2: Aktørbeskrivelse, Operatør

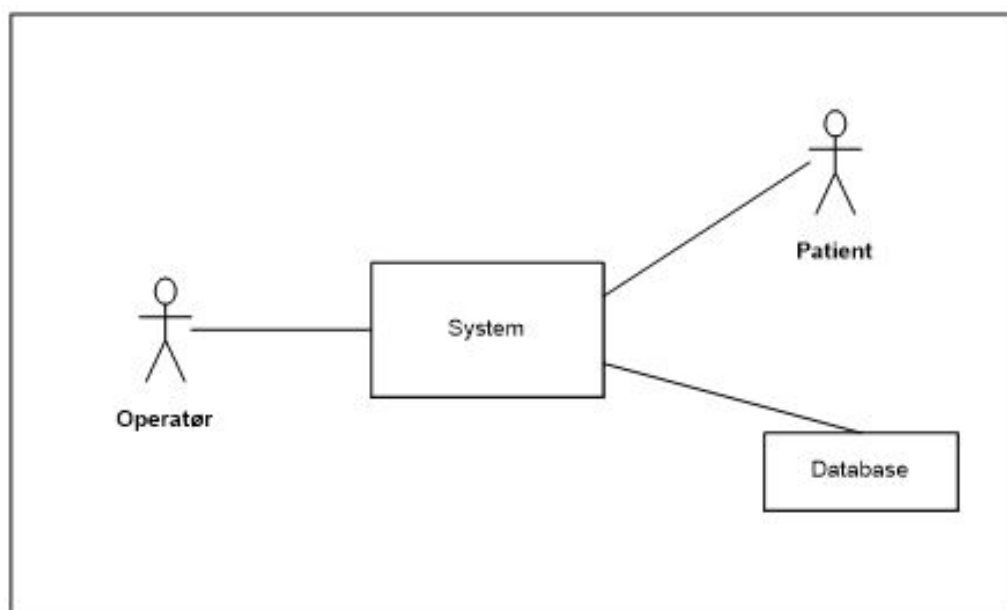
Aktørnavn	Type	Beskrivelse
Patient	Sekundær	Patienten er den aktør, hvorpå skylningen af katetret foregår.

Tabel 1.3: Aktørbeskrivelse, Patient

Aktørnavn	Type	Beskrivelse
Database	Sekundær	Databasen er der, hvor den loggede data gemmes.

Tabel 1.4: Aktørbeskrivelse, Database

1.4.2 Aktør-kontekst diagram



Figur 1.2: Aktør-kontekst diagram

1.5 MoSCoW analyse

MoSCoW-metoden betegner krav, som systemet skal opfylde (must), de krav som systemet bør realisere (should), de krav som systemet kunne realisere, men ikke har indvirkning på de andre krav (could), og de krav som ikke implementeres, men er mulige fremtidige opdateringer og udvidelser (would like, but won't get).

1.5.1 Must:

- Systemet skal have en manuel funktion, som starter skylleprogrammet, når operatøren ønsker det.
- Systemet skal, ved hjælp af en timer, vide hvornår patienterne skal skylles næste gang.

- Systemet skal logge patientdata i en database efter hver skylning med information om: patient, skylletidspunkt, skyllelængde og status.
- Systemet skal differentiere patienterne ved hjælp af stregkode.
- Ved scanning af stregkoder fortæller systemet via. brugerinterface, hvornår patienten sidst er blevet skyllet og hvornår næste skylning skal foregå.
- Systemet skal have et touch-interface, så operatøren kan styre systemet.
- Systemet skal styres af RPi 2 model B.
- Programmet skal programmeres i C# fra Windows.

1.5.2 Should:

- Systemet skal indeholde grænseværdier for, hvor ofte systemet kan køre for samme patient.
- Systemet skal notificere operatøren, hvis en patient ikke er blevet skyllet på en påkrævet skylningsdag.

1.5.3 Could:

- Systemet skal være i en beskyttende boks, så hardwaren ikke kan tilgås direkte.
- Systemet skal give fejlmelding ved ukorrekt skylning.
- Systemet skal afbryde skylningen ved fejl.

1.5.4 Would like, but won't get:

- Der skal implementeres en database, så devices fra forskellige afdelinger kan tilgå samme informationer om skylldata.
- Den loggede data skal gemmes i EPJ (Elektronisk Patient Journal).
- Systemet skal kunne køre på genopladeligt batteri, så den er transportabel.

1.6 Ikke-funktionelle krav

Til at beskrive de ikke-funktionelle krav gøres der brug af FURPS+. FURPS+ står for:

- F: Funktionelle krav
- U: Usability krav
- R: Reliability krav
- P: Performance krav
- S: Supportability krav
- +: Ekstra krav til systemet, som ikke hører under ovenstående

Dog er de funktionelle krav (F) i FURPS+ udeladt, og i stedet er der benyttet use cases, jf. afsnit 1.7.

1.6.1 Usability

- **U1:** Operatøren skal kunne afbryde skylning ved nødvendighed.
- **U2:** En førstegangsbruger skal ikke bruge mere end 1 time på at lære systemet.
- **U3:** Operatøren skal kunne indskrive afdelingens mulige operatørers navne i systemet på under 15 minutter.
- **U4:** Operatøren skal kunne følge med i, hvor lang tid, der er tilbage af den igangværende skylningen på brugerinterfacet.
- **U5:** Operatøren skal kunne se, hvor mange patienter, der skal skylles på dags dato.

1.6.2 Reliability

- **R1:** Systemet skal ikke kunne køre hvis et kateter ikke er tilsluttet.
- **R2:** Systemet skal have en maksimal MTTR på 10 minutter.
- **R3:** Systemet skal have en MTBF på 100 dage.
- **R4:** Systemet skal have en availability på:

$$\frac{100\text{dage}}{100\text{dage} + 10\text{min}} = 99,993\% \quad (1.1)$$

1.6.3 Performance

- **P1:** Operatøren skal kunne starte skylleprogram på under 3 minutter efter korrekt opsætning af udstyr.
- **P2:** Softwareprogrammet skal kunne starte på under 1 minut.
- **P3:** Svartid skal være max 1 sekund ved tryk på en vilkårlig knap.
- **P4:** Data skal være logget i databasen maksimalt 10 sekunder efter endt skylning.

1.6.4 Supportability

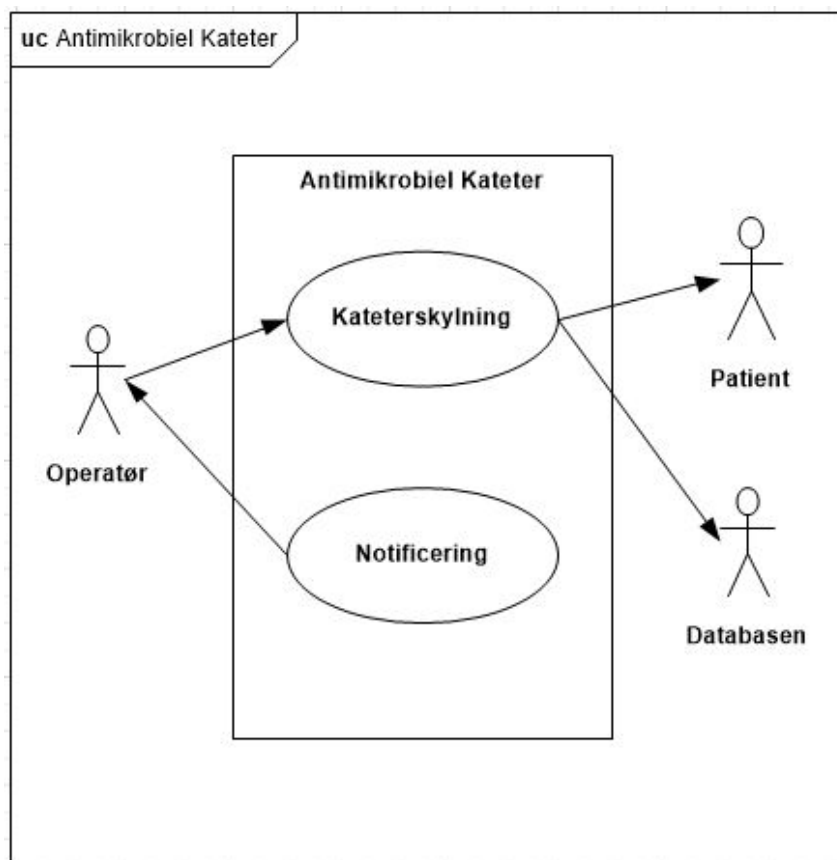
- **S1:** Programmet skal udvikles efter 3-lagsmodellen.
- **S2:** Programmet skal kunne overføres til RPi via. USB-stik.
- **S3:** Operatøren skal kunne tilgå komponenter i RPi og i pumpen.

1.6.5 + Ekstra

- **E1:** RPi skal have et output på 5V til pumpen
- **E2:** Touchskærmen, der er koblet til RPi, skal have en størrelse på minimum 5"og maximum 10".

1.7 Funktionelle krav

1.7.1 Use case diagram



Figur 1.3: Use case diagram

1.7.2 Fully-dressed use cases

Use case 1:

Navn	Opsætning
Use case ID	UC1
Samtidige forløb	1
Primær aktør	Operatør.
Initiering	Operatør trykker "Opsætning".
Frekvens	Første gang systemet anvendes på en afdeling.
Forudsætninger	Systemet er opstartet og operatør kender afdelingens andre mulige operatører.

Resultat	Afdelingens operatører er tilføjet.
Hovedforløb	<ol style="list-style-type: none"> 1. Operatør trykker "Opsætning". 2. Interface åbner nyt vindue med mulighed for tilføjelse af nye operatører. 3. Operatøren trykker "Tilføj nye operatører". [Extension 3a. <i>Operatør vil fjerne unødvendige operatører.</i>] 4. Operatøren indtaster navne og trykker "OK". 5. Liste over afdelingens operatører vises. 6. Operatøren trykker "Godkend". 7. Listen af operatører opdateres og gemmes på systemet. 8. Startskærmen vises og systemet er klar brug.
Extension	<p>3a. [<i>Operatør vil fjerne unødvendige operatører.</i>]</p> <ol style="list-style-type: none"> 1. Operatøren trykker "Fjern operatør". 2. Operatøren klikker på checkboxen ud fra de navne, der skal fjernes. 3. Operatøren klikker knappen "Fjern". 4. Liste over afdelingens operatører vises med opdaterede operatøre. 5. Operatøren trykker "Godkend". 6. Listen af operatører opdateres og gemmes på systemet. 7. Startskærmen vises og systemet er klar brug.

Tabel 1.5: Use case 1, Opsætning

Use case 2:

Navn Kateterskyllning

Use case ID	UC2
Samtidige forløb	1
Primær aktør	Operatør.
Initiering	Operatøren trykker på "Start skylleprogram".
Frekvens	Hvert kateter skal gennemskyldes hver 3. dag.
Forudsætninger	Kateteret er tilkoblet pumpen korrekt og programmet kører.
Resultat	Kateteret er skyllet igennem, data er logget i database og pumpen er afkoblet katetret.
Hovedforløb	<ol style="list-style-type: none"> 1. Operatøren ser ud fra dagsordnen på interfacet, hvilke patienter, der skal have kateterskyldning. 2. Operatøren finder de patienter, der skal have kateterskyldning. 3. Operatøren scanner stregkode. 4. Displayet viser et interface med patientens navn, CPR-nummer, log over sidste skyldninger og hvornår næste skyldning skal foregå. [Extension 4a. <i>Der skal ikke skyldes på den pågældende dag.</i>] 5. Operatøren tilslutter pumpe på katetret. 6. Operatøren trykker på knappen "Start-skyldprogram". 7. Messageboks fremtoner med teksten: "Tjek at kateteret er tilkoblet". 8. Operatøren trykker på "OK" på messageboksen. 9. Besked vises med "Skyldning er igang. Vent venligst" 10. Interfacet viser en nedtælling til, hvornår skyldning er færdig. 11. "Afslut-knappen fader ud og kan ikke trykkes på, mens skyldprogrammet foregår. [Extension 11a. <i>Operatør trykker "Afbryd skyldprogram".</i>] 12. Messageboks med teksten: "Skyldprogram er færdigt" vises.

13. Knappen "Afslut" fremstår nu normalt og kan trykkes på. "Afbryd-knappen" fades ud.
14. Operatøren trykker på "Afslut-knappen".
15. Messageboks med teksten: "Afkobl kateter" vises.
16. Operatøren afkobler katetret og trykker på "OK".
17. Startskærmen vises og systemet er klar til ny kateterskylning.

Extensions

- 4a. [*Der skal ikke skylles på den pågældende dag.*]
 1. Messageboks vises med teksten: "Katetret skal ikke skylles i dag".
 2. "Start skylleprogram" kan ikke trykkes, og dermed kan der ikke skylles.
 3. Operatøren afslutter ved at trykke på krydset i hjørnet.
 4. Systemet er klar til at scanne en ny patient.
- 11a. [*Operatør trykker "Afbryd skylleprogram".*]
 1. Skylleprogrammet afbrydes.
 2. Data om afbrydelsen registreres i databasen.
 3. Startskærmen vises og systemet er klar til ny kateterskylning.

*Tabel 1.6: Use case 2, Kateterskylning***Use case 3:**

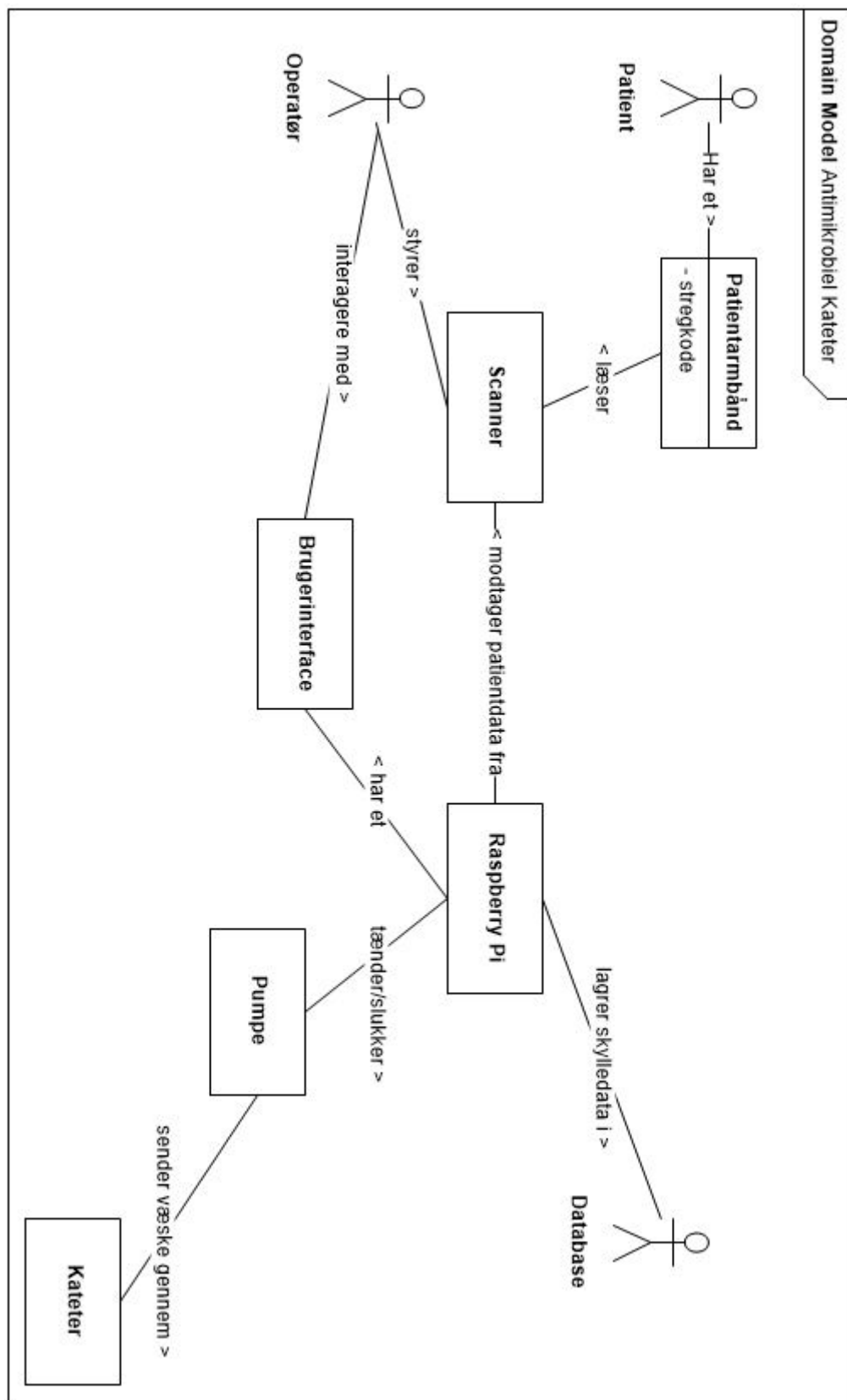
Navn	Notificering
Use case ID	UC3
Samtidige forløb	1
Primær aktør	Operatør.

Initiering	System har registreret, at intervallet for planlagt skylning er overskredet.
Frekvens	Når en kateterskylning er glemt.
Forudsætninger	Der er foretaget skylninger og registreret patienter i databasen.
Resultat	Personalet er blevet informeret om forglemmelse af kateterskylning.
Hovedforløb	<ol style="list-style-type: none"> 1. Messageboks viser: "Patient xxx mangler kateterskylning". 2. Operatør finder patienten. 3. Operatøren trykker "OK". 4. Use case 2 køres. [Extension 4a. <i>Patienten er udskrevet.</i>]
Extension	<p>4a. [<i>Patienten er udskrevet.</i>]</p> <ol style="list-style-type: none"> 1. Operatøren vælger "Patient er udskrevet". 2. Databasen opdateres ved at slette patientens data om skylninger. 3. Startskærmen vises og systemet er klar til ny kateterskylning.

Tabel 1.7: Use case 3, Notificering

1.8 Domænemodel

Domænemodellen giver et overblik over systemets elementer, samt elementernes associationer. Blokkene er konceptionelle klasser som er essentiel for netop dette system. Modellen giver dermed et ikke-teknisk overblik over elementerne i det samlede system.



Figur 1.4: Domænemodel