



Bacheloroppgave

IBE600

Sporing av medisinsk utstyr med RFID

Johan Sæther

Totalt antall sider inkludert forside: 32

Molde, 31.5.2022



Obligatorisk egenerklæring/gruppeerklæring

Den enkelte student er selv ansvarlig for å sette seg inn i hva som er lovlige hjelpemidler, retningslinjer for bruk av disse og regler om kildebruk. Erklæringen skal bevisstgjøre studentene på deres ansvar og hvilke konsekvenser fusk kan medføre. Manglende erklæring fritar ikke studentene fra sitt ansvar.

Du/dere fyller ut erklæringen ved å klikke i ruten til høyre for den enkelte del 1-6:		
1.	Jeg/vi erklærer herved at min/vår besvarelse er mitt/vårt eget arbeid, og at jeg/vi ikke har brukt andre kilder eller har mottatt annen hjelp enn det som er nevnt i besvarelsen.	<input checked="" type="checkbox"/>
2.	Jeg/vi erklærer videre at denne besvarelsen: <ul style="list-style-type: none">• ikke har vært brukt til annen eksamen ved annen avdeling/universitet/høgskole innenlands eller utenlands.• ikke refererer til andres arbeid uten at det er oppgitt.• ikke refererer til eget tidligere arbeid uten at det er oppgitt.• har alle referansene oppgitt i litteraturlisten.• ikke er en kopi, duplikat eller avskrift av andres arbeid eller besvarelse.	<input checked="" type="checkbox"/>
3.	Jeg/vi er kjent med at brudd på ovennevnte er å <u>betrakte som fusk</u> og kan medføre annullering av eksamen og utestengelse fra universiteter og høgskoler i Norge, jf. Universitets- og høgskoleloven §§4-7 og 4-8 og Forskrift om eksamen §§16 og 36.	<input checked="" type="checkbox"/>
4.	Jeg/vi er kjent med at alle innleverte oppgaver kan bli plagiatkontrollert, jf. høgskolens regler og konsekvenser for fusk og plagiat	<input checked="" type="checkbox"/>
5.	Jeg/vi er kjent med at høgskolen vil behandle alle saker hvor det forligger mistanke om fusk etter høgskolens retningslinjer for behandling av saker om fusk	<input checked="" type="checkbox"/>
6.	Jeg/vi har satt oss inn i regler og retningslinjer i bruk av kilder og referanser på biblioteket sine nettsider	<input checked="" type="checkbox"/>

Personvern

Personopplysningsloven

Forskningsprosjekt som innebærer behandling av personopplysninger iht.

Personopplysningsloven skal meldes til Norsk senter for forskningsdata, NSD, for vurdering.

Har oppgaven vært vurdert av NSD?

ja nei

- Hvis ja:

Referansenummer:

- Hvis nei:

Jeg/vi erklærer at oppgaven ikke omfattes av Personopplysningsloven:

Helseforskningsloven

Dersom prosjektet faller inn under Helseforskningsloven, skal det også søkes om forhåndsgodkjenning fra Regionale komiteer for medisinsk og helsefaglig forskningsetikk, REK, i din region.

Har oppgaven vært til behandling hos REK?

ja nei

- Hvis ja:

Referansenummer:

Publiseringsavtale

Studiepoeng: 15

Veileder: Bjørn Jæger

Fullmakt til elektronisk publisering av oppgaven

Forfatter(ne) har opphavsrett til oppgaven. Det betyr blant annet enerett til å gjøre verket tilgjengelig for allmennheten (Åndsverkloven. §2).

Alle oppgaver som fyller kriteriene vil bli registrert og publisert i Brage HiM med forfatter(ne)s godkjenning.

Oppgaver som er unntatt offentlighet eller båndlagt vil ikke bli publisert.

Jeg/vi gir herved Høgskolen i Molde en vederlagsfri rett til å gjøre oppgaven tilgjengelig for elektronisk publisering:

ja nei

Er oppgaven båndlagt (konfidensiell)?

ja nei

(Båndleggingsavtale må fylles ut)

- Hvis ja:

Kan oppgaven publiseres når båndleggingsperioden er over?

ja nei

Dato:

Samandrag

I denne oppgåva har eg undersøkt korleis ein kan setje opp og gjennomføre effektiv sporing av medisinsk utstyr ved hjelp av RFID-teknologi og GS1-standardar i samanheng med prosjektet «Entydig strekkoding» ved Helse Sør-Øst. Eg har gått igjennom noko teori om RFID og GS1-standardar, og diskutert ulike bruksområde for RFID ved Helse Sør-Øst. På bakgrunn av dette så har eg laga eit case for registrering av RFID-hendingar og sett opp eit IT-miljø for å simulere dette. Dette har eg brukt til å lese av rå RFID-hendingar og registrert dei som EPCIS forretningshendingar, som eg etterpå har kunne utført spørjingar til datalageret om. Eg har diskutert resultatet av spørjingane og feil og manglar som har oppstått. Til slutt har eg presentert nokre forslag til forbetringar og vidare arbeid.

Innhald

1.0	Introduksjon	1
1.1	Bakgrunn	1
1.2	Problemstilling	1
1.3	Prosjektering.....	2
2.0	Teori	2
2.1	RFID.....	3
2.1.1	RFID vs. strekkoding	4
2.1.2	Moglege bruksområde for RFID ved Helse Sør-Øst	11
2.2	GS1	4
2.3	EPCIS	4
2.3.1	Event data og master data	5
2.3.2	Core Business Vocabulary	5
2.3.3	What	5
2.3.4	When	6
2.3.5	Where	6
2.3.6	Why.....	7
2.3.7	Retting av hendingar	7
2.3.8	EPCIS repository.....	9
2.3.9	Capture client	9
2.3.10	Query client.....	11
3.0	Analyse og design	11
4.0	Implementering og testing	14
4.1	Oppsett av IT-miljø	14
4.1.1	Programvare	14
4.1.2	Maskinvare.....	15
4.2	Lesing av RFID-events.....	16
4.2.1	Bruk av RFID-lesaren	16
4.2.2	When	16

4.2.3	What	16
4.2.4	Where	17
4.2.5	Why	17
4.2.6	Capture data	17
4.3	Resultat	18
5.0	Konklusjonar	20
5.1	Manglar og forslag til forbetringar	21
5.2	Vidare arbeid	21
6.0	Referanser	21

1.0 Introduksjon

1.1 Bakgrunn

Helse Sør-Øst er det største regionale helseforetaket i Norge med 31 000 tilsette og med ansvar for spesialisthelsetenestene til 3,1 millioner mennesker. Foretaket eier 11 helseforetak, mellom annet Noregs største, Oslo Universitetssykehus. (Helse Sør-Øst 2022) Foretaket eier store mengder utstyr og helsepersonellet har behov for å kunne finne riktig utstyr raskt. Helse Sør-Øst har vedtatt at GS1 og ISBT128 skal etableres som regionale standarder for automatisk identifikasjon og datafangst. På bakgrunn av dette startet de i 2018 prosjektet «Entydig strekkoding (GS1)». (Helse Sør-Øst 2022) Dette prosjektet skal legge grunnlaget for tre gevinstområder: sporing og tilbakekalling av implantat, lukkede lækjemiddelsløyfer og sporing av medisinsk teknisk utstyr. Dette prosjektet skal være ferdig i løpet av 2022.

Eg fikk vite at Bjørn Jæger hadde en potensiell bacheloroppgåve i sammenheng med Entydig Strekkoding like før jul i 2021. Med bakgrunn i erfaringene fra faget IBE405 Cyber-Physical Systems and Smart Cities, der jeg også arbeidet med et IoT-prosjekt, syntest jeg dette kunne være interessant. Eg meldte derfor straks min interesse. I denne bacheloroppgåva skal jeg bruke standard programvare og GS1-standarder for å simulere sporing av medisinsk teknisk utstyr.

1.2 Problemstilling

Sidan hensikten med prosjektet Entydig Strekkoding er å kunne holde oversikt over hvor det medisinske utstyret til enhver tid, blir problemstillinga mi:

«Korleis sette opp og gjennomføre effektiv sporing av medisinsk utstyr ved hjelp av RFID-teknologi og GS1-standarder.»

For å løyse denne problemstillinga vil eg lage eit proof-of-concept for sporing av gjenstandar. Til dette skal eg bruke eit EPCIS repository og RFID-utstyr. Eg skal sette opp ein case for sporing ved hjelp av IT-miljøet eg har sett opp og få meningsfylt informasjon ut av dette.

1.3 Prosjektering

Eg har delt opp prosjektet i 4 fasar:

1. Sette opp IT-miljø
 - a. Sette opp maskinvare og installere programvare og få desse til å virke saman
2. Lese av data
 - a. Lese av tags ved hjelp av lesaren
3. Analysere data
 - a. Gjere om rå RFID-hendingar til forretningshendingar
 - b. Utføre spørjingar til datalageret om hendingane
4. Skrive sluttrapport

Eg sette av om lag ein månad til kvar fase.

2.0 Teori

GS1 er eit sett med standardar for effektiv vare- og informasjonsflyt. EPC (Electronic Product Code) er det standardiserte elektroniske formatet for unike identifikatorar knytta til fysiske produkt eller gjenstandar. (GS1 2022) EPC brukast særleg til identifikasjon ved hjelp av RFID-tags. Ein EPC kan fortelje oss nøkkeldata om produktet, som produksjonsdata, opphav og partinummer. I samsvar med GS1 så skal alle EPCar følgje eit skjema. For «trade items» er dette SGTIN, eller serialized GTIN, som består av eit global trade item number og eit serienummer. Serienummeret er då det som skiljer produktet frå alle likedane produkt.

Dei aktuelle GS1-standardane for Helse Sør-Øst er:

Global Trade Item Number (GTIN) for identifikasjon av legemiddel

Unique Device Identification (UDI) utstedt av GS1 for identifikasjon av medisinsk utstyr

Global Location Number (GLN) for identifikasjon av lokasjonar

Global Service Relationship Number (GSRN) for identifikasjon av pasientar og helsepersonell (GS1 2022)

I tillegg er det tilrådd å bruke ein GS1-standard for datafangst, og RFID er ein av desse. (GS1 2022)

2.1 RFID

RFID (Radio Frequency IDentification) er ein metode for automatisk identifisering og datafangst. Som så mange andre teknologiar er RFID som vi kjenner det i dag eit resultat av ei lang utvikling med mange involverte. Ein tidleg teknologi relatert til RFID er «identification, friend or foe» (IFF) for kampfly. Dette er eit transpondersystem med lang rekkevidde som brukast til å identifisere fly. (Sepaco Tech 2022) Det første patentet som innehaldt ordet «RFID» vart registrert i 1973 av Charles Walton. (Paragon ID 2022) Sidan den gong har RFID-teknologien hatt ei eksplosiv utvikling og vorte teke i bruk på stadig nye område. (Sepaco Tech 2022)

RFID blir i dag brukt på mange område, som lagerstyring, adgangskontroll, betaling og identifisering av dyr. Eit RFID-system består av ein radiosendar/radiomottakar eller transponder som kallast for ein RFID-brikke. Denne brikka består av ein mikrochip, ei antenne og og eit underlag. Når transponderen får eit radiosignal, så sender den automatisk ein kode tilbake. Ei RFID-brikke kan vere passiv eller aktiv. Passive brikker får overført energi frå radiosignala den mottok og treng difor ikkje batteri for å sende radiosignal. Rekkevidde på desse kan vere frå 1 cm til 12 meter. Aktive brikker brukar batteri og kan difor sende sterkare radiobølgar med lengre rekkevidde. Desse kan ha ei rekkevidde frå 1 cm til fleire 100 m. I tillegg kan dei innehalde sensorar for temperatur, lys, stråling osv. Lesarar kan også vere aktive eller passive, og vi har dermed 3 typar RFID-system:

Active Reader Passive Tag (ARPT)

Passive Reader Active Tag (PRAT)

Active Reader Active Tag (ARAT)

RFID-system kan bruke frekvensar i LF, HF, UHF og mikrobølgebandet, der det siste krev aktive tags. (Phillips 2017)

2.1.1 RFID vs. strekkoding

Ein mykje brukt teknologi som også blir brukt til identifisering av objekt er strekkodar. Strekkodar blir lest av ved hjelp av laserrefleksjonar. (Lowry Solutions 2022) Strekkoder er som regel trykt på objektet eller ein etikett, og prisen tilsvarar dermed prisen på trykksverta og eventuelt etiketten. Til samanlikning kan ein RFID-brikke koste frå ei krone til eit par hundre kroner. Dette kan i mange tilfelle tilsvare ein betydeleg del av verdien på objekta som skal merkast. Dersom verdien til dømes er 50 kroner, og så vil ein billeg RFID-brikke på 2 kroner tilsvare 4% av verdien på sjølv objektet som skal sporast. Fordelen med RFID framfor strekkoding er at vi kan lese av RFID-brikker utan å ha dei i synslinja. Det betyr at vi kan lese dei av likevel om dei er bak andre objekt eller er innpakka, og at vi kan lese dei av på avstand og fleire om gongen. Dette gjer det mogleg å gjere unna mange registreringar på kort tid. Ein annan fordel med RFID-brikker er at dei kan lagre unike data som utløpsdato og produksjonsdato.

På grunn av dei ulike eigenskapane til trykte strekkoder og RFID-brikkene, så er spørsmålet om kva ein skal bruke til å identifisere objekt noko einkvar verksemd bør vurdere i forhold til sine egne behov og forretningsprosessar.

2.2 GS1

GS1 er ein organisasjon som vedlikeheld standardane som er kjent under same namn. GS1 vart grunnlagt i 1974 for å administrere strekkodestandarden UPC og har i dag medlemsorganisasjonar i over 110 land. Blant standarane utvikla av GS1 er EPC, eller Electronic Product Code, som er ein syntaks for unike identifikatorar for unike objekt, lokasjonar eller andre entitetar. EPC har fleire ulike representasjonar, og ein av desse er binær representasjon som passar til bruk på RFID-tags. (GS1 2017)

2.3 EPCIS

EPCIS er ein standard utvikla av GS1 som gjer det mogleg for handelspartnerar å dele informasjon om dei fysiske rørsleane og statusen til produkt mens dei forflyttar seg

gjennom verdikjeda. Målet er at forskjellige applikasjonar skal kunne dele denne informasjonen mellom seg, både mellom organisasjonar og internt. (GS1 u.d.) Sentralt i EPCIS er dei fire spørsmåla når, kva, kvar og kvifor (when, where, what and why). EPCIS definerer ein datamodell for å representere informasjon om fysiske objekt. Hovudbestanddelane i datamodellen er EPC, hendingstid, forretningssteg, disposisjon, lesepunkt, forretningslokasjon og forretningstransaksjon.

2.3.1 Event data og master data

EPCIS omhandlar to forskjellige typar data: event data og master data. Event data er data som oppstår under forretningsprosessane, og master data er mellom anna vokabulara som gjer oss konteksten for å tolke event data. (GS1 2016)

2.3.2 Core Business Vocabulary

EPCIS er meint å brukast saman med GS1 Core Business Vocabulary (CBV). Denne standarden definerer verdiane som vi brukar til å fylle datastrukturane med hendingsdata. Core Business Vocabulary definerer vokabulærelement for Business Step, Disposition, Business Transaction Type og Source/Destination type. Dette gjer dataa våre forståelege på tvers av foretak. CBV gjer oss også malar for å konstruere vokabulærelement for Object, ObjectClass, Location, Business Transaction ID, Source/Destination ID og Transformation ID. (GS1 2017)

2.3.3 What

What-dimensjonen identifiserer objekta som hendinga gjeld. What-dimensjonen kan innehalde ein eller fleire identifikatorar for fysiske eller digitale objekt, eller klassar av fysiske eller digitale objekt. Denne bør vere på forma EPC URI, som er ei streng på forma *urn:epc:id:scheme:component1.component2....* Dei tre første elementa, *urn:epc:id:*, er alltid dei same. Det neste elementet, *scheme:*, indikerer kva slags objekt vi har med å gjere. For logistikkeiningar som ei eske eller pall med innhald brukar vi *scca:*. For enkeltstående objekt brukar vi *sgtin:*. SGTIN står for serialised global trade item number og består av global trade item number (GTIN) etterfulgt av eit serienummer. GTIN indikerer eit

produkt, mens serienummeret indikerer ein forekomst (instance) av produktet. Dersom GTIN er det unike nummeret for ein sengemodell, så er SGTIN er eit unikt nummer for akkurat *denne* senga. GTIN kan ha 8, 12, 13 eller 14 siffer, men det vanlegaste er 13 siffer, eller GTIN-13. (GS1 2022) Døme på GTIN-nummer er 07012345670001 og 07012345670018, der dei det 2. og 3. sifferet er landkode, dei sju påfølgande er firmakode, dei tre neste er artikkelkode og det siste er kontrollsiffer. Vi lagrar dette utan kontrollsifferet. Etter GTIN-nummeret kan ein legge til eit serienummeret for å få ein SGTIN. Ein kan gå til <https://www.gs1.org/services/epc-encoderdecoder> for å konvertere ein SGTIN til EPC URI. Med landkode 70, firmakode 1234567, artikkelkode 001 og serienummer 100 så får vi URI-en *urn:epc:id:sgtin:708012341.0001.100*. (GS1 2017)

2.3.4 When

When-dimensjonen skal innehalde dato og tid, samt tidssone, for hendinga på ISO-6801-formatet. I samsvar med ISO-6801 så er skrivemåten for data YYYY-MM-DD og tid hh:mm:ss. Dersom ein skal uttrykke både dato og klokkeslett set ein ein T framfor klokkeslettet, slik at skrivemåten blir YYYY-MM-DDThh:mm:ss. Dersom klokkeslettet er i tidssona UTC set ein ein Z til slutt, mens andre tidssonar uttrykkast ved ±hh. Så for norsk sommartid set ein +02 til slutt. (GS1 2017)

2.3.5 Where

Where-dimensjonen kan innehalde data om lokasjonen der hendinga skjedde, og lokasjonen objektet befinn seg etter hendinga. Where-dimensjonen består av ReadPoint og BusinessLocation, der begge er valfrie. Where-dimensjonen omfattar to dataelement: ReadPoint og BusinessLocation. (GS1 2017) ReadPoint er lokasjonen der avlesninga skjedde, og BusinessLocation er gjenstanden sin lokasjon etter avlesninga. For å identifisere ein lokasjon kan vi gje den eit Global Location Number (GLN), eller vi kan bruke koordinatar. (GS1 2017)

Ein kan tenkje på ReadPoint som ei dør og BusinessLocation som eit rom. Etter at gjenstanden er lese av i døra, så blir den flytta til rommet bak. ReadPoint er då staden der gjenstanden sist er sett, mens BusinessLocation er kvar gjenstanden er akkurat no. (GS1 2017) I vårt tilfelle så er det det siste som vi er ute etter, så det gjer meining å registrere

dette. Dersom vi tar ein gjenstand ut av eit rom, så hamnar den ikkje på destinasjonen med ein gong. Vi legg den difor inn på den lokasjonen som er eitt nivå mindre spesifikt, nemleg sjukehuset. Først når vi tek den inn i rommet den skal til, så registrerer vi dette rommet som lokasjon. Dersom gjenstanden derimot skal ut av sjukehuset så kan det derimot hende at lokasjonen blir ukjent fram til den kjem fram til destinasjonen. Eit alternativ er å registrere den på ein bil, men dette blir utanfor omfanget til denne oppgåva.

Ein mogleg forenkling er å kun registrere objekt når dei kjem inn i rommet for første gong, og ikkje når ein tek dei ut. Ein treng då berre å fylle ut eitt av felta ReadPoint eller BusinessLocation, sidan staden objektet sist er sett og staden den hamna etter avlesninga er den same. Det vil då vere sjølvforklarande kva som hender med objektet, og enklare å automatisere registreringa av forflytningane. Så lenge objektet blir flytta frå rom til rom innan få minutt, vil dette sannsynlegvis ikkje føre til større problem. Problema oppstår først dersom objektet av ein eller annan grunn blir forlagt utan ein ny registrering. Det kan då bli vanskeleg å finne det att, sidan vi ikkje har spor etter tidspunktet då dei blei teke ut av lageret/rommet. Difor bør alle objekt bli registrert både på veg inn og ut av lokasjon. Dersom vi skal lage eit program som automatiserer jobben med å registrere om objektet gjekk inn eller ut, kan det først spørje datalageret om objektet finst på rommet frå før. Dersom det gjer det, så er objektet på veg ut av rommet, og viss ikkje så er det på veg inn.

2.3.6 Why

Why-dimensjonen inneheld informasjon om forretningskonteksten. Denne dimensjonen inneheld fleire valfrie felt. Dei to første er BusinessStep og Disposition, som indikerer forretningssteget som er utført og statusen til produktet etterpå. Deretter har vi mellom anna BusinessTransactionID, som kan lenkje hendinga til ein spesifikk forretningstransaksjon.

2.3.7 Retting av hendingar

Av og til kan det hende at registrerte data ikkje reflekterer kva som skjedde i den fysiske verda. Korkje capture interface eller query interface kan slette eller endre på ein tidlegare registrert hending. Måten ein løyser dette på er å sjå på oppdaginga og rettinga av ein tidlegare feil som ein eiga hending. Eit døme på ein feil er at ein har utelatt ein vare i ei

hending som omfattar sending av fleire varer. Løysinga på dette er å lage ei ny hending for den varen som tidlegare var utelatt. I det motsette tilfellet, dersom ein har lagt til ei vare for mykje i hendinga, så kan vi bruke *void_shipping* i feltet business step, som er laga for akkurat dette formålet.

Nokre typar hendingar kan ikkje rettast på ved å lage ei ny hending. Dette kan vere tilfellet dersom det er ei hending som slettar eit objekt, eller den inneheld feil data. I så fall må vi legge til eit ErrorDeclaration-element i hendingsdokumentet. Denne kan ha eit reason-felt, som skal innehalde *urn:epcglobal:cbv:er:* etterfulgt av eitt av to verdier: *did_not_occur* eller *incorrect_data*. Dersom vi brukar *incorrect_data* så kan vi følgje opp med ei ny hending med dei korrekte dataa.

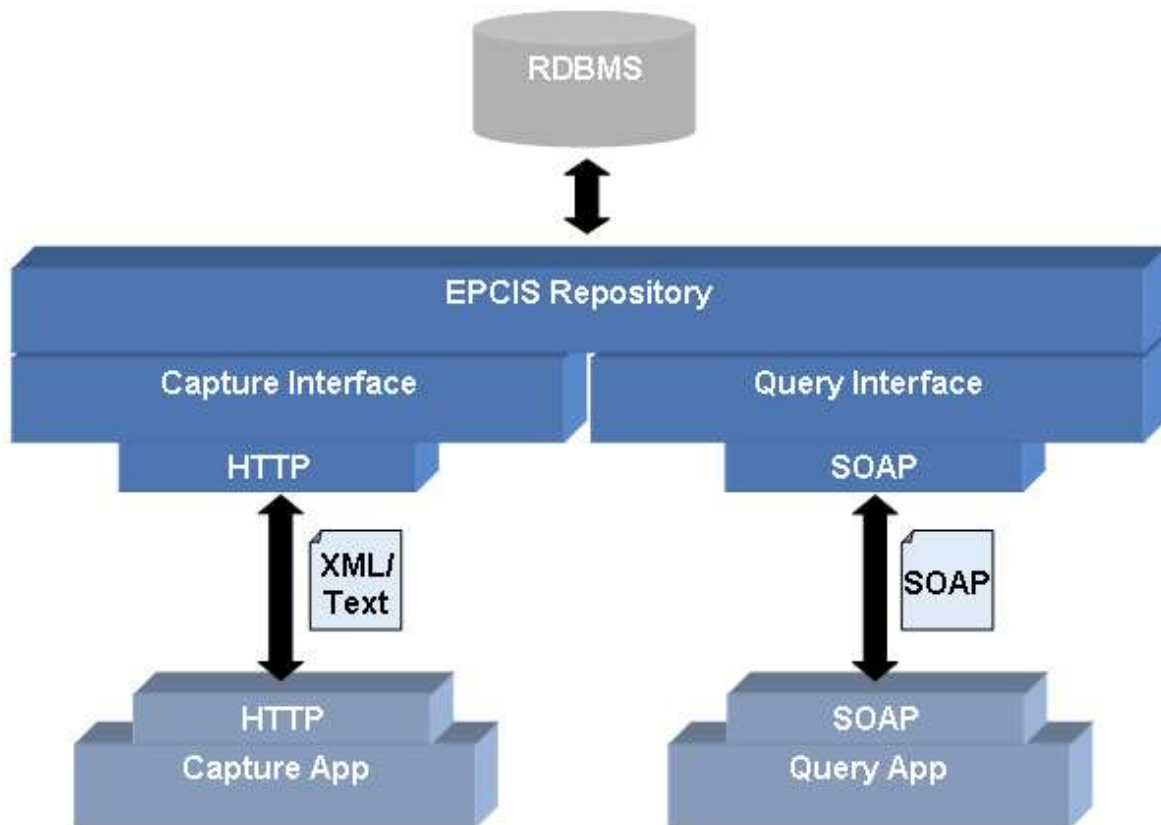
GUI-et til capture interface har ikkje noko felt for ErrorDeclaration, men vi kan bruke html-skjemaet til dette ved å legge til:

```
<ErrorDeclaration>
  <declarationTime>
    ...
  </declarationTime>
  <reason>
    ...
  </reason>
  <correctiveEventIDs>
    ...
  <correctiveEventIDs>
</ErrorDeclaration>
```

Der dei to siste felta (reason og correctiveEventIDs) er valfrie, men kan brukast til å oppgje grunn og kva hendingar som skal bli retta opp. (GS1 2017)

2.3.8 EPCIS repository

For at vi skal kunne dele hendingsdata må vi ha eit sentralt datalager. Det er dette som er EPCIS repository. Her lagrar vi data på ein mySQL-server, og dette skal brukarane få tilgjenge til ved hjelp av ein webserver.



Figur 1: Arkitektur for dei tre elementa EPCIS repository, EPCIS capture application og EPCIS query application. Frå <https://fosstrak.github.io/epcis/docs/architecture-guide.html>

2.3.9 Capture client

Datalageret har eit capture interface som kan ta i mot hendingar i HTTP POST. Eg kan bruke GUI-et til capture interface client til å sende slike hendingar. Eg kan også lage/fylle ut XML-dokumentet sjølv og sende det ved å bruke HTTP POST, til dømes ved å opne <http://localhost:8080/epcis-repository-0.5.0/capture> i nettlesaren og bruke skjemaet som er tilgjengeleg der. Dersom eg skriv inn følgjande data inn i GUI-en:

Event time: 2006-09-20T06:36:17Z

EPCs: urn:epc:id:sgtin:0057000.123780.7788

Action: ADD

Business step: urn:epcglobal:cbv:bizstep:commissioning

Disposition: urn:epcglobal:cbv:bizstep:active

Read point: urn:epc:id:sgln:0614141.00729.rp97

Business location: urn:epc:id:sgln:0614141.00729.loc5

Så tilsvarar det å sende EPCIS Capture Interface følgjande som eit XML-dokument:

```
<?xml version="1.0" encoding="UTF-8"?>
<epcis:EPCISDocument xmlns:epcis="urn:epcglobal:epcis:xsd:1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" creationDate="2008-03-
06T11:42:15.016+01:00" schemaVersion="1.0"
xmlns:myNs="http://my.unique.namespace">
  <EPCISBody>
    <EventList>
      <ObjectEvent>
        <eventTime>2006-09-20T06:36:17Z </eventTime>
        <eventTimeZoneOffset>+00:00</eventTimeZoneOffset>
        <epcList>
          <epc>urn:epc:id:sgtin:0057000.123780.7788</epc>
        </epcList>
        <action>ADD</action>
        <bizStep> urn:epcglobal:cbv:bizstep:commissioning </bizStep>
        <disposition> urn:epcglobal:cbv:bizstep:active </disposition>
        <readPoint>
          <id> urn:epc:id:sgln:0614141.00729.rp97</id>
        </readPoint>
        <bizLocation>
          <id>urn:epc:id:sgln:0614141.00729.loc5</id>
        </bizLocation>
        <!-- uncomment the following for capturing an event field extension -->
        <!-- <myNs:myExtensionField>My Extension Value</myNs:myExtensionField> -->
      </ObjectEvent>
    </EventList>
  </EPCISBody>
</epcis:EPCISDocument>
```

2.3.10 Query client

For å få data ut frå datalageret brukar vi query client. Denne har også eit GUI, der vi på ein brukarvenleg måte kan konstruere spørjingar mot datalageret. Query-klienten kan ta i mot spørringar i ein SOAP request. SOAP er ein XML-basert protokoll for å utveksle strukturert informasjon.

3.0 Analyse og design

I dette kapittelet vil eg først diskutere moglege bruksområde for RFID ved Helse Sør-Øst, og deretter skissere eit case som skal danne grunnlaget for resten av oppgåva.

3.1.1 Moglege bruksområde for RFID ved Helse Sør-Øst

RFID kan som nemnd brukast til å lokalisere utstyr. Eit sitat frå Jørgen Corneliussen i Sykehuspartner illustrerer dette:

«– Mange pasienter kommer inn via akutten. Der blir de tatt imot, lagt i en seng og koblet på en pumpe. De får utstyr oppi sengen, og så blir de trillet videre inn på sykehuset, beskriver han.

Det tekniske utstyret blir med videre til den avdelingen der pasienten blir innlagt. – Men det er ofte slik at utstyret ikke blir levert tilbake til akutten. Da må personell på akutten lete rundt på sykehuset for å finne pumpene og få dem tilbake igjen, sier Corneliussen.» (GS1 u.d.)

Med RFID kan dette utstyret bli registrert etterkvart som det passerer forskjellige lesepunkt på sjukehuset, slik som døra til pasienten sitt rom. Dei tilsette som jobbar på akutten kan då raskt finne ut kvar utstyret er utan å måtte forlate akuttavdelinga.

RFID-merking er også nyttig med tanke på vedlikehald av utstyr, sidan ein då ofte må finne ein av veldig mange like gjenstandar. Som Corneliussen seier:

«Hvis du har 500 like pumper rundt på sykehuset, kan det ta mye tid å finne riktig. Alt dette er mobilt utstyr som flytter seg rundt» (GS1 u.d.)

Det er lett å sjå korleis ein kan spare tid ved å lagre lokasjonsdata for kvar unike gjenstand på eit sentralt datalager. Det er også naturleg å lagre data om siste tidspunkt for vedlikehald i det same datalageret. Ei enkel spørjing kan dermed både gje oss svar på kva pumper som snart må inn til vedlikehald, og kvar på sjukehuset dei ligg.

I Drammen brukar ein RFID-tags til å registrere når sengar går inn og ut av vaskeriet. (GS1, 2022) Denne metoden kan ein overføre til vask og vedlikehald av anna utstyr, slik som rullestolar eller maskar.

Eit anna bruksområde er å halde oversikt over lagerstatus slik at ein kan få varsel eller sende bestilling automatisk når lageret er i ferd med å gå tomt. Med aktive RFID-tags med sensorar kan ein også overvake temperatur for gjenstandar og medisinar som må lagrast under ein viss temperatur. Ein kan også overvake eksponering for stråling. Dette kan vere aktuelt fordi mange sjukehus brukar strålekjelder, til dømes til strålebehandling eller nuklærmedisin. RFID kan også brukast til å spore personar ved å bruke innebygde RFID-brikkar i ID-kort eller liknande. Dette kan brukast til adgangskontroll, timeregistrering og smittesporing.

3.2 Skissering av case

Når ein brukar RFID til overvaking av utstyr er det i hovudsak to måtar å gjere det på. Ein kan sørge for at utstyret alltid er innanfor rekkevidde ein eller fleire lesarar, slik at ein kan få lokasjonsdata i sanntid. Ein vil då måtte ta i mot store mengder data, men til gjengjeld så vil ein ikkje gå glipp av nokre forflytningar. Dette kan til dømes vere aktuelt dersom ein ønskjer å sikre seg mot tjuveri eller uautorisert bruk. Den andre metoden er å føre utstyret gjennom eit avlesningspunkt kun når ein skal flytte på eller utføre ein handling på utstyret. Kvar avlesning vil då svare til ei hending innanfor EPCIS-modellen. Ein kan då tilføye

informasjon om kvifor forflytninga skjedde. Sidan eg er interessert i å lage diskrete hendingar utan å måtte tråle igjennom altfor store mengder rådata, vil eg bruke den siste metoden i denne oppgåva.

På eit sjukehus vil det til einkvar tid vere store mengder utstyr og medisinar. Desse blir teke ut av lager og inn på romma der dei skal brukast etter behov. Det er viktig for helseperonellet å vite kvar utstyret dei treng er når dei treng det, slik at dei slepp å bruke tid på å leite. Dette sparar både tid og stress, og gjer kvardagen til dei tilsette meir produktiv. I mange tilfelle er det også viktig å vite kvar utstyret har vore og kva som er gjort med det, særleg med tanke på vask og vedlikehald. Ved å innføre RFID-sporing av utstyr så vil all denne informasjonen kunne bli gjort tilgjengeleg for dei tilsette ved sjukehuset.

Det er ønskjeleg at utstyret skal kunne sporast på veg inn og ut av lokasjonar, slik som rom, hyller, skuffer osv. Dette skal kunne gjerast utan for mykje administrativt arbeid frå helsepersonellet sin side, og om mogleg heilt automatisk. Etterpå skal ein raskt og enkelt kunne slå opp kvar utstyret er.

Eg har ikkje tilgjenge til medisinsk utstyr eller eit sjukehus med mange rom, men kan likevel simulere eit case ved å bruke utstyret eg har til rådighet. Her er det sjølv sagt uendeleg med moglegheiter, men sidan dette kun er eit proof-of-concept så vil eg gjere det enklast mogleg. Så lenge eg ender opp med meiningsfulle data så har eg eit godt grunnlag for å kunne utvide løysninga til meir kompliserte løysningar med fleire antenner, fleire objekt og meir avanserte spørjingar.

Caset eg vil ta utgangspunkt i denne oppgåva tek utgangspunkt i at vi har eit lager med utstyr, og tek ein av gjenstandane ut av lageret og inn på operasjonssalen. Vi skal då kunne finne lokasjonen både til det som er teke med på operasjonssalen og det som ligg at på lageret, samt når forflytningane skjedde. Dette caset simulerer eg på følgjande måte:

Eg har to lokasjonar, lager/lokasjon 1 og operasjonssal/lokasjon 2. Antenne 1 svarar til lageret og antenne 4 svarar til lokasjon 2.

Først les eg av 2 gjenstandar på lageret (lokasjon 1)

Gjenstand 1 blir lest av på antenne 1 og tas ut av lageret

Gjenstand 1 blir lest av på antenne 2 og tas inn i operasjonssalen

Forventa status: Gjenstand 1 er på operasjonssalen. Gjenstand 2 er på lageret. Eg kan lese av tid for hendingane ved hjelp av query interface.

Impinj Itemtest har ein loggfunksjon som kan skrive dataene til ei csv-fil. Denne fila kan eg importere til Excel. I Excel så kan eg lese av EPC, tidspunkt og kva antenne som er brukt. Dette dannar grunnlaget for dataa som eg legg inn i capture interface client.

Alternativt kan eg også lese dataa direkte frå Impinj Itemtest.

4.0 Implementering og testing

I dette kapittelet skal eg beskrive korleis eg reint praktisk sette opp systemet og gjorde eit forsøk med lesing av RFID-hendingar.

4.1 Oppsett av IT-miljø

IT-miljøet består av programvare som eg har installert på eigen PC, og maskinvare som er plassert på eit rom på Høgskolen i Molde. I dette kapittelet vil eg beskrive korleis eg gjorde dette.

4.1.1 Programvare

På <https://fosstrak.github.io/epcis/> ligg det to installasjonsguidar: developer guide og user guide. Eg begynte installasjonen med å følgje developer guide, men fann etter kvart ut at det var nok å bruke user guide til denne oppgåva.

Eg satte opp eit Fosstrak EPCIS repository med capture og query clients etter instruksjonane på <https://fosstrak.github.io/epcis/docs/user-guide.html>. Som del av dette må ein installere Apache Tomcat Servlet Container og ein SQL server. For å ta i mot data frå RFID-lesaren installerte også Impinj Itemtest 1.14.0. Den siste versjonen av Impinj Itemtest (per mai 2022) er 2.7.0, men på grunn av utfordringar med å få denne til å virke så brukte eg versjon 1.14.0 i staden.

Eg støtte på ein del utfordringar med å få programvaren til å køyre. Dette gjaldt spesielt query interface client, som ikkje ville la seg køyre i Java Platform SE Binary. Det tok ein del tid før eg fann ut at denne kunne køyrast i OpenJDK Platform Binary.

4.1.2 Maskinvare

Eg fekk låne følgjande utstyr av Bjørn Jæger:

1 Impinj Speedway Revolution R420 RFID UHF

Antenner

RFID -tags

Impinj Speedway Revolution er ein RFID-lesar med 4 portar for antenner. Eg kopla 2 antenner på denne og kopla den til høgskulens nett gjennom Ethernet-porten. Heile caset er simulert på eitt rom, men så lenge eg kan legge antennene langt nok frå kvarandre at dei kan lese av individuelt utan «forureiningar» av dataa så fungerer dette som ein simulasjon av to rom. I ein reell situasjon der ein må bruke fleire lesarar på grunn av avstanden mellom lokasjonane er det tilsynelatande trivielt å konfigurere Impinj Itemtest til å ta i mot avlesningar frå fleire lesarar.

Ei utfordring ved oppsettet av utstyr var å få forbindelse mellom min PC og RFID-lesaren. Lesaren eg brukte er laga for POE (Power Over Ethernet), det vil seie at den får straum gjennom Ethernet-kabelen. PC-en min er ikkje laga etter denne standarden, og kunne difor ikkje forsyne lesaren med straum frå PC-en sin Ethernet-port. Eg måtte difor kople lesaren til eit nettverksuttak i veggen i staden. Sidan RFID-lesaren dermed var kopla til Høgskulen sitt intranett så måtte eg få tilgjenge til dette. Dette fekk eg god hjelp til av IT-avdelinga på Høgskolen.

4.2 Lesing av RFID-events

4.2.1 Bruk av RFID-lesaren

På RFID-lesaren kan ein kople opp til 4 antenner, eller opptil 32 ved bruk av antennehubs. For å kunne lese av RFID-hendingar på ein PC må ein køyre Impinj Itemtest eller tilsvarande programvare. Dersom ein ønskjer ein logg av RFID-hendingane må ein skrive inn eit filnamn i feltet for Log file i nedste venstre hjørne. Når ein så fører RFID-brikka til antenna, så vil avlesningane komme opp i Impinj Itemtest. Deretter kan ein ta vekk RFID-brikka igjen. Det er viktig at det er ein viss avstand mellom antennene og RFID-brikkene når ein ikkje skal gjere avlesningar, sånn at ein ikkje får «skitne» data. I Impinj Itemtest kan ein velje kva kolonner som skal visast. Her bør ein iallfall ha first timestamp og/eller last timestamp. Dersom ein brukar fleire enn ein antenne, slik som eg skal gjere i dette forsøket, treng ein også å vise kolonna for antenne. Dataa som visast i Impinj Itemtest eller i loggen kan då danne grunnlaget for dataa som skal tastast inn i capture client.

4.2.2 When

I loggfila laga av Impinj Itemreader ligg det timestamps, i formatet mikrosekund etter 1. januar 1970, for kvar rå RFID-hending. Eg fann første og siste timestamp for kvar EPC på kvar antenne og reknar ut gjennomsnittet. Resultatet konverterer eg til millisekund og konverterer så til ISO-6801 ved hjelp av <https://www.vercalendario.info/en/how/convert-timestamp-date.html>. Dette limer eg inn i feltet «Event time» i capture interface client.

4.2.3 What

Gjenstandane som blir flytta må få kvar sin unike EPC. Dei RFID-tagane eg brukte har EPC 000000000000317649610589 og 000000000000317649610637. Sidan eg ikkje har kunne tilordne GTIN-nummer eller liknande til desse så blir inputen til capture-klienten urn:raw:96:000000000000317649610589 og urn:raw:96:000000000000317649610637

4.2.4 Where

Eg har valt å gje lageret 0708012341000 som GLN, operasjonssalen 0708012342000 og sjukehuset 0708012340000. Strengen som skal inn i capture interface blir dermed `urn:epc:id:sgln:708012341000..0` for lageret.

I dette forsøket brukar eg antenne 1 til å representere Read Point på lageret, og antenne 4 til å representere Read Point på operasjonssalen. Sidan eg berre har ei antenne på kvart «rom», så gjer eg antennene same GLN som romma. Når eg flyttar eit objekt inn på eit rom, blir då Business Location det same som Read Point, og når eg flyttar objektet ut så blir Business Location sjukehuset.

4.2.5 Why

Dette feltet er valfritt, men vi kan bruke det til å lagre informasjon om varen gjekk inn (mottak) eller ut (plukking) av lokasjonen. Alle data som skrives inn i Business Step skal vere på forma `urn:epcglobal:cbv:disp:payload`, der *payload* er ein av strengene definert i standarden. (GS1 2017) Dersom den kom inn skriv vi `urn:epcglobal:cbv:bizstep:receiving` og dersom den gjekk ut skriv vi `urn:epcglobal:cbv:bizstep:picking`.

4.2.6 Capture data

Eg les av timestamp, EPC og antenne frå reknearka i vedlegg 7.1 og 7.2. Eg reknar ut gjennomsnittleg tid for kvar hending og noterer meg antennennummeret. Antenne 1 tilsvarar read point 708012341000..0 og antenne 2 tilsvarar read point 708012342000..0. Basert på dette så blir dataa eg skriv inn i capture interface:

Første hending (basert på data i vedlegg 7.1):

Timestamp 1653631428879380 = 2022-05-27T08:03:48.87Z

Event time: 2022-05-27T08:03:48.87Z

EPCs: urn:epc:raw:96.0000000000000317649610589 urn:epc:raw:96.

0000000000000317649610637

Read point: urn:epc:id:sgln:708012341000..0
Business location: urn:epc:id:sgln:708012341000..0
Business step: urn:epcglobal:cbv:bizstep:receiving

Andre hending (basert på data i vedlegg 7.2):
Timestamp 1653631444600840 = 2022-05-27T08:04:60Z

Event time: 2022-05-27T08:04:60Z
EPCs: urn:epc:raw:96.0000000000000317649610589
Read point: urn:epc:id:sgln:708012341000..0
Business location: urn:epc:id:sgln:708012340000..0
Business step: urn:epcglobal:cbv:bizstep:picking

Tredje hending (basert på data I vedlegg 7.2):
Timestamp 1653631449075400 = 2022-05-27T08:09:07Z

Event time: 2022-05-27T08:09:07Z
EPCs: urn:epc:raw:96.0000000000000317649610589
Read point: urn:epc:id:sgln:708012342000..0
Business location: urn:epc:id:sgln:708012342000..0
Business step: urn:epcglobal:cbv:bizstep:receiving

4.3 Resultat

I GUI-et til query interface fylte eg ut felte for query arguments slik:
EPC = urn:epc:raw:96.0000000000000317649610589 urn:epc:raw:96.
0000000000000317649610637
Record time >= 2022-05-27T16:19:00+02:00

Record Time er lagt inn som argument fordi eg har gjort fleire forsøk på å legge inn hendingar med desse to EPC-ane, og berre vil ha den «riktige» avlesninga.

Eg fekk følgande relevante data:

Første hending:

Event time: 2022-05-27T08:03:48.870+02:00

EPCs: urn:epc:raw:96.000000000000317649610589 urn:epc:raw:96.000000000000317649610637

Read point: urn:epc:id:sgln:708012341000..0

Business location: urn:epc:id:sgln:708012341000..0

Business step: urn:epcglobal:cbv:bizstep:receiving

I klartekst: Den 27.5.2022 kl. 08:03:48.87 norsk tid blei 2 objekt med EPC 000000000000317649610589 og 000000000000317649610637 teke inn på lageret.

Andre hending:

Event time: 2022-05-27T08:04:04.600+02:00

EPCs: urn:epc:raw:96.000000000000317649610589

Read point: urn:epc:id:sgln:708012341000..0

Business location: urn:epc:id:sgln:708012340000..0

Business step: urn:epcglobal:cbv:bizstep:picking

I klartekst: Den 27.5.2022 kl. 08:04:04.60 norsk tid blei objektet med EPC 000000000000317649610589 teke ut av lageret.

Tredje hending:

Event time: 2022-05-27T08:09:07.0+02:00

EPCs: urn:epc:raw:96.000000000000317649610589

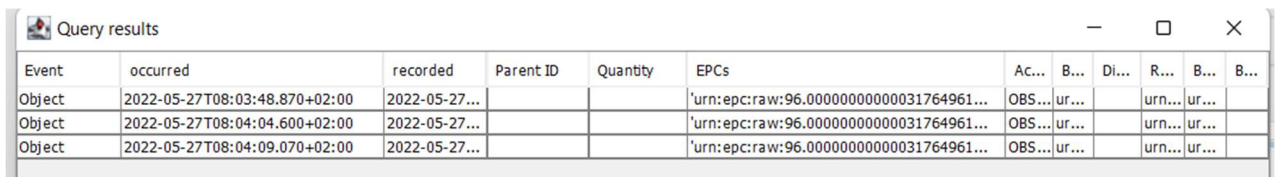
Read point: urn:epc:id:sgln:708012342000..0

Business location: urn:epc:id:sgln:708012342000..0

Business step: urn:epcglobal:cbv:bizstep:receiving

I klartekst: Den 27.5.2022 kl. 08:04:09.07 norsk tid blei objektet med EPC 000000000000317649610589 teke inn på operasjonssalen.

Interessant nok blei tidspunktet for hendingane lagra i den lokale tidssona (UTC+2) på trass av at dei var tasta inn som UTC. Avlesningane stemte heller ikkje med det faktiske klokkeslettet då eg registrerte hendingane, men dette har sannsynlegvis å gjere med den interne klokka til lesaren. Bortsett frå det så er dataa som forventa identiske med det som eg skreiv inn i GUI-en til capture interface.



Event	occurred	recorded	Parent ID	Quantity	EPCs	Ac...	B...	Di...	R...	B...	B...
Object	2022-05-27T08:03:48.870+02:00	2022-05-27...			'urn:epc:raw:96.00000000000031764961...	OBS...	ur...		urn...	ur...	
Object	2022-05-27T08:04:04.600+02:00	2022-05-27...			'urn:epc:raw:96.00000000000031764961...	OBS...	ur...		urn...	ur...	
Object	2022-05-27T08:04:09.070+02:00	2022-05-27...			'urn:epc:raw:96.00000000000031764961...	OBS...	ur...		urn...	ur...	

Figur 2: Resultatet av spørjinga gjort med Query Interface Client. Denne er noko knotete å bruke, særleg sidan ein ikkje kan kopiere frå den.

5.0 Konklusjonar

Eg har i denne oppgåva laga eit IT-miljø for registrering av RFID-hendingar. Eg har brukt dette til å lese av rå RFID-data og registrere dei som EPC forretningshendingar. Tid, lokasjonsdata og forretningssteg er registrert i samsvar med GS1-standardane. Når det gjeld EPC så burde den ha vore registrert som GTIN eller UDI, men sidan eg ikkje har kunne endre på EPC-ane på RFID-tagane har dette ikkje vore i samsvar med GS1. Eg har omtala korleis ein EPC skal vere i samsvar med GS1.

Eg har lagt forretningshendingane inn i EPCIS repositoryet eg har laga ved hjelp av GUI-en til capture interface client. Eg har også diskutert korleis denne kan leggest inn som eit XML-dokument.

Eg har lese av dataa ved hjelp av query client. Desse dataa samsvarte med dataa eg tasta inn i capture GUI-en.

5.1 Manglar og forslag til forbetringar

Som nemnd tidlegare så burde EPC-ane på objekta vere i samsvar med GS1. For medisinsk utstyr så betyr dette at kvart objekt skal ha Unique Device Identification (UDI). (GS1 2022) UDI består av GTIN og ein dynamisk del (AI). (GS1 2017) På trass av at dette ikkje er gjort så meiner eg at eg har løyst problemstillinga, sidan eg har spora to objekt med unike EPC.

5.2 Vidare arbeid

Eit neste steg vil vere å installere eller lage middleware for å kunne sende dataen rett frå RFID-lesaren til capture-klienten utan at ein må taste inn alle dataa manuelt.

Ein kan lage meir avanserte case, til dømes ved å fininndele romma etter hylleplass og liknande, eller montere ei antenne i kvart rom i ei bygning. Prøve forskjellige typar RFID-brikker og sensitivitet på lesaren for å finne det som er optimalt for den spesifikke lokasjonen for å unngå forureining av data og manglande data.

6.0 Referanser

GS1. 2017. «Core Business Vocabulary Standard.» Oktober.

<https://www.gs1.org/sites/default/files/docs/epc/CBV-Standard-1-2-2-r-2017-10-12.pdf>.

—. 2016. «EPC Information Services (EPCIS).» *gs1.org*. September.

<https://www.gs1.org/sites/default/files/docs/epc/EPCIS-Standard-1.2-r-2016-09-29.pdf>.

—. 2017. «EPC Tag Data Standard.» Mars.

https://www.gs1.org/sites/default/files/docs/epc/GS1_EPC_TDS_i1_10.pdf.

—. 2022. *EPC/RFID*. Mai. <https://gs1.no/epc-rfid/>.

—. u.d. *EPCIS*. Funnet Mai 26, 2022. <https://www.gs1.org/standards/epcis>.

- . 2017. «EPCIS and CBV Implementation Guideline.» Februar.
https://www.gs1.org/docs/epc/EPCIS_Guideline.pdf.
- . 2017. *GSI Guide on Unique Device Identification (UDI)*. Mars.
https://www.gs1.org/sites/default/files/docs/healthcare/position-papers/gsl_udi_guide_final_20170324.pdf.
- . 2022. *gs1.org GTIN*. Mai. <https://gs1.no/gtin/>.
- . u.d. *Slutt på å lete etter medisinsk utstyr*. Funnet Mai 26, 2022. <https://gs1.no/slutt-paa-lete-etter-medisinsk-utstyr/>.
- . 2022. *Veileder for bruk av GSI-standarder*. Mai. Funnet Mai 26, 2022.
<https://gs1.no/veileder-for-bruk-av-gs1-standarder/>.
2022. *Helse Sør-Øst*. Mai. <https://helse-sorost.no/om-oss#om-helse-sor-ost-rhf>.
- Helse Sør-Øst. 2022. *Entydig Strekkoding*. Mai. <https://helse-sorost.no/om-oss/vart-opdrag/hva-gjor-vi/digitalisering-og-e-helse/regionale-ikt-prosjekter/entydig-strekkoding-gs1>.
- Lowry Solutions. 2022. *The Difference Between Barcode and RFID Technology*. Mai.
<https://lowryolutions.com/blog/the-difference-between-barcode-and-rfid-technology/>.
- Paragon ID. 2022. *The History of Radio Frequency Identification Technology*. Mai.
<https://www.paragon-id.com/en/inspiration/history-radio-frequency-identification-technology>.
- Phillips, Gavin. 2017. *How Does RFID Technology Work?* 31 Mai.
<https://www.makeuseof.com/tag/technology-explained-how-do-rfid-tags-work/>.
2022. *Sepaco Tech*. Mai.
https://web.archive.org/web/20170809131526id_/http://www.sepaco-tech.com/modules/Manager/Articles/the%20history%20of%20rfid.pdf.

7.0 Vedlegg

7.1 1. avlesning av rådata – begge objekt inn på lager

Column1	Column2	Column3	Column4	Column5
// 27.05.2022 13:39:02				

// AntennaIDs=1	2	3	4	InventoryMode=Single_Target
// Timestamp	EPC	Antenna	RSSI	Channel
1,65363E+15	000000000000317649610589	1	-64	865.7
1,65363E+15	000000000000317649610589	1	-66.5	865.7
1,65363E+15	3,1765E+11	1	-62.5	865.7
1,65363E+15	000000000000317649610589	1	-58.5	865.7
1,65363E+15	000000000000317649610589	1	-58	865.7
1,65363E+15	000000000000317649610589	1	-58	865.7
1,65363E+15	000000000000317649610589	1	-59	865.7
1,65363E+15	000000000000317649610589	1	-64	865.7
1,65363E+15	000000000000317649610589	1	-64	865.7
1,65363E+15	000000000000317649610589	1	-62.5	865.7
1,65363E+15	000000000000317649610589	1	-66.5	865.7
1,65363E+15	000000000000317649610637	1	-65	865.7
1,65363E+15	000000000000317649610637	1	-60.5	865.7
1,65363E+15	000000000000317649610637	1	-57.5	865.7
1,65363E+15	000000000000317649610637	1	-52	865.7
1,65363E+15	3,1765E+11	1	-49.5	865.7
1,65363E+15	3,1765E+11	1	-49	865.7
1,65363E+15	3,1765E+11	1	-49.5	865.7
1,65363E+15	000000000000317649610637	1	-50	865.7
1,65363E+15	000000000000317649610637	1	-50	865.7
1,65363E+15	000000000000317649610637	1	-51.5	865.7
1,65363E+15	3,1765E+11	1	-55	865.7
1,65363E+15	000000000000317649610637	1	-60	865.7

1653631428879380

1653631430220290

7.2 2. avlesning av rå data – Eit objekt ut av lageret og inn på operasjonssalen

Column1	Column2	Column3	Column4	Column5
// 27.05.2022				
13:39:20				
// AntennaIDs=1	2	3	4	InventoryMode=Single_Target
// Timestamp	EPC	Antenna	RSSI	Channel
1,65363E+15	000000000000317649610589	1	-63.5	865.7
1,65363E+15	000000000000317649610589	1	-59.5	865.7
1,65363E+15	000000000000317649610589	1	-58.5	865.7
1,65363E+15	000000000000317649610589	1	-60.5	865.7
1,65363E+15	000000000000317649610589	1	-65	865.7
1,65363E+15	000000000000317649610589	1	-63.5	865.7
1,65363E+15	000000000000317649610589	1	-62.5	865.7

1,65363E+15	000000000000317649610589	1	-60.5	865.7
1,65363E+15	000000000000317649610589	1	-60	865.7
1,65363E+15	000000000000317649610589	1	-60.5	865.7
1,65363E+15	000000000000317649610589	4	-66	865.7
1,65363E+15	000000000000317649610589	4	-65.5	865.7
1,65363E+15	000000000000317649610589	4	-65.5	865.7
1,65363E+15	000000000000317649610589	4	-63.5	865.7
1,65363E+15	000000000000317649610589	4	-62	865.7
1,65363E+15	000000000000317649610589	4	-60.5	865.7
1,65363E+15	000000000000317649610589	4	-60	865.7
1,65363E+15	000000000000317649610589	4	-59	865.7
1,65363E+15	000000000000317649610589	4	-60	865.7
1,65363E+15	000000000000317649610589	4	-60	865.7
1,65363E+15	000000000000317649610589	4	-60.5	865.7
1,65363E+15	000000000000317649610589	4	-60.5	865.7
1,65363E+15	000000000000317649610589	4	-59	865.7
1,65363E+15	000000000000317649610589	4	-58	865.7
1,65363E+15	000000000000317649610589	4	-57	865.7
1,65363E+15	000000000000317649610589	4	-55.5	865.7
1,65363E+15	000000000000317649610589	4	-53.5	865.7
1,65363E+15	000000000000317649610589	4	-53.5	865.7
1,65363E+15	000000000000317649610589	4	-54	865.7
1,65363E+15	000000000000317649610589	4	-54	865.7
1,65363E+15	000000000000317649610589	4	-55	865.7
1,65363E+15	000000000000317649610589	4	-56.5	865.7
1,65363E+15	000000000000317649610589	4	-57.5	865.7
1,65363E+15	000000000000317649610589	4	-56.5	865.7
1,65363E+15	000000000000317649610589	4	-57.5	865.7
1,65363E+15	000000000000317649610589	4	-56.5	865.7
1,65363E+15	000000000000317649610589	4	-55.5	865.7
1,65363E+15	000000000000317649610589	4	-55.5	865.7
1,65363E+15	000000000000317649610589	4	-54	865.7
1,65363E+15	000000000000317649610589	4	-54.5	865.7
1,65363E+15	000000000000317649610589	4	-55	865.7
1,65363E+15	000000000000317649610589	4	-56	865.7
1,65363E+15	000000000000317649610589	4	-58.5	865.7

ut 1653631444600840
inn 1653631449075400