

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Informaatikainstituut

Teadmussüsteemide õppetool

**Ravitööd toetava infosüsteemi
reeglipõhine otsustustoe tugisüsteem**

Magistritöö

Üliõpilane: Villu Jürimaa

Üliõpilaskood: 124495 IABMM

Juhendaja: Jaak Tepandi

Tallinn
2016

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

(kuupäev)

(allkiri)

Annotatsioon

Käesoleva magistritöö teemaks on „Ravitööd toetava infosüsteemi reeglipõhine otsustustoe tugisüsteem“. Töös vaadeldakse olemasolevaid ravitööd toetavaid otsustustoe süsteeme, luuakse analüüsi projekt võimaliku uue otsustustoe tugisüsteemi loomiseks ning antakse soovitusel kirjeldatud süsteemi rakendamiseks.

Töö eesmärgiks on kavandada otsustustoe tugiinfo kasutuselevõtt ravitööd toetava infosüsteemi juures, kasutades selleks otsustustoe tugisüsteemi.

Töös käsitletakse probleeme, mis on seotud erineva lähtekoha ning klassifikatsiooniga informatsiooni tõlgendamise ning ühendamise. Samuti vaadeldakse reeglisüsteemi rakendamisel tekkivaid küsimusi nii sobiliku järelusmootori valiku ja rakendamise kui ka reeglibaasi andmemudeli struktuuri ning reeglibaasi haldamise osas.

Töös analüüsitakse otsustustoe tugisüsteemile esitatavaid nõudeid, vaadeldakse olemasolevaid samalaadseid süsteeme ning otsustatakse uue süsteemi loomise mõttekus. Vaadeldakse väliseid otsustustoe infoallikaid koos iseärasustega ning luuakse väliste andmeallikate kavandatava süsteemi juurde integreerimise kava. Kirjeldatakse süsteemi sisemise reeglibaasi nõudeid ning luuakse reeglibaasi ning reeglite haldusliidese kirjeldus. Teostatakse reeglibaasi juures kasutatava järelusmootori valik ning kirjeldus järelusmootori tugisüsteemi juures rakendamiseks. Kirjeldatakse erinevate osapoolte andmemudelid ning esitatakse andmemudelites oleva informatsiooni ühtsel kujul edastamise põhimõtted.

Lõpuks kirjeldatakse võimalusi töö tulemusena kavandatud otsustustoe tugisüsteemi integreerimiseks ravitöö infosüsteemiga ning mõningaid võimalikke kavandatud tugisüsteemi edasiarendamise suundi.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 77 leheküljel, 6 peatükki, 18 joonist, 12 tabelit.

Abstract

Jürimaa, V. (2016). Rule-based decision support system for clinical information system. Master's Thesis, Tallinn University of Technology.

In this thesis the author studies existing clinical decision support systems and analyses the need to create a new custom system. The author produces an analysis project for creation the new system and gives recommendations for integrating the new decision support system with the processes of the clinical information system.

The main goal of the thesis is to allow clinical decision support information to be used for supporting the processes of the clinical information system. The description of the decision support information system is created to provide the possibility. The thesis will address the concerns of interpreting differently classified information from different sources as well as building a suitable data model, management system and inference engine for a rule-based decision support system.

The author of the thesis describes the requirements for the decision support system, considers a list of existing decision support systems and analyses the need for creating a new one. External decision information sources and their features are examined and a scheme is created to integrate them with the planned decision support system. The list of requirements is created for the internal rules database of the planned system and a description is provided for developing the rule base and its management system. The comparison and selection of an inference engine is conducted and a description is created for integrating the engine with the planned system. Data models of the different parties of the planned system are considered and the description is created for integrating them in the planned system.

Finally some possibilities are described for integrating the planned system with the clinical information system and for supporting its clinical business processes.

The thesis is in Estonian and contains 77 pages of text, 6 chapters, 18 figures, 12 tables.

Lühendite ja mõistete sõnastik

Kliiniliste otsuste tugisüsteem	<i>Clinical Decision Support System (CDSS)</i> Infotehnoloogilisi vahendeid kasutatav süsteem, mis integreerides meditsiiniliste teadmiste baasi, patsiendi andmed ning järeldusmootori, genereerib erialaseid nõuandeid.
Teadmussüsteem	Infotöötlussüsteem, mis aitab lahendada teatava ainevalla või rakendusala probleeme teadmusbasisest järelduste tegemisega (1 lk 60).
Järeldusmootor	<i>Inference engine</i> Tehisintellekti meetodeid rakendav arvutisüsteem või selle osa, mis olemasoleva teadmiste baasi peal reegleid rakendades võimaldab leida uusi teadmisi.
Andmepõhine järeldamine	Teadmussüsteemi reeglite põhjal rakendatav tegevus, mis püüab olemasolevate andmete põhjal tekitada võimalikult palju järeldusi. (1 lk 30)
SNOMED CT	<i>Systematized Nomenclature of Medicine - Clinical Terms</i> Laiaulatuslike meditsiiniterminoloogia, mille eesmärk on võimaldada meditsiinilise info hoiustamist ning tähendusliku info leidmist (s.h. arvutisüsteemide poolt). (2)
EBMeDS	<i>Evidence-Based Medicine electronic Decision Support</i> Tõenduspõhine meditsiiniline otsustustoe süsteem, mis on välja töötatud Soomes ning mille kasutuselevõttu kavandatakse ka Eestis.
SFINX-PHARAO	Soome ja Rootsi meditsiinitöötajate seas enimkasutatud ravimite koostoimete andmebaas, mida on alates 2014 aastast hakatud kasutama ka Eestis.

LOINC	<i>Logical Observation Identifiers Names and Codes</i> Laboratoorsete mõõtmistulemuste kirjeldamiseks kasutatav universaalne terminite ning ühikute andmebaas ning standard.
RHK-10, ICD-10	<i>Rahvusvaheline Haiguste Klassifikatsioon</i> Rahvusvaheline haiguste ning nendega seotud terviseprobleemide statistilise klassifikatsiooni versioon number 10.
SOA	<i>Service - Oriented Architecture</i> Tarkvara arhitektuuri muster, kus tarkvara komponendid pakuvad üksteisele ühtse andmevahetuskihi kaudu erinevaid teenuseid.
IHE XDS	<i>Cross-Enterprise Document Sharing</i> Standardite süsteem patsiendi meditsiiniandmete jagamiseks erinevate osapoolte vahel.
IHTSDO	<i>International Health Terminology Standards Development Organisation</i> Rahvusvahelisi meditsiiniterminoloogia standardeid haldav ning arendav organisatsioon, mis ühendab paljusid liikmesriike.
BRMS	<i>Business Rule Management System</i> Tarkvarasüsteem, mis võimaldab organisatsiooni tööprotsesse toetavate otsustusreeglite kirjeldamist, haldamist ning rakendamist.
API	<i>Application Programming Interface</i> Tööpõhimõtete, protokollide ning tööriistade kogum arvutisüsteemide suhtluse korraldamiseks. (3)
HL7	<i>Health Level Seven International</i> Rahvusvaheline mittetulunduslik standardeid väljatöötav organisatsioon, mille eesmärgiks on töötada välja välja standardid meditsiinilise informatsiooni hoiustamiseks ning edastamiseks. (4)

Jooniste nimekiri

Joonis 1: Kavandatava tugisüsteemi komponendid ning interaktsioonid.....	13
Joonis 2: Ravitöö infosüsteemi valikuline isikustamata andmemudel	22
Joonis 3: Tervise Infosüsteemi otsustustoe päringu andmemudel	23
Joonis 4: Tervise Infosüsteemi otsustustoe päringu vastuse andmemudel.....	24
Joonis 5: Ravimi koostoimete päringu andmemudel.....	25
Joonis 6: Ravimi koostoimete päringu vastuse andmemudel.....	25
Joonis 7: Otsustustoe tugisüsteemi arhitektuur	30
Joonis 8: Sisemise reeglibaasi loogiline andmemudel	38
Joonis 9: Otsustustoe tugisüsteemi integratsioonisõnumite jadadiagramm	39
Joonis 10: Otsustustoe tugisüsteemi ja välise otsustustoe süsteemide integratsiooni- mooduli arhitektuur.....	42
Joonis 11: Otsustustoe tugisüsteemi integratsioonimooduli adapteri arhitektuur	43
Joonis 12: Sisemise reeglibaasi füüsiline andmemudel.....	44
Joonis 13: Otsustustoe päringu andmemudel	46
Joonis 14: Otsustustoe vastuse andmemudel.....	46
Joonis 15: Järeldusmootori arhitektuur	58
Joonis 16: Järeldusmootori protsesside jadadiagramm	59
Joonis 17: SNOMED CT termini omadused	62
Joonis 18: Otsustustoe tugisüsteemi ning ravitöö infosüsteemi integratsiooniprotsesside jadadiagramm	67

Tabelite nimekiri

Tabel 1: Olemasolevad kliiniliste otsuste tugisüsteemid.....	19
Tabel 2: Parameetrid Eesti ravijuhendite kasutamiseks kavandatavas otsustustoe tugisüsteemis	27
Tabel 3: Sisemise reeglibaasi poolt toetatavad parameetrid.....	32
Tabel 4: Sisemise reeglibaasi elementide kirjeldus.....	33
Tabel 5: Sisemise reeglibaasi haldusliidese funktsionaalsed nõuded.....	35
Tabel 6: Sisemise reeglibaasi haldusliidese mittefunktsionaalsed nõuded	36
Tabel 7: Sisemise reeglibaasi haldusliidese omadused	37
Tabel 8: Järeldusmootorile esitatavad funktsionaalsed nõuded	49
Tabel 9: Järeldusmootorile esitatavad mittefunktsionaalsed nõuded	50
Tabel 10: Valimis osalenud järeldusmootorite omaduste võrdlus	54
Tabel 11: Valimis osalenud järeldusmootorite vastavus esitatud nõuetele.....	56
Tabel 12: Valimis osalenud järeldusmootorite punktisummad	57

Sisukord

1. Sissejuhatus	11
1.1 Taust ja probleem	11
1.2 Ülesande püstitus	13
1.3 Metoodika	14
1.4 Ülevaade tööst	15
2. Olemasolevate otsustustoe tugisüsteemide kirjeldus.....	16
2.1 Vaadeldud kliiniliste otsuste tugisüsteemid	17
2.2 Järeldused	19
3. Väliste andmeallikate andmemudelite kirjeldus.....	21
3.1 Ravitöö infosüsteemi valikulise andmemudeli kirjeldus.....	21
3.2 Eesti Tervise Infosüsteemi otsustustoe allsüsteemi andmemudeli kirjeldus	23
3.3 Ravimi koostoimete süsteemi andmemudeli kirjeldus	24
3.4 Eesti ravijuhendite andmebaasi andmemudeli kirjeldus	26
3.5 Järeldused	28
4. Otsustustoe tugisüsteemi kirjeldus	29
4.1 Süsteemi arhitektuur	30
4.2 Sisemise reeglibaasi kirjeldus.....	31
4.2.1 Sisemise reeglibaasi haldusliidese kirjeldus.....	34
4.3 Integratsiooni kirjeldus	39

4.4 Andmemudeli kirjeldus	40
4.4.1 Integreeritavate süsteemide andmemudelite ühildamise kava	41
4.4.2 Sisemise reeglibaasi andmemudeli kirjeldus.....	43
4.4.3 Integratsioonisõnumite andmemudelite kirjeldus.....	45
4.5 Järeldusmootori kirjeldus	47
4.5.1 Järeldusmootorile esitatavad nõuded.....	49
4.5.2 Vaadeldud järeldusmootorite ülevaade	50
4.5.3 Järeldusmootori valik	53
4.5.4 Järeldusmootori rakendamise kirjeldus	58
4.6 Edasiarendamise võimalused.....	61
4.6.1 SNOMED CT terminoloogia rakendamine	61
4.6.2 Raviandmete põhjal reeglite tuletamine	63
4.7 Kokkuvõte	64
5. Otsustustoe tugisüsteemi ning ravitöö infosüsteemi integratsiooni kirjeldus	66
6. Kokkuvõte	69
Summary.....	71
Kasutatud kirjandus	73
Lisa 1 - Sisemise reeglibaasi haldusliidese ekraanivormide prototüübid.....	76
Lisa 2 - Sisemise reeglibaasi reegli andmekoosseisu näide	77

1. Sissejuhatus

IT-lahendused on olnud Eesti tervishoiu lahutamatuks osaks juba rohkem kui 20 aastat. Täpsemalt tekkis perearstipraksistel kohustus arvuti ja internetiühenduse omamiseks aastal 2002 ning aastal 2003 oldi selle poolest esirinnas kogu maailmas. Tervishoiuinformaatika ajalugu sai aga alguse teadustööde käigus juba koos esimeste arvutitega 1950. aastatel USAs. (5 lk 1)

Selle ajaga on meditsiinasutuste IT-süsteemidesse kogunenud ka märkimisväärne hulk infot, mille maht suureneb pidevalt ning mida oleks otstarbekas nii ravitöö kvaliteedi tõstmiseks kui kasutajate töö hõlbustamiseks ära kasutada.

2008. aastal loodud riikliku Tervise Infosüsteemi eesmärgiks on muuhulgas samuti ravitöö kvaliteedi tagamine. Tegemist on keskse andmekoguga, mis sisaldab erinevate registrite kaudu teavet rahvastiku tervisliku seisundi kohta ning mis võimaldab seda infot ka ravitöös kasutada. 2015 aasta 2. mai seisuga oli Tervise Infosüsteemi kesksüsteemis meditsiinidokumente 1 358 019 inimesel, seega valdaval osal Eesti rahvastikust (6).

Eestis on alustatud ka tegevustega, võtmaks digitaalsetes tervise infosüsteemides kasutusele tsentraalne eestikeelne kliiniliste otsuste tugisüsteem, mis väljastaks tervishoiuteenuse osutaja infosüsteemiga integreeritult hoiatusi, soovitusi ning meeldetuletusi. Selline tugisüsteem põhineks olemasolevatele võõrkeelsetele reeglibaasidele, mis loodetakse tõlkida eesti keelde. (7 lk 2)

Siiski peab mainima, et käesoleva töö kirjutamise ajal on Eestis kliiniliste otsuste tugisüsteemide kasutus igapäevapraktikas alles algusjärgus. (7 lk 1)

1.1 Taust ja probleem

Kuigi otsustustoe süsteemide kasutuselevõttuga on Eestis algust tehtud, ollakse hetkel siiski veel enamuste puhul faasis, kus kohandatakse otsustustoe süsteeme endid. Näiteks tegeletakse nendes leiduvate ravijuhendite tõlkimisega (7 lk 13).

Kui süsteemid saavad riiklike vajadustega kohandatud, on järgmine samm vaja astuda süsteeme kasutada soovivate meditsiinasutuste tarkvarade haldajatel, kelle ülesandeks saab

otsustustoe süsteemid mingil moel olemasolevasse tööprotsessi sobitada ning luua tehniline võimekus otsustustoe süsteemidega suhtlemiseks.

Et erinevat otsustustoe infot pakkuvaid süsteeme võib olla mitu (näiteks ravimi koostoimete infosüsteem, kliiniliste ravijuhendite süsteem) ning ka nende arendajad on enamasti erinevad (nagu see on ka Eestis), peab meditsiinasutuse tarkvara haldaja looma võimekuse suhtlemaks nende kõigiga (looma erinevad liidesed).

Samas võivad otsustustoe pakkumisel vaid väljastpoolt tulevate teadmiste kasutamisel tekkida mõned sisulised puudujäägid, nagu näiteks:

- Kasutamata jääb asutuse enda aja jooksul kogunenud andmestik, millest oleks samuti võimalik seaduspärasusi leida. Kohalik andmestik omab kõige täpsemat infot asutuse tööprotsesside ning kasutajate harjumuste kohta.
- Välise otsustustoe kasutamisel puudub kasutajal enamasti võimalus kasutatavate reeglite haldamiseks ning nende tõendus põhise kontrollimiseks (süsteem küll tagastab mingit infot, kuid miks just sellist, seda ei tea).

Oleks mõttekas, kui lisaks väljastpoolt tulevale infole saaks meditsiinasutus soovi korral ka ise enda vajadustele vastavaid reegleid kirjeldada (või veelgi enam - neid olemasolevate andmete pealt leida). Samuti oleks otstarbekas vähendada ravitöö infosüsteemiga suhtlevate otsustustoe tugisüsteemide arvu ning kaasata ühe süsteemi juurde nii asutuse enda reeglid kui välistest süsteemidest tulev otsustustoe info. Seeläbi saaks ravitöö infosüsteemiga integreerida vaid ühe tugisüsteemi, mis enda reeglibaasi ning liidestatud süsteeme kasutades leiab info mitmest eri allikast.

Käesolevas töös pakutakse välja üks võimalik ravitöö infosüsteemiga integreeritava otsustustoe tugisüsteemi lahendus, mis on võimeline otsustustoe leidmiseks kasutama nii väliseid infoallikaid kui ka meditsiinasutuse enda kirjeldatud reegleid.

Esimeses etapis saab süsteem võimekuse kaasata välistest süsteemidest järgmiseid:

- Eestis toimiv SFINX - PHARAO ravimi koostoimete infot tagastav tugisüsteem;
- Eesti Tervise Infosüsteemi pakutav otsustustoe tugisüsteem, mis kasutab EBMeDS andmebaasi.

Samuti luuakse loodava tugisüsteemi juurde võimekus sisemiste reeglite haldamiseks ning säilitamiseks.

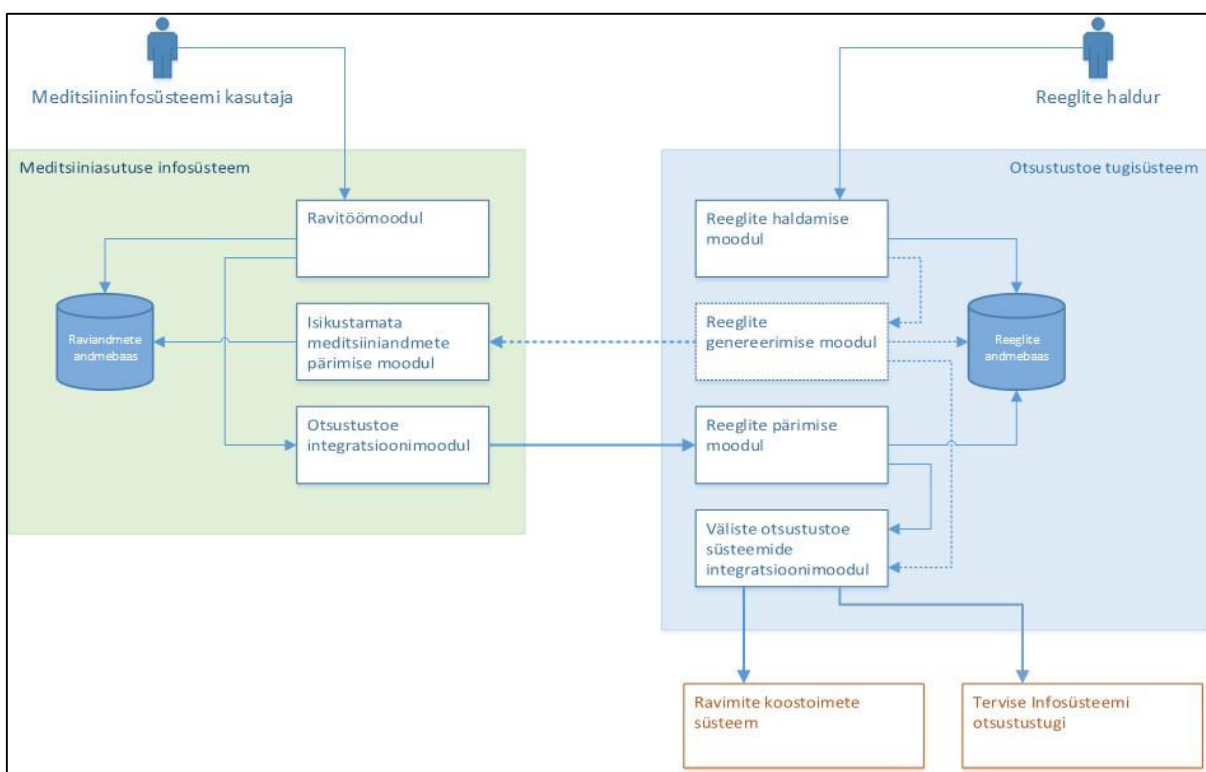
Töö tulemusi saavad kasutada meditsiinasutuste IT-üksused ning neile teenuseid pakkuvad tarkvaratootjad, kes soovivad ravitöö infosüsteemi juures mingil kujul otsustustoe teenuseid pakkuda.

1.2 Ülesande püstitus

Ravitööd toetava infosüsteemiga liidestatava reeglitel põhineva otsustustoe tugisüsteemi kavandamine on käesoleva magistritöö eesmärgiks. Esimeses (käesolevas magistritöös käsitletavas) etapis kavandatav süsteem peab võimaldama:

- meditsiinivaldkonna valikulisel andmemudelil põhineva reeglibaasi haldamist ning reeglite hoiustamist;
- etteantud raviandmetele põhinevate reeglite otsingut;
- otsustustoe tugisüsteemi integreerimist teiste otsustustoe süsteemidega.

Järgneva joonise paremal poolel on näidatud kavandatava otsustustoe tugisüsteemi komponendid ning nendevaheline andmevahetus. Punktiirjoonega tähistatud osad jäävad käesoleva magistritöö skoobist välja, kuid vajaduse ilmnemisel saab need tugisüsteemi arendamise järgmistes etappides lisada.



Joonis 1: Kavandatava tugisüsteemi komponendid ning interaktsioonid

1.3 Metoodika

Töös vaadeldakse esmalt maailmas olemasolevaid otsustustuge pakkuvaid süsteeme ning nende poolt pakutavaid teenuseid, et teha kindlaks, kas kavandatavat süsteemi juba mingil kujul olemas ei ole ning kas uue süsteemi loomine on põhjendatud.

Juhul kui olemasolevat sobivat süsteemi ei leita, vaadeldakse ühe konkreetse meditsiinasutuse infosüsteemi andmemudelit koos väliste andmeallikate andmemudelitega ning määratakse nende põhjal kavandatava tugisüsteemi reeglibaasi ning andmevahetuses edastatavate andmete struktuur. Loodav tugisüsteem püütakse luua võimalikult universaalne ning sõltumatu konkreetsetest ravitöö infosüsteemist, kuid kuna tugisüsteem peab õige tähendusega otsustustoe pakkumiseks toetama ravitöö infosüsteemis kasutatavaid loendeid ning klassifikaatoreid ning nendevahelisi seoseid, siis on konkreetse ravitöö infosüsteemiga liidestamiseks tehtavad kohandamised siiski vältimatud. Samas tulenevad loendite ja klassifikaatorite iseärasused enamasti riiklikest nõuetest (näiteks klassifitseeritakse Eesti teenuseid Haigekassa hinnakirjakoodide põhjal) ning ühe riigi piires peaks erinevusi olema vähe. Käesolevas töös kavandatav otsustustoe tugisüsteem kohandatakse ühe konkreetse ravitöö infosüsteemi omadustest lähtuvalt, mis peatselt mõningates Eesti meditsiinasutustes kasutusele võetakse.

Andmemudelite analüüsimisel saadud tulemuste põhjal määratakse kavandatava tugisüsteemi reeglibaasi ning selle haldamise mooduli detailsemad omadused ning nõuded. Teostatakse võimalike reeglibaasi juures kasutatavate järelusmootorite omaduste võrdlus ning valitakse sobivaim. Samuti kirjeldatakse andmevahetus ravitöö infosüsteemiga ning väliste otsustustoe tugisüsteemidega.

Viimaks vaadeldakse lühidalt ravitöö infosüsteemi ning kavandatud otsustustoe tugisüsteemi integreerimise võimalusi ravitöö infosüsteemi poolelt ning näidatakse mõned võimalikud kohad, kus ravitöö infosüsteem otsustustuge kasutada saaks.

1.4 Ülevaade tööst

Käesolev magistritöö koosneb viiest peatükist. Eraldi osad on kasutatud mõistete seletused, kokkuvõtte ning töö lisad.

Esimene peatükk on sissejuhatus, mis annab ülevaate töö temaatikast, eesmärkidest ning eesmärkideni jõudmise viisidest.

Teises peatükis antakse ülevaade olemasolevatest otsustustoe süsteemidest ning nende omadustest. Jõutakse järelduseni, kas planeeritava süsteemi loomine on põhjendatud.

Kolmandas peatükis kirjeldatakse ühe liidestatava ravitöö infosüsteemi isikustamata andmemudel, mida kavandatakse otsustustoe tugisüsteem peab toetama. Samuti vaadeldakse Eestis väljatöötatud ravijuhendite ning Tervise Infosüsteemi ning ravimi koostoimete andmebaasi poolt pakutava otsustustoe andmemudeleid, mida kavandatakse otsustustoe tugisüsteem samuti peaks toetama.

Neljandas peatükis kirjeldatakse kolmandas peatükis toodu põhjal kavandatava otsustustoe tugisüsteemi reeglibaasi ning andmevahetuse käigus edastatavate andmete struktuur. Esitatakse tugisüsteemi alammodulite kirjeldus. Kirjeldatakse tugisüsteemile esitatavad nõuded. Teostatakse kasutatava järeldusmootori valik. Antakse soovitusi süsteemi ning selle kasutusvõimaluste võimalikeks täiendavateks arenguteks.

Viiendas peatükis kirjeldatakse lühidalt kavandatava otsustustoe tugisüsteemi ning ühe konkreetse ravitöö infosüsteemi integreerimise võimalusi.

Kokkuvõtte osas kirjeldatakse töös saadud tulemusi ning esinenud probleeme.

2. Olemasolevate otsustustoe tugisüsteemide kirjeldus

Kuigi praegusel ajastul on informatsioon lihtsamini kättesaadavam kui kunagi varem, on kasutajate ette kerkinud uus probleem - leida hiiglaslike andmemahtude seast üles endale vajalik ning osata infot eesmärgipäraselt ja süsteemselt kasutada. Siin ei ole erandiks ka meditsiinivaldkond, kus meditsiinipersonalil on keeruline end pidevalt uuenevate ravijuhiste, uute ravimite ning ravivõimalustega kursis hoida.

Mainitud probleemi leevendamisel tulevad appi kliiniliste otsuste tugisüsteemid, mille ülesandeks on järeldusmootori abil laialdast meditsiinilise teadmiste baasi ning patsiendi andmeid analüüsides genereerida erialaseid nõuandeid teadmiste ning järelduste vormis. (7 lk 3)

Kliiniliste otsustustoe tugisüsteemide eesmärkideks on:

- ravivigu ning kõrvaltoimeid ennetades suurendada patsientide ohutust;
- suurendada ravijuhiste ning ajakohase informatsiooni kasutamist, vähendada arstil ravijuhise koostamiseks kuluvat aega ning seeläbi parandada ravi kvaliteeti;
- suurendada efektiivsust, vähendades kulusid ning ebavajalikke dubleerivaid analüüse ning uuringuid. (7 lk 3)

Tallinna Tehnikaülikooli Kardiovaskulaarse meditsiini instituudi E-Meditsiini Laboratoorium on 2014 aastal teinud ülevaate kliiniliste otsuste tugisüsteemidest, kus valiku aluseks olid järgmised aspektid:

- integreerimise võimalus/lihtsus;
- lahendused, mis oleksid suunatud eri- ja /või perearstidele;
- lahendused, mis suurendaksid patsientide kaasatust (7 lk 4).

Järgnevalt on välja toodud mainitud valikus kirjeldatud süsteemid ning võimalused nende integreerimiseks ravitöö infosüsteemiga.

2.1 Vaadeldud kliiniliste otsuste tugisüsteemid

Süsteem	Kirjeldus	Kasutamise ning integreerimise võimalused
Micromedex Clinical Knowledge Suite (USA)	<p>Tõendus põhine kliiniliste otsuste tugisüsteem, mis pakub kliinilisi lahendusi haiguse ja seisundi hindamiseks, ravimite määramiseks, ravimite alast informatsiooni ning toksikoloogia alaseid nõuandeid (7 lk 5).</p> <p>On kasutusel enam kui 5000 meditsiinasutuses enam kui 83 riigis. (8)</p> <p>Tasuline.</p>	<p>Omab veebirakendust ning eraldi rakendust nutiseadmetes kasutamiseks.</p> <p>Võimaldab HL7 standardile vastavat andmevahetust ning terve või osalise andmekogu allalaadimist ning kasutamist.</p> <p>On integreeritud enam kui 2000 patsiendi meditsiiniandmeid sisaldava süsteemiga USA-s. (8)</p>
Archimedes IndiGO (USA)	<p>Kalkuleerib ning hindab patsiendi meditsiinisündmuste põhjal riske ning pakub soovitusi nende vähendamiseks.</p> <p>Sisendiks võetakse patsiendi meditsiinilised andmed (analüüside tulemused, diagnoosid, kasutatavad ravimid) ning riskifaktorid (suitsetamine, pärilikud haigused).</p> <p>Tasuline.</p>	<p>Sisendandmed laetakse üles IndiGO süsteemi, mis tagastab seejärel tulemuse (soovituste ning graafikute kujul).</p> <p>Eraisikutele pakutakse tasuta rakendust nutiseadmes kasutamiseks, mis hoiatab neid võimalike terviseriskide eest. (9)</p>
Elsevier ClinicalKey (USA)	<p>Veebipõhine tõendus põhine otsingusüsteem (7 lk 4).</p> <p>Otsingus kasutatakse Elseviere-i andmepanka, kus on palju erialaseid andmeallikaid- üle 1000 raamatu, üle 600 ajakirja, üle 17000 video, üle 300</p>	<p>Kasutajad saavad kasutada veebipõhist meditsiinitöötajatele kohandatud otsingumootorit.</p> <p>Võimalik on otsingusüsteem siduda kohaliku raviinfosüsteemiga. (10)</p>

	<p>protseduure kirjeldava video, üle 2 000 000 pildi, üle 2900 ravimi kirjelduse, üle 15 000 patsiendi infolehe jne. (10)</p> <p>Tasuline.</p>	
Map of Medicine (UK)	<p>Kliiniliste otsuste tugisüsteem, mis koos NHS Choices-ga moodustab nii arstidele kui ka patsientidele suunatud terviseinfo keskkonna kajastades nii tervise seisundeid, tervislikke eluviise, sotsiaalhoolekannet kui ka koondades erinevad teenusepakkujad (7 lk 5).</p> <p>Võimaldab lisaks sisemisele andmepangale ka kasutaja individuaalset reeglite ning soovitusete kirjeldamist. (11)</p> <p>Tasuline.</p>	On mõeldud kohaliku raviinfosüsteemiga sidumiseks.
EBMeDS (Soome)	<p>Tõendus põhine kliiniliste otsuste tugisüsteem, mis väljastab meeldetuletusi, linke diagnoosiga seotud ravijuhistele ning ravimitega seotud näidustusi, vastunäidustusi ja koosmõjusid. 70% väljastatavatest tulemustest on seotud ravimitega. Sisaldab ligikaudu 600 rahvusvahelist algoritmi. On hetkel kasutusel 16 riigis. (7 lk 6)</p> <p>Tasuline</p>	<p>On mõeldud kohaliku raviinfosüsteemiga sidumiseks. Andmevahetus toimub XML formaadis. (12)</p> <p>Otsuste päring saadetakse kohalikus raviinfosüsteemist andes kaasa patsiendi meditsiinilised andmed. EBMeDS tagastab tulemuse XML või HTML formaadis, mille kasutajaliideses esitamise teostab raviinfosüsteem. (7 lk 6)</p>
Sfinx Pharao	<p>Tõendus põhine ravimite infole keskendunud otsustustoe süsteem.</p>	Võimaldab integreerimist olemasoleva raviinfosüsteemi

(Soome)	Sisaldab informatsiooni ligikaudu 18 000 ravimi ning ravimite võimalike koosmõjude kohta. Sisendiks võetakse olemasolevate või uute ravimite info ning tagastatakse tulemus võimalike koosmõjude ning ravisoovituste kohta koos viitega otsuse põhjustanud allikale. (13)	juurde, mis pärib Sfinx-ilt otsustuge ning saab vastuseks tulemuse, mille esitab kasutajale sobival moel ning ajal.
---------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------

Tabel 1: Olemasolevad kliiniliste otsuste tugisüsteemid

2.2 Järeldused

Kuna enamused süsteemid on tasulised, on süsteemide kõrge hind üheks esmaseks takistuseks nende rakendamisel (7 lk 3).

Olemasolevad otsustustoe tugisüsteemid on väljatöötatud kõrge elatustasemega riikides. Kuigi konkreetset hinnapoliitikat materjalides ei avalikustata, võib siiski eeldada, et ka nende hinnatase on võrdlemisi kõrge. Eesti tervishoid jääb IT-arenduste rahastamisel oluliselt maha kiirelt arenevatest riikidest ning ka Euroopa komisjoni soovitatud tasemest (5 lk 1). Seega võib arvata, et enamuse keerukamate eelnevalt kirjeldatud tasuliste vahendite kasutuselevõtt käib eraldiseisvatele meditsiinasutustele üle jõu.

Siiski on Eesti riiklikul tasemel alustanud EBMeDS andmebaasi lokaliseerimisega. EBMeDS andmebaasi kasutamist planeeritakse läbi Tervise Infosüsteemi ning seeläbi oleks see meditsiinasutustele arvatavasti tasuta.

Mõningad meditsiinasutused on praeguseks hetkeks liidestunud Eestis tegutseva teenusepakkujaga, mis võimaldab pärida ravimi koostoimete infot Sfinx-Pharao andmebaasist. Seega erinevate süsteemidega liidestumisega on algust tehtud.

Siiski ei võimalda mainitud lahendused raviasutusel isiklike reeglite defineerimist. Samuti tuleb raviasutuse infosüsteemi integratsiooniks otsustustoe süsteemidega luua eraldiseisvad ning erinevatel andmemudelitel põhinevad liidesed.

Kuna meditsiiniline otsustustugi on kiirelt arenev valdkond, siis võib tulevikus lisanduda teisigi otsustustoe andmeallikaid. Nende kasutuselevõtt vajaks samuti uusi liideseid. Samas otsustustuge vajavad tööprotsessid on raviasutuse infosüsteemides tõenäoliselt samad.

Eelkirjeldatule tuginedes järeltab käesoleva töö autor, et vähemalt Eesti olusid arvestades on mõttekas luua uus otsustustoe tugisüsteem - integratsiooniplatvorm, mis võimaldab:

- liidestamist erinevate väliste otsustustoe süsteemidega;
- kohaliku reeglibaasi omamist ning haldamist;
- liidestamist ravitöö infosüsteemiga;
- välistele andmeallikate ning sisemisele reeglibaasile tugineva otsustustoe info pärimist ühe ühtse liidese kaudu.

Kirjeldatud lahendus võimaldab meditsiinasutuse infosüsteemi liidestada vaid loodava otsustustoe tugisüsteemiga ning otsustustoe päringud tööprotsessidesse lülitada. Muutused otsustustoe andmeallikate kasutuses (välised otsustustoe süsteemid, kohalik reeglibaas) ei too enam kaasa vajadust ravitöö infosüsteemi muutmiseks.

Kohaliku reeglibaasi omamine võimaldab kohalike reeglite kirjeldamist. Samuti on tulevikus võimalik andmekaeve meetodeid kasutades ravitöö infosüsteemi andmetele põhinevate reeglite loomine ning kasutamine.

Kirjeldatud viisil on võimalik ravitöö infosüsteemile otsustustoe pakkumisel rakendada nii üldtunnustatud teadmisi kui kohalikke vajadusi ning lülitada kogu saadaolev otsustustoe info ravitöö infosüsteemi poolt toetatavate tööprotsesside juurde läbi ühe ühtse liidese.

3. Väliste andmeallikate andmemudelite kirjeldus

Kavandatav süsteem peab võimaldama lisaks sisemisele reeglibaasile ka väliste andmeallikate kasutamist.

Käesoleva töö raames on seatud eesmärk luua võimalus liidestada kavandatav otsustustoe tugisüsteemi kahe välise andmeallikaga:

- Eesti Tervise Infosüsteemi otsustustoe allsüsteemiga;
- Sfinx-Pharao ravimi koostoimete süsteemiga.

Samuti on otstarbekas, et kavandatava tugisüsteemi sisemine reeglibaas võimaldaks sinna lisada Eestis väljatöötatud ravijuhendite andmeid. Ning lisaks eeltoodule peab kavandatav otsustustoe tugisüsteem ühilduma meditsiinasutuse ravitöö infosüsteemiga.

Eeltoodust tulenevalt on vajalik vaadelda seotud süsteemide andmemudeleid, mille põhjal saab defineerida kavandatava otsustustoe tugisüsteemi andmemudeli.

Järgnevalt vaadeldaksegi ravitöö infosüsteemi, Eesti Tervise Infosüsteemi otsustustoe allsüsteemi, Sfinx-Pharao ravimi koostoimete süsteemi ning Eesti ravijuhendite loogilisi andmemudeleid, mis saavad aluseks kavandatava otsustustoe tugisüsteemi ning selle sisemise reeglibaasi andmemudeli defineerimisel.

3.1 Ravitöö infosüsteemi valikulise andmemudeli kirjeldus

Kavandatav otsustustoe tugisüsteem peab ravitöö infosüsteemiga ühilduma kahes erinevas rollis:

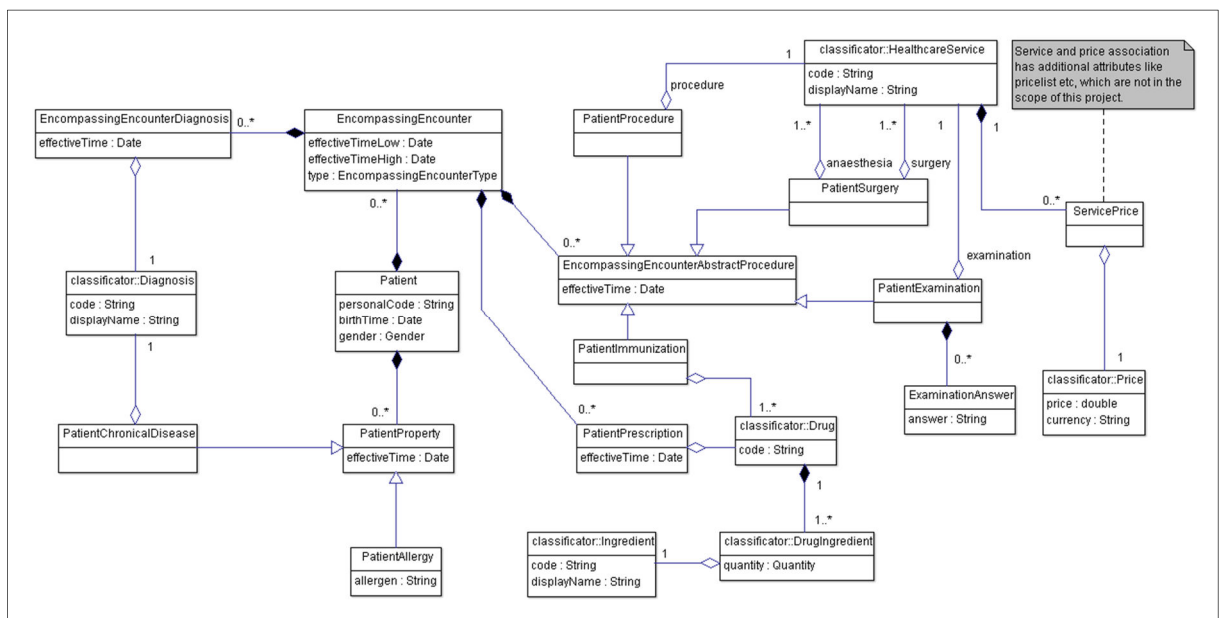
- võimaldama ravitöö infosüsteemi andmemudeliga ühilduvate reeglite kirjeldamist;
- võimaldama ravitöö infosüsteemi andmemudeliga ühilduva otsustustoe info pärimist.

Seega on ravitöö infosüsteemi andmemudel oluline nii otsustustoe tugisüsteemi reeglibaasi kui ka interaktsiooniliidese kavandamise osas.

Järgnevalt on kirjeldatud ühe Eestis kasutatava ravitöö infosüsteemi valikuline ning isikustamata andmemudel. Mudeli koostamisel vaatles töö autor ühe konkreetse ravitöö infosüsteemi relatsioonilise andmebaasi arhitektuuri, mille arendamise juures autor on

püsivalt osalenud. Mudelis on välja toodud kavandatava otsustustoe tugisüsteemi juures võimalikud tähtsust omavad (ravitoiminguid ning patsiendi omadusi kirjeldavad) andmed ning jäetud kõrvale ülejäänud (näiteks aruandluse ning detailse raviarvelduse jaoks kasutatav andmestik), mis käesoleva töö skoobis planeeritava otsustustoe pakkumisel tähtsust ei oma. Andmemudelit on otsustustoe tugisüsteemi vajadustest lähtuvalt mõneti üldistatud ning elementide ja parameetrite nimetamisel on parema arusaadavuse huvides kasutatud osaliselt HL7 standardites kasutatavaid nimetusi.

Andmemudel on küll koostatud ühe konkreetse süsteemi põhjal, kuid kuna samalaadset ravi pakkuvate meditsiinasutuste tööprotsessid on üldjuhul sarnased, siis võib mudelit vaadelda kui üldist samalaadset ravitööd toetava infosüsteemi valikulist andmemudelit.



Joonis 2: Ravitöö infosüsteemi valikuline isikustamata andmemudel

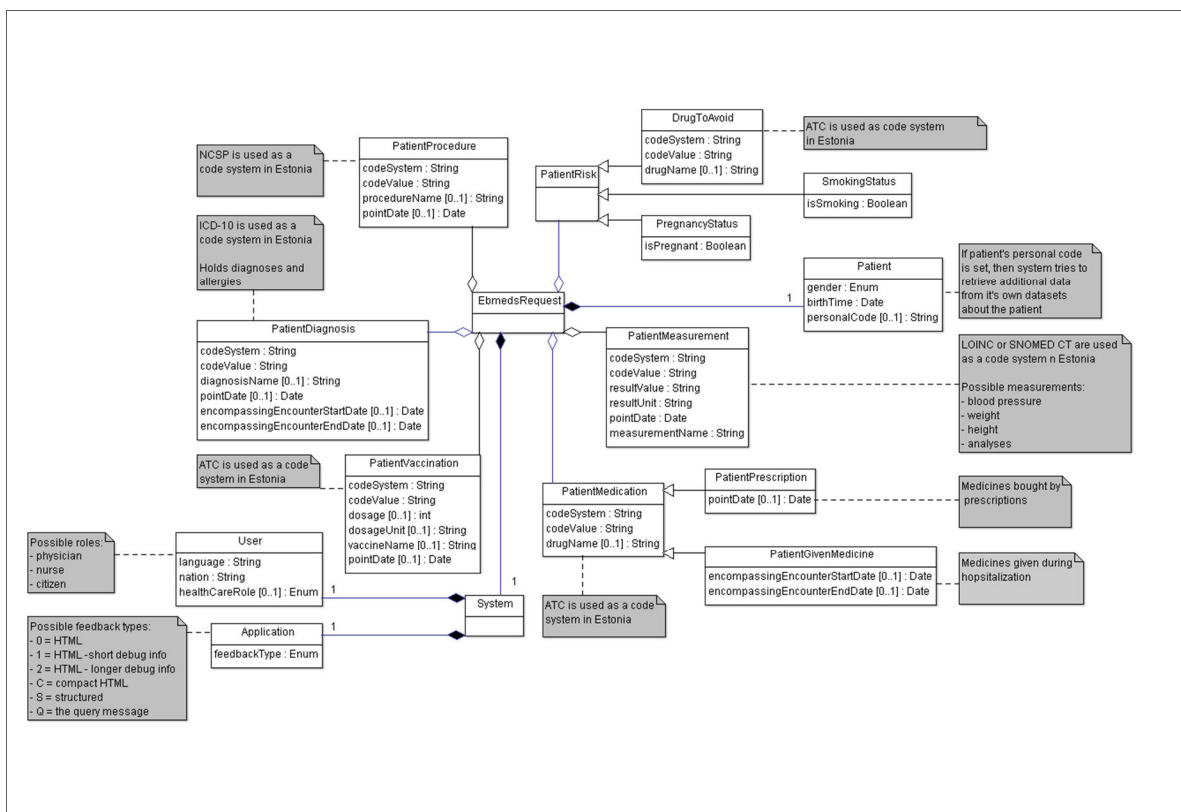
Nagu jooniselt võib näha, on vaadeldud ravitöö infosüsteemi andmemudel patsiendipõhine. Patsiendiga on seotud tema üldised omadused (näiteks allergiad, kroonilised haigused, tarvitavad ravimid läbi „PatientProperty“ elemendi) ning konkreetsed haigusjuhtumid („EncompassingEncounter“ element). Enamus konkreetseid ravitoiminguid (protseduurid, lõikused, uuringud, analüüsid) on patsiendiga seotud läbi haigusjuhtumi.

3.2 Eesti Tervise Infosüsteemi otsustustoe allsüsteemi andmemudeli kirjeldus

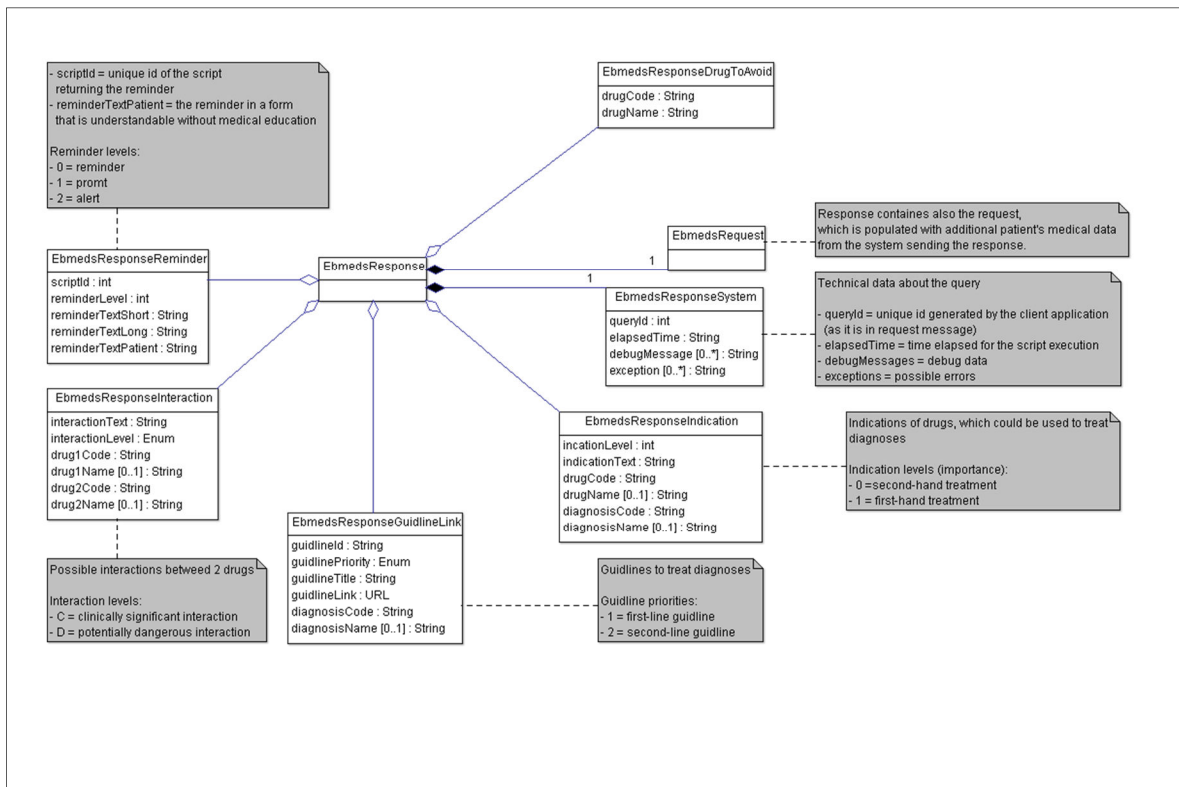
Tervise Infosüsteem kasutab otsustustoe pakkumisel Soomes väljatöötatud EBMeDS andmebaasi, mis väljastab meeldetuletusi, diagnoosiga seotud ravijuhiste viiteid, ravimitega seotud näidustusi, vastunäidustusi ning koosmõjusid.

Otsustustoe päringus antakse kaasa andmed patsiendi ning meditsiinitoimingute kohta. Tervise Infosüsteemi otsustustoe süsteem täiendab olemasolul patsiendi andmeid isiklikust andmekogust ning tagastab tulemuse, kus on võimalikud soovitusel, hoiatused, ravimite näidustused jne.

Järgnevalt on välja toodud Tervise Infosüsteemi otsustustoe päringu ning päringu vastuse andmemudelid. Mudelid on käesoleva töö autori poolt koostatud, kasutades Eesti Tervise infosüsteemi EBMeDS liidese spetsifikatsiooni (14). Elementide ning parameetrite nimetamisel on töö autor parema arusaadavuse huvides kasutanud samatähendusliku info puhul ravitöö infosüsteemi andmemudelid (joonis 2) kasutatud nimetusi.



Joonis 3: Tervise Infosüsteemi otsustustoe päringu andmemudel



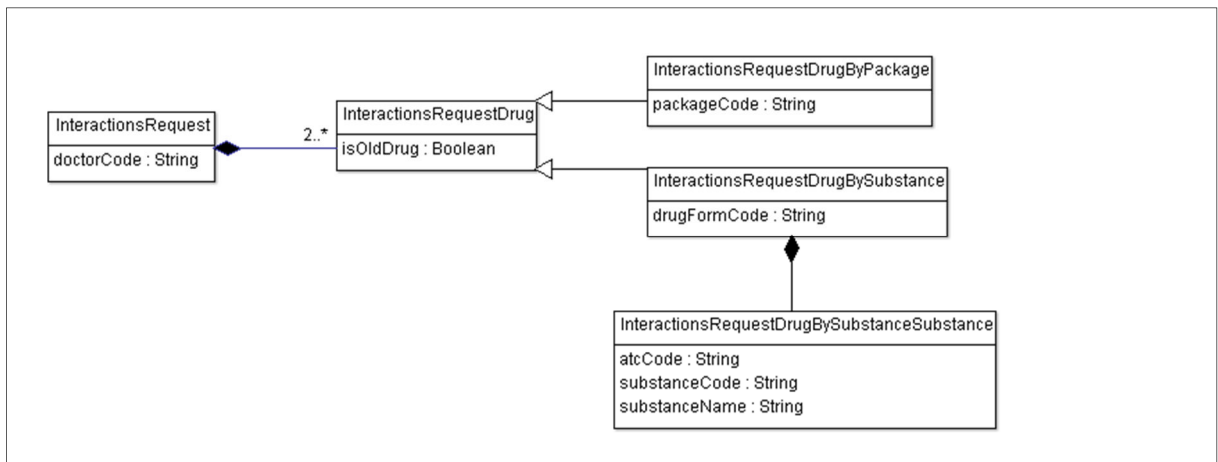
Joonis 4: Tervise Infosüsteemi otsustustoe päringu vastuse andmemudel

Nagu joonistelt 3 ning 4 võib näha, on Tervise Infosüsteemi otsustustoe päring ning vastus patsiendipõhised. Päringus edastatakse patsiendi üldised andmed ning hetkel aktuaalsed raviandmed. Tervise Infosüsteemi otsustustoe süsteem vastab edastatud andmetele tuginevate ravimi koostoimete ning soovitusetega. Tervise Infosüsteem on võimeline kasutama ka enda patsiendi andmestikku olemasolevate andmete leidmiseks, kuid see ei ole kohustuslik. Nimetatud võimaluse kasutamiseks tuleb päringus edastada ka patsiendi isikukood. (14)

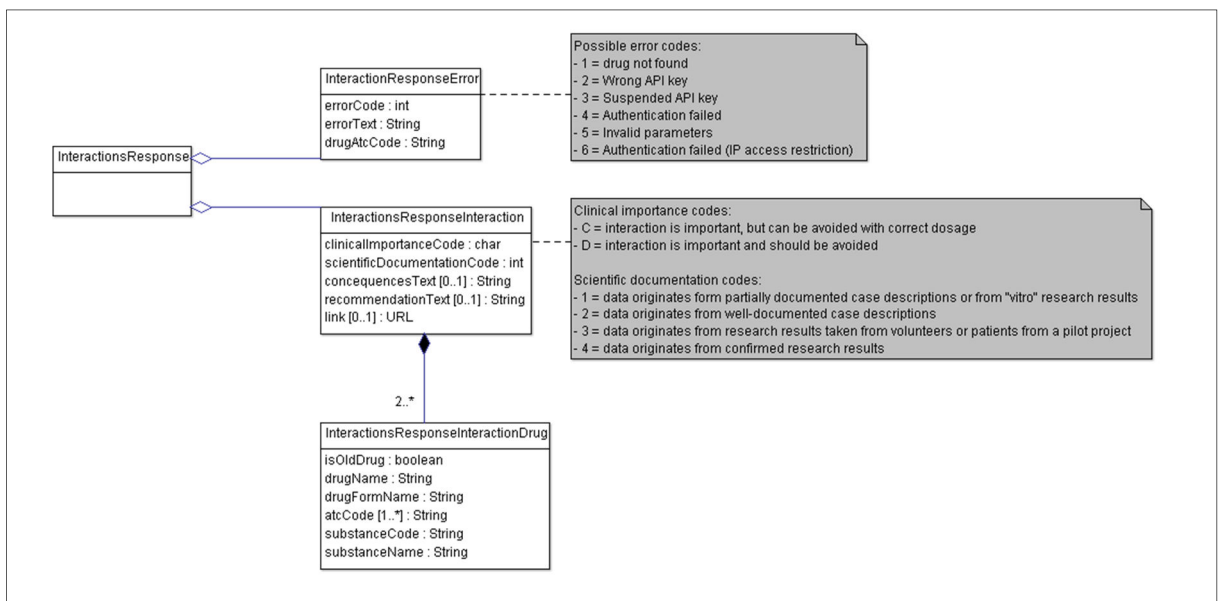
3.3 Ravimi koostoimete süsteemi andmemudeli kirjeldus

Ravimi koostoimete süsteem tugineb Sfinx-Pharao andmebaasile. Päringus antakse kaasa nimekiri kasutatavatest ravimitest (sealjuures määratakse, kas ravim on juba kasutusel või plaanitakse kasutusele võtta). Vastuses tagastab koostoimete süsteem nimekirja ravimite võimalikest koosmõjudest, määrates iga koosmõju puhul selle kriitilisuse ning leitud koostoime aluseks olevad andmeallikad.

Järgnevalt on välja toodud ravimi koostoimete päringu ning päringu vastuse andmemudelid. Mudelid on käesoleva töö autori poolt koostatud, kasutades Eestis Sfinx-Pharao koostoimete süsteemi haldava Celsius Healthcare ning nende partnerite poolt loodud liidese spetsifikatsiooni (15).



Joonis 5: Ravimi koostoimete päringu andmemudel



Joonis 6: Ravimi koostoimete päringu vastuse andmemudel

Nagu eelnevalt kahelt jooniselt võib näha, on ravimi koostoimete päring ning vastus võrdlemisi lihtsa struktuuriga - edastatakse manustatavate ja kavandatavate ravimite andmed (joonisel 5 element „InteractionsRequestDrug“) ning saadakse vastuses nimekiri võimalikest ravimite koosmõjudest (joonisel 6 element „InteractionResponseInteraction“), mille juures on ka info koosmõju põhjustatavates konkreetsetes ravimitest (joonisel 6 element „InteractionsResponseInteraction-Drug“). Koosmõju põhjustavate ravimite info vastuses on vajalik, kuna koosmõjusid võib olla mitu ning neid võib esineda vaid osade päringus edastatud ravimite puhul.

Ka Eesti Tervise Infosüsteemi otsustustoe päringu vastuses on koht ravimi koostoimetele (joonis 4, element „EbmedsResponseInteraction“). Seega võib tekkida küsimus, kumba on otstarbekam kasutada.

Kuna Sfinx-Pharao andmekogu on pühendunud vaid ravimi koostoitete infole, siis on mõlema andmeallika olemasolul pigem põhjust kasutada ravimi koostoitete päringu tulemust. Kuid kuna kavandatav otsustustoe tugisüsteem võib olla liidestatud ka vaid ühe välise andmekoguga, peaks olema võimalik kasutada mõlema andmeallika ravimi koostoitete infot. Kuna mõlemas päringu vastuses kasutatavad väärtused kuuluvad samasse väärtuste süsteemi, siis ei tohiks erinevatest andmekogudest pärinevate koostoitete tõlgendamisel ühilduvusprobleeme tekkida.

3.4 Eesti ravijuhendite andmebaasi andmemudeli kirjeldus

Ravijuhend on dokument, mis annab soovitusi tervist mõjutavate tegevuste kohta. Selles antakse tervishoiutöötajatele tõenduspõhiseid juhiseid haiguste diagnoosimiseks ning ravimiseks. Samuti võib see sisaldada soovitusi haiguste ennetuseks, patsientide koolitamiseks jms. Ravijuhendis toodud teave aitab teha valikuid eri sekkumisviiside vahel, mis mõjutavad tervist, ravikvaliteeti ning tervishoiuressursside kasutamist. Ravijuhendi ja õpiku peamine erinevus on, et ravijuhendis keskendutakse eeskätt haiguse diagnoosimise ja raviga seotud tegevustele, õpikus aga kirjeldatakse põhjalikult haiguse kõiki aspekte. (16)

Eeltoodule tuginedes võib öelda, et kuna väljatöötatud ravijuhendid põhinevad just haiguste diagnoosimisele ning diagnoosimine on suuresti patsiendi erinevate meditsiiniliste mõõtmistulemuste analüüs, siis on mõttekas loodavas süsteemis ka ravijuhendite kaasamisega arvestada.

Järgnevalt on välja toodud ühe ravijuhendi ("Eesti 2. tüüpi diabeedi juhend 2008") diagnoosimise sektsiooni osaline andmestik.

Diabeeti diagnoositakse, kui patsiendil on:

- glükoos paastuplasmas $\geq 7,0$ mmol/l või
- hüperglükeemia sümptomaatika ja juhuslik plasmaglükoos $\geq 11,1$ mmol/l või
- glükoositaluvuse proovis 2 t pärast 75 g glükoosi manustamist glükoos plasmal $\geq 11,1$ mmol/l.

(17)

Antud haiguse diagnoosimisel soovitatakse seega osaliselt tugineda diagnooside ("hüperglükeemia") ning analüüsile tulemuste analüüsile. Selliste väärtuste arvestamine on võimalik käesolevas töös kavandatava tugisüsteemi juures realiseerida.

Lisaks on ravijuhendis aga ka määranguid, mida vajalike andmete puudumisel reegliteks muuta on võrdlemisi keeruline. Üks näide on erinevaid analüüside võtmise aegu määravad tingimused:

- glükoos 2 t pärast sööki $\geq 7,5$ mmol/l;
- glükoositaluvuse proovis 2 t pärast 75 g glükoosi manustamist glükoos plasmas $\geq 11,1$ mmol/l.

Siiski on enamuste ravijuhendite puhul võimalik määrata teatud komplekt diagnoosimise tingimusi, mis on reeglite kujul võimalik tugisüsteemile arusaadavaks muuta. Reeglitele vastava juhtumi korral saab tugisüsteem väljastada viite detailsele ravijuhendile, mille põhjal saab ekspertteadmistega isik (arst) juba täiendavalt analüüsida ravijuhendi tingimusi ning otsustada diagnoosi põhjendatuse üle. Tugisüsteemi ülesanne ei saagi olla otsuste vastuvõtmine, vaid kasutaja abistamine.

Eeltoodule ning publitseeritud ravijuhendite andmetikele tuginedes on järgnevas tabelis välja toodud parameetrid, mille määramist kavandatav tugisüsteem peaks võimaldama.

Parameeter	Andmetüüp	Parameetri ühik
Analüüs	LOINC klassifikaatori kood	
Analüüsi tulemus	Reaalarv	Analüüsi ühikute loend
Diagnoos	RHK-10 klassifikaatori kood	
Patsiendi vanus	Naturaalarv	Ajaühikute loend
Patsiendi pikkus	Naturaalarv	Pikkusühikute loend
Patsiendi kehakaal	Naturaalarv	Kaaluühikute loend
Patsiendi sugu	Sugu	

Tabel 2: Parameetrid Eesti ravijuhendite kasutamiseks kavandatavas otsustustoe tugisüsteemis

Nimetatud parameetrite väärtuste määramisel on võimalik luua esmased seosed ravijuhenditega, mis sisaldavad juba detailsemaid andmeid konkreetsete haiguste diagnoosimiseks.

3.5 Järeldused

Käesolevas peatükis kirjeldati nelja süsteemi, mille andemudeleid kavandatav otsustustoe tugisüsteem toetama peab. Nendeks olid:

- Eesti Tervise Infosüsteemi otsustustoe allsüsteem (EBMeDS);
- ravimi koostoimete süsteem (SFINX-PHARAO);
- Eesti ravijuhendite andmebaas;
- meditsiinasutuse ravitöö infosüsteem.

Kirjeldatud süsteemide kasutatav andmestik on klassifitseerimise osas küllaltki sarnane - enamasti kasutatakse samatähendusliku info määratlemiseks samu klassifikaatoreid (näiteks diagnooside puhul rahvusvaheline ICD-10 süsteem). Samas on kasutatavad andmestikud erinevad ning kui oleks vajalik defineerida ühine komplekt andmeid mis kõikide integreeritavate süsteemide vajadusi rahuldaks, siis läheks andmestik võrdlemisi suureks.

Kavandatav otsustustoe tugisüsteem peab aga olema võimeline erinevates süsteemides paikneva ning erinevalt struktureeritud informatsiooni mõistmiseks. Seega on vajalik erinevad andmemudelid mingil viisil ühildada. Integreeritavate süsteemide andmemudelite ühildamise kava kirjeldatakse täpsemalt käesoleva töö peatükis 4.3.1.

Kui tulevikus tekib vajadus liidestada kavandatava otsustustoe tugisüsteemiga mõni süsteem mis juba kasutuselolevat infot erinevalt klassifitseerib (näiteks võib see diagnooside määratlemiseks kasutada ICD-10 asemel CNOMED CT süsteemi), siis tekib vajadus ka klassifikatsioonide teisendamiseks. Selliseid võimalikke arenguid on otstarbekas arvestada ka käesoleva otsustustoe tugisüsteemi kavandamisel.

4. Otsustustoe tugisüsteemi kirjeldus

Kavandatava otsustustoe tugisüsteemi ülesandeks on erinevaid andmeallikaid kasutades pakkuda ravitöö infosüsteemile otsustustoe teenust. Andmeallikateks on nii süsteemi sisemine reeglibaas kui ka välised otsustustoe infot pakuvad süsteemid, mis käesoleva töö raames kavandatava tugisüsteemiga liidestatakse.

Sealjuures peab kavandatav tugisüsteem toime tulema ka erinevates andmeallikates erinevalt klassifitseeritud informatsiooniga, oskama seda õigesti tõlgendada ning päri va süsteemi jaoks vajalikule kujule viia.

Käesolevas peatükis kirjeldatakse kavandatavat otsustustoe tugisüsteemi täpsemalt.

Peatükis 4.1 antakse ülevaade kavandatava süsteemi arhitektuurist ning komponentidest.

Peatükis 4.2 kirjeldatakse sisemise reeglibaasi omadusi ning reeglibaasi haldusliidest.

Peatükis 4.3 vaadeldakse süsteemi erinevaid integratsioone ning sõnumivahetust integratsioonide käigus.

Peatükis 4.4 pühendatakse andmemudelitele - kirjeldatakse integreeritavate süsteemide andmemudelite ühildamise kava, sisemise reeglibaasi andmemudel ning kavandatava tugisüsteemi ning ravitöö infosüsteemi integratsioonisõnumite andmemudelid.

Peatükis 4.5 kirjeldatakse sisemise reeglibaasi kasutamiseks vajaliku järelusmootori nõudeid, teostatakse järelusmootori valik ning kirjeldatakse järelusmootori rakendamise kava.

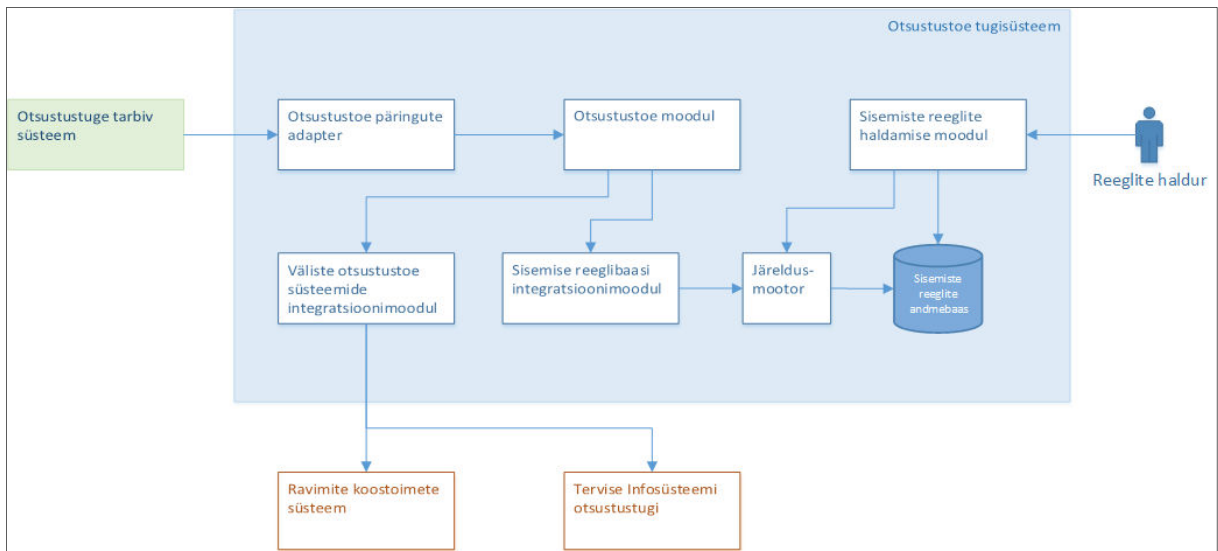
Peatükis 4.6 kirjeldatakse mõningaid võimalusi käesolevas töös kavandatud tugisüsteemi edasiarendamiseks.

Käesolevast magistriltööst on jäetud välja konfidentsiaalsete isikuandmete lekke ohu analüüs, kuna kavandatav otsustustoe tugisüsteem isikuandmeid enda andmebaasis ei hoiusta ning tugisüsteemi kasutatav andmestik on isikustamata. Süsteem on esimeses rakenduskohas kavandatud töötama ravitöö infosüsteemiga samas keskkonnas (meditsiiniastutuse sisevõrgus) ning andmete transporti üle avaliku võrgu ei toimu.

4.1 Süsteemi arhitektuur

Otsustustoe tugisüsteemi arhitektuuri iseloomustab järgnev joonis 7, kus on näidatud tugisüsteemi põhilised komponendid ning interaktsioonid. Käesolevas peatükis kirjeldatakse neid komponente täpsemalt.

Joonisel 7 kujutatud arhitektuuriline skeem on koostatud eelnevates peatükkides kirjeldatud süsteemile esitatavate nõuete põhjal. Võrreldes joonisel 1 toodud üldisema süsteemide skeemiga on keskendunud detailsemalt kavandatavale tugisüsteemile ning ravitöö infosüsteem näidatud vaid pärija rollis. Sisemise reeglibaasi kasutamiseks on lisatud järelusmootori komponent ning kavandatava tugisüsteemi sisemised otsustustoe päringud on lahendatud SOA põhimõtetest lähtuvalt – iga andmekogu (sisemine reeglibaas, välised süsteemid) ees on spetsiifiline integratsioonimoodul, mis pakub antud andmekoguga suhtlemise teenust.



Joonis 7: Otsustustoe tugisüsteemi arhitektuur

Otsustustoe päringute adapteri ülesandeks on võtta vastu otsustustoe päringuid, viia neis paiknev info süsteemi jaoks sobivale kujule ning edastada otsustustoe moodulile. Otsustustoe päringuid teostab antud töö kontekstis ravitöö infosüsteem ning päringus edastatava info hulka kuuluvad päringu tüüp (mis liiki infot soovitakse) ning faktid, mida päringu töötlemisel on vaja arvestada.

Sisemise reeglibaasi ning reeglite haldamise moodulite ülesanne on võimaldada vajaliku andmemudeliga sisemiste reeglite loomist ning haldamist. Sisemist reeglibaasi kirjeldatakse täpsemalt peatükis 4.2.

Välise otsustustoe süsteemide integratsioonimoodul võimaldab otsustustoe moodulil välise süsteemidega suhelda ning sealt tulevat infot mõista. Integratsioonimooduli ülesanne on saadud info sobivale kujule teisendada. Integratsiooni kirjeldatakse täpsemalt peatükis 4.3.

Järeldusmootori ning temaga seotud integratsioonimooduli ülesanne on sisemise reeglibaasi reeglite põhjal järelduste tegemine ning järeldamise põhjal saadud info sobivale kujule teisendamine. Järeldusmootorit kirjeldatakse täpsemalt peatükis 4.5.

Lisaks kirjeldatakse peatükis 4.4 moodulite juures kasutatavaid andmemudeleid ning nende ühildamise kava.

4.2 Sisemise reeglibaasi kirjeldus

Sisemise reeglibaasi kavandamisel on seatud eesmärk, et see võimaldaks enimkasutatavatel ravitoimingutel ning patsiendi omadustel põhinevate reeglite kirjeldamist.

Lähtudes Eesti ravijuhendite andmebaasi parameetritest (kirjeldatud peatükis 3.4 tabelis 2) ning liidestatava ravitöö infosüsteemi isikustamata andmemudelist (kirjeldatud peatükis 3.1 joonisel 2), valiti sisemises reeglibaasis toetatavate parameetrite hulka järgmises tabelis kirjeldatud parameetrid.

Parameeter	Kirjeldus
Ravitoiming	Kirjeldatakse toimingut identifitseeriva tunnuse määramisega. Ravitoiming võib olla näiteks analüüs, protseduur, operatsioon.
Diagnoos	Diagnooside vahemik (diagnooside klassifikatsioon on hierarhiline). Kirjeldatakse ülemise ning alumise piirväärtuse määramisega.
Analüüsi tulemus	Analüüsi tulemuste vahemik. Kirjeldatakse ülemise ning alumise piirväärtuse ning analüüsi ühiku määramisega.
Patsiendi vanus	Patsiendi vanusevahemik. Kirjeldatakse ülemise ning alumise piirväärtuse määramisega.
Patsiendi pikkus	Patsiendi pikkuse vahemik. Kirjeldatakse ülemise ning alumise

	piirväärtuse määramisega.
Patsiendi kehakaal	Patsiendi kehakaalu vahemik. Kirjeldatakse ülemise ning alumise piirväärtuse määramisega.
Patsiendi sugu	Kirjeldatakse patsiendi soo määramisega.
Tekst	Kirjeldatakse tekstilise väärtuse määramisega. Saab kasutada vaid reegli paremal (tulemuse) poolel.

Tabel 3: Sisemise reeglibaasi poolt toetatavad parameetrid

Kirjeldataud parameetreid kasutades võimaldab reeglibaas kirjeldada erinevatest elementidest koosnevaid reegleid. Reegli element võib olla kas reegel ise, tingimuste grupp või tingimus. Järgnevas tabelis on reeglibaasi kavandatavaid elemente täpsemalt kirjeldatud.

Reeglibaasi element	Kirjeldus
Reegel	<p>Reegel on reegli hierarhilise struktuuri kõige kõrgemalseisev element, mis määratleb mingitest tingimustest tuleneva järelduse. Reeglit võib vaadelda kui valikulauset, mis koosneb tingimuse ning tulemuse poolest.</p> <p>Reegli parameetriks on tüüp, mis määrab kas tegemist on piirangut, soovitusel või lisainfot kajastava reegluga. Tüübist tulenevalt saab ravitöö infosüsteem reegli tulemust sobivalt esitada.</p> <p>Reegel koosneb vasakust ning paremast poolest, kus vasak pool defineerib reegli tingimuse (IF) ning parem pool tingimuse tõseks osutumisel rakendatava tulemuse (THEN).</p> <p>Reegli vasaku ning parema poole elemendiks on tingimuse grupp, mis võib omakorda sisaldada ühte või mitut tingimust või tingimuste gruppi.</p>
Tingimuste grupp	Tingimuste grupp koondab ühe või mitu tingimust või tingimuste gruppi ning omab parameetrina loogikatehet (esialgu AND või

	OR). Loogikatehe määrab, kas antud tingimuste grupi tõeseks osutumiseks peavad tõesed olema kõik (AND puhul) või vähemalt üks (OR puhul) gruppi kuuluvatest tingimustest või tingimuse gruppidest.
Tingimus	<p>Tingimusega määratakse mingi parameetri väärtus või väärtuste vahemik, millele väline sama informatsiooni sisaldav väärtus peab antud tingimuse tõeseks osutumiseks vastama.</p> <p>Parameetrid esitavad erinevat liiki infot, mis olid kirjeldatud ülaltoodud tabelis nr 3 (Sisemise reeglibaasi toetatavad parameetrid).</p> <p>Tingimused jagunevad meditsiinilisteks ning mittemeditsiinilisteks tingimusteks. Meditsiinilistel tingimustel on lisatunnuseks skoop, mis määrab kas tingimuse parameetreid tuleb vaadelda kõigi patsiendi meditsiinitoimingute või käesoleva haigusjuhtumi meditsiinitoimingute kontekstis (näiteks krooniliste haiguste diagnoosid on olulised kogu patsiendi elueal, kuid mõne trauma diagnoos omab tähtsust võibolla ainult ühe haigusjuhtumi piires).</p>

Tabel 4: Sisemise reeglibaasi elementide kirjeldus

Füüsiliselt on reegli vasakul ning paremal poolel võimalik kirjeldada lõpmatult tingimusi ning tingimuste grupe. Loogiliselt on reegli paremal (tulemuse) poolel aga enamasti vaid üks tingimus. Seda põhjusel, et arvutiprogrammil on keerukas mitme tulemuse puhul sobivat välja valida. Reegli vasakul poolel saab loogiliselt tingimusi olla üks või enam.

On tingimusi, mida loogilisest vaatenurgast saab kasutada vaid reegli paremal (tulemuse) poolel. Üheks selliseks tingimuseks on tekstiline, mis sisaldab väärtusena vabateksti. Sellist tingimust kasutatakse enamasti soovitude ning muude kirjelduste edastamiseks. Üks abstraktne näide reeglist, kus vabatekstilist tingimust saaks kasutada, on: "kui patsiendi eksisteerib kopsuvähiga seonduv diagnoos, siis anna soovitusi suitsetamisest loobumiseks". Soovitude osa vabatekst võib sealjuures sisaldada ka veebiviiteid ning ka struktureeritud infot, mida ravitöö infosüsteem sobivale kujule teisendada suudab.

Käesoleva töö lisa nr 2 on toodud üks näide reegli andmekoosseisust, mis sisaldab vasakul poolel mitut ning paremal poolel ühte tingimust.

4.2.1 Sisemise reeglibaasi haldusliidese kirjeldus

Reeglibaasi haldusliides on vajalik sisemise reeglibaasi haldamiseks - reeglite nägemiseks, lisamiseks, muutmiseks, kustutamiseks ning reeglite rakendumise testimiseks.

Nagu iga uue tarkvaraga, on ka reeglibaasi haldusvahendi juurutamine võrdlemisi keerukas protsess. Seda põhjusel, et kasutajatel ei ole varem sarnase vahendiga kokkupuudet olnud ning nende jaoks võib olla ähmane ka vahendi kasutuselevõtust saadav kasu. Kui reeglibaasi haldamine on keerukas ja arusaamatu, on kasutajate vastuseis selle kasutamisele suureks riskiks kogu süsteemi rakendamisel.

Seega on vajalik luua reeglite haldamiseks arusaadav ning esialgu pigem minimalistlik vahend, mida tulevikus saab vastavalt kasutaja soovidele täiendada.

Järgmises tabelis on kirjeldatud käesoleva töö raames kavandatavale reeglibaasi haldusliidesele esitatavad funktsionaalsed nõuded.

Nõude ID	Kirjeldus
RQ_RB_F_1	Reeglibaasi haldusliides peab võimaldama uute reeglite lisamist ning olemasolevate muutmist ning kustutamist.
RQ_RB_F_2	Reeglibaasi haldusliides peab võimaldama kirjeldatud reeglite seast otsingut. Otsingu tingimusteks peab olema võimalik määrata kõikide "reegel" ning "tingimus" liiki elementide parameetreid. Otsing peab tagastama kõik reeglid, kus otsingus määratud tingimused eksisteerivad.
RQ_RB_F_3	Reeglibaasi haldusliides peab võimaldama reeglibaasis olevate reeglite nimekirja kuvamist koos reegli juurde kuuluvate tingimuse gruppide ning tingimustega. Selgelt peab olema eristatud reegli alamelementide hierarhia ning kuuluvus.

RQ_RB_F_4	Reeglibaasi haldusliides peab võimaldama reegli juurde kuuluvate tingimuste ning tingimuse gruppide muutmist, lisamist ning kustutamist.
RQ_RB_F_5	Reeglibaasi haldusliides peab võimaldama elementide "reegel", "tingimuste grupp" ning "tingimus" ning elemendi liigi juurde kuuluvate parameetrite sisestamist.
RQ_RB_F_6	Reeglibaasi haldusliides peab võimaldama reeglibaasi testimist. Haldusliideses peab olema võimalik määrata reeglibaasis toetatavate parameetrite väärtuseid (fakte), misjärel peab olema näha millised tulemused järeldusmootor nende põhjal tagastab.

Tabel 5: Sisemise reeglibaasi haldusliidese funktsionaalsed nõuded

Järgmises tabelis on kirjeldatud reeglibaasi haldusliidesele esitatavad mittefunktsionaalsed nõuded.

Nõude ID	Kirjeldus
RQ_RB_NF_1	Reeglibaasi haldusliidese kasutamine peab olema võimalik vaid vastavat õigust omavale kasutajale. Kasutaja autentimine peab toimuma kasutajanime ning parooli kasutades.
RQ_RB_NF_2	Reeglibaasi haldusliidese reageerimisaeg ükskõik millisel seal võimaldataval toimingul ei tohi olla pikem kui 10 sekundit.
RQ_RB_NF_3	Reeglibaasi haldusliides peab olema kasutatav läbi veebilehitseja.
RQ_RB_NF_4	Reeglibaasi haldusliides peab andma toimingu ebaõnnestumise korral kasutajale vastavasisulise teate.
RQ_RB_NF_5	Reeglibaasi haldusliides peab andma andmete muudatustoimingute lõpus kasutajale teate toimingu õnnestumise või ebaõnnestumise kohta.

RQ_RB_NF_6	Reeglibaasist peab olema võimalik teha varukoopiaid ning reeglibaasi peab olema võimalik varukoopiast taastada.
------------	-----------------------------------------------------------------------------------------------------------------

Tabel 6: Sisemise reeglibaasi haldusliidese mittefunktsionaalsed nõuded

Kirjeldatud nõuetest tulenevalt on järgmiseks kirjeldatud reeglibaasi haldusliidese loomise põhimõtted ning peamised omadused.

Omaduse ID	Kirjeldus	Viide nõuetele
RB_P_1	Reeglibaasi haldusliides luuakse veebirakendusena, mille kasutamine on võimalik kõigi A-klassi veebilehitsejate viimaste ning eelviimaste versioonidega.	RQ_RB_NF_3
RB_P_2	Ligipääsu piirangud reeglibaasi haldusliidesele ning kasutajate autentimine realiseeritakse rakendusserveri ligipääsude piiramise meetoditega.	RQ_RB_NF_1
RB_P_3	Reeglibaasi haldusliidese andmevahetus realiseeritakse asünkroonsete ajax päringutega, mis võimaldab korraga uuendada vaid vajalikku osa ekraanipildist ning vältida ebavajalikku lisaandmevahetust.	RQ_RB_NF_2
RB_P_4	Reeglibaasi haldusliidese juurde luuakse vaade reeglite nimekirja nägemiseks, kus kuvatakse reeglid ning reegli alla kuuluvad tingimuste grupid ning tingimused. Iga elemendi kohta esitatakse elemendi liigile iseloomulikud andmed.	RQ_RB_F_3
RB_P_5	Reeglibaasi haldusliidese juurde luuakse võimalus uue reegli lisamiseks, millega luuakse reeglibaasi uus reegel ning selle vasaku ning parema poole külge üks tingimuste grupp.	RQ_RB_F_1
RB_P_6	Reeglibaasi haldusliidese reeglite nimekirjas on iga reegli elemendi juures võimalik teostada järgmiseid toiminguid:	RQ_RB_F_1, RQ_RB_F_4,

	<ul style="list-style-type: none"> • reegel: muutmine, kustutamine; • tingimuste grupp: kustutamine, muutmine, uue alam-tingimuste grupi lisamine, uue alam-tingimuse lisamine; • tingimus: muutmine, kustutamine. 	RQ_RB_F_5
RB_P_7	Reeglibaasi haldusliidese juurde luuakse võimalus reeglite otsinguks. Otsingu juures on võimalik kasutada vabateksti ning "tingimus" liiki elementide parameetreid. Otsing tagastab kõik tulemused, mis määratud tingimustele vastavad.	RQ_RB_F_2
RB_P_8	Reeglibaasi haldusliidese juures realiseeritakse üldine teavituste süsteem, mis edastab kasutajale teateid toimingute õnnestumistest ning veaolukordadest ning kirjutab veaolukorra info ka süsteemi logidesse.	RQ_RB_NF_4, RQ_RB_NF_5
RB_P_9	Reeglibaasi haldusliidese kaudu sisestatud reeglite andmed hoiustatakse relatsioonilises andmebaasis, mille vahenditega on võimalik reeglibaasist teha varukoopiaid ning reeglibaasi taastamist.	RQ_RB_NF_6
RB_P_10	Reeglibaasi haldusliidese juurde luuakse võimalus tingimustes kasutatavate parameetrite väärtuste (faktide) määramiseks ning faktide tõttu käivitatud reeglite tulemuste nägemiseks.	RQ_RB_F_6

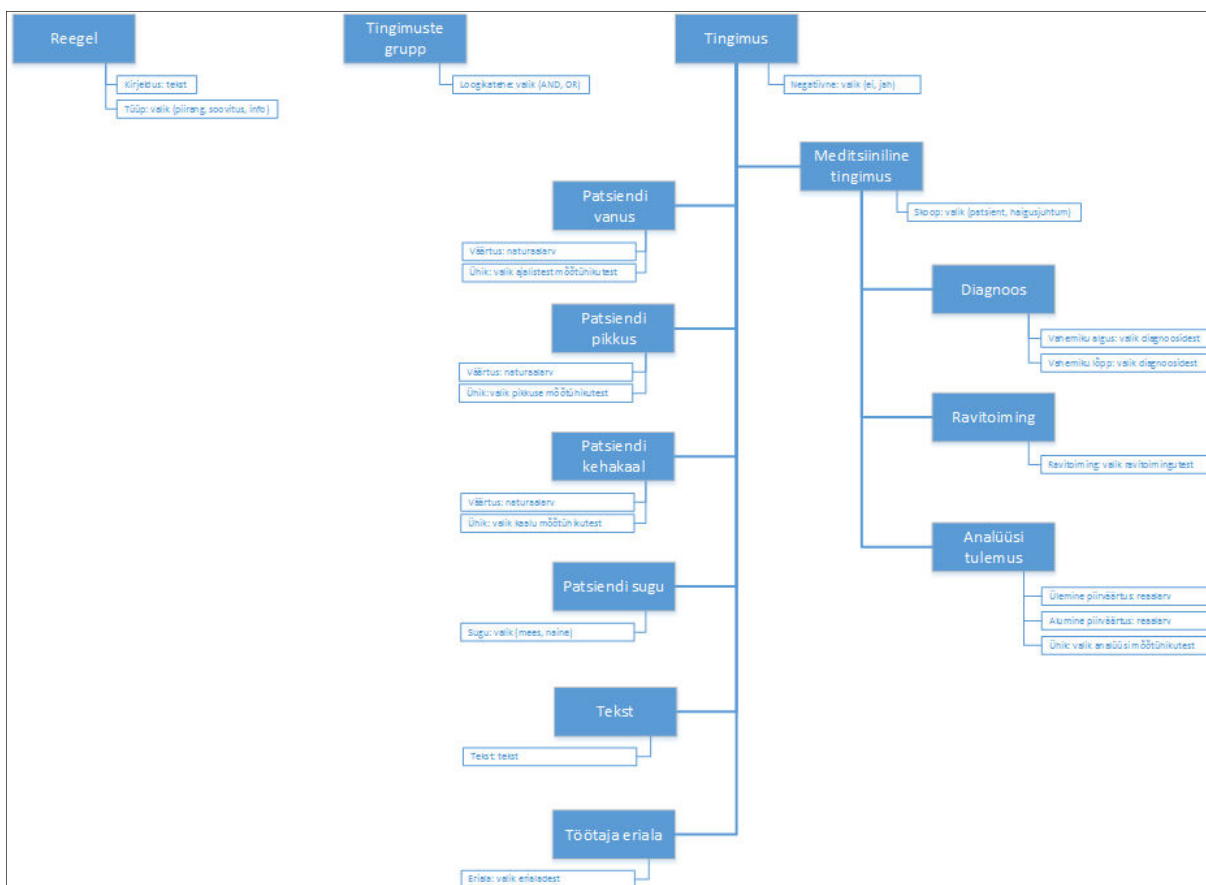
Tabel 7: Sisemise reeglibaasi haldusliidese omadused

Reeglibaasi haldusliidese realiseeritakse seega võimalus reeglite vaatamiseks, otsimiseks, testimiseks ning kolme liiki põhielemendi (reegel, piirangute grupp, piirang) lisamiseks, muutmiseks ning kustutamiseks.

Reegli ja tingimuste grupi andmekoosseis on üheselt määratud. Tingimuse andmekoosseis sõltub aga tingimuse liigist.

Käesolevas peatükis kirjeldatud reeglibaasi toetatavate parameetrite (tabel 3), elementide (tabel 4) ning nõuete põhjal on järgneval joonisel kujutatud reeglibaasi elementide loogiline andmekoosseis. Iga kujutatud elemendi juurde kuuluvad tema enda kui ka temast hierarhiliselt

kõrgemal paikneva elemendi parameetrid (näiteks tingimus "diagnoos" omab parameetreid "vahemiku algus", "vahemiku lõpp", "skoop" ning "negatiivne").



Joonis 8: Sisemise reeglibaasi loogiline andmemudel

Sisemise reeglibaasi füüsiline andmemudel kirjeldatakse käesoleva töö peatükis 4.4.2.

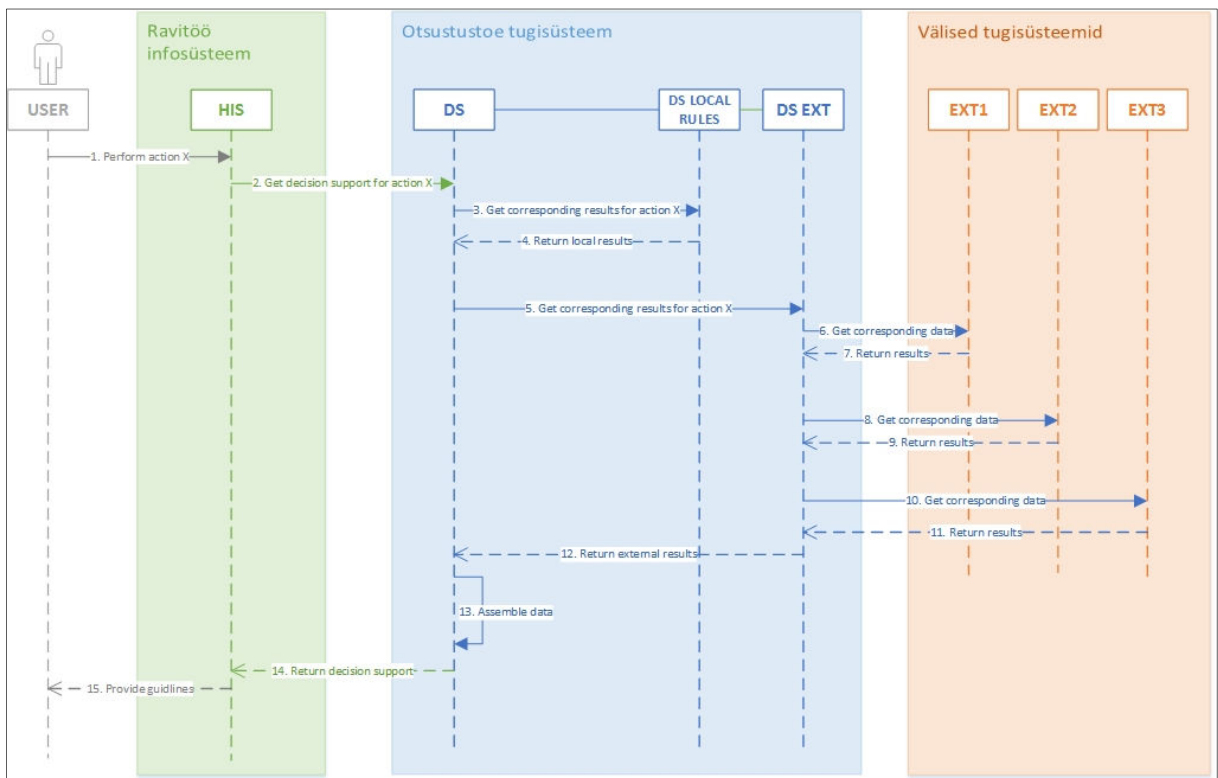
Sisemise reeglibaasi haldusliides realiseeritakse esmalt osalise funktsionaalsusega prototüüpide, prototüüpide kasutusvõimaluste analüüsi ning lõpliku realiseerimise teel. Käesoleva töö kirjutamise hetkel on jõutud osalise funktsionaalsusega prototüüpide valmimiseni, mille ekraanivormid on nähtavad käesoleva töö lisan nr 1.

Hetkel on prototüübi tasemel realiseerimata veel otsingu ning testimise funktsionaalsus, kuna ei ole selge, mil viisil seda kõige mõistlikum oleks teha. Kavas on uurida analoogset funktsionaalsust omavaid rakendusi ning leitu põhjal kujundada ka sisemise reeglibaasi haldusliidese otsingu ülesehitus.

4.3 Integratsiooni kirjeldus

Kavandatav otsustustoe tugisüsteem peab olema integreeritav ravitöö infosüsteemi ning väliste otsustustoe tugisüsteemidega. Samuti peab olema võimalik sisemise reeglibaasi kasutamine.

Alloleval joonisel on kujutatud integreeritavad süsteemid ning nende vahel liikuvad sõnumid otsustustoe leidmisel. Skeemi koostamisel on kasutatud kirjeldatud süsteemide arhitektuurilistel skeemidel näidatud komponentide interaktsioone (joonised 1 ja 7), mille põhjal on konkreetsete komponentide andmevajadustest lähtuvalt kirjeldatud päringute järjekord ning liikuvad andmed. Järgnevalt kirjeldatakse täpsemalt ka päringute sisu.



Joonis 9: Otsustustoe tugisüsteemi integratsioonisõnumite jadadiagramm

Sõnumivahetuse alustajaks on ravitöö infosüsteemi kasutatav lõppkasutaja, kes mingi toiminguteostamise kaudu annab ravitöö infosüsteemile korralduse otsustustoe pärimiseks (joonisel p.1).

Seejärel koostab ning edastab ravitöö infosüsteem päringu otsustustoe tugisüsteemi päringuliidesele (joonisel p.2). Päringus edastatakse toimingute andmed ning küsitakse toimingule vastavat otsustustoe infot.

Otsustustoe tugisüsteemi päringuliides võtab siseneva otsustustoe päringu vastu ning koostab ja edastab sisemises päringumoodulis päringud kahele alamoodulile:

- sisemise reeglibaasi päringumoodulile (joonisel p.3);
- välise tugisüsteemide päringumoodulile (p.5).

Vastavalt rakenduskoha reaalsele vajadusele on võimalik, et sisemist reeglibaasi või väliseid tugisüsteeme ei kasutata. Sel juhul jääb ära ka päring vastavale alamoodulile.

Väliseid süsteeme võib otsustustoe tugisüsteemiga olla liidestatud mitu (alloleval joonisel on neid kolm). Need võivad olla spetsialiseerunud mingile kindlale erialasele infole, kuid sama infot võivad tagastada ka erinevad süsteemid (üks selline näide on informatsioon ravimi koostoimete kohta). Välise süsteemide päringumoodul pärib infot kõigilt päringu tingimustele vastavalt väliselt süsteemidelt ning saab vastused (joonisel p.6 - p.11). Seejärel tagastab leitud info otsustustoe tugisüsteemi päringumoodulile (joonisel p.12). Sisemise reeglibaasi kasutamisel tagastab leitud info ka sisemise reeglibaasi päringumoodul (joonisel p.4).

Seejärel koostab otsustustoe tugisüsteemi päringumoodul alamoodulitest saadud info põhjal ühtse kombineeritud vastuse (joonisel p.13) ning tagastab selle ravitöö infosüsteemile (joonisel p.14).

Ravitöö infosüsteem viib saadud info sobivale kujule (näiteks koostab ekraanipildi saadud hoiatustest või soovitustest) ning edastab selle lõppkasutajale (joonisel p.15).

Kuna süsteemid on erinevad, võib süsteemide integreerimisel tekkida nende eripäradest lähtuvalt vajadus ka nende andmemudelites paiknevate andmete teisendusteks (näiteks võib sama info olla erinevates süsteemides esitatud erinevalt). Integreeritavate süsteemide andmete ühildamise kava kirjeldatakse täpsemalt käesoleva töö peatükis 4.4.1.

4.4 Andmemudeli kirjeldus

Kuna kavandatav otsustustoe tugisüsteem peab võimaldama ravitöö infosüsteemi andmemudelile vastava otsustustoe leidmist ning esmajoones rakendatakse kavandatav süsteem ühe konkreetse ravitöö infosüsteemi juures, siis on mõttekas ka otsustustoe tugisüsteemi andmemudel kavandada konkreetse ravitöö infosüsteemi andmemudelil lähtuvalt. Juhul kui edaspidi tekib vajadus otsustustoe tugisüsteemi rakendamiseks mõne

teistsuguse andmemudeliga ravitöö infosüsteemi juures, on vajalik paigaldada ravitöö infosüsteemi ning otsustustoe tugisüsteemi vahele integratsioonimoodul, mis andmed sobivale kujule teisendab. Kui teisendamine osutub võimatuks (andmeid ei saa üheselt seostada), siis tekib vajadus ka otsustustoe tugisüsteemi andmemudeli muutmise järele. Siiski on samalaadse ravi pakkujate tööprotsessid ning protsessides kasutatav informatsioon võrdlemisi sarnased. Erinevusi võib olla samasisulise info klassifitseerimisel.

Käesoleva alampeatüki esimeses osas kirjeldatakse kavandatava otsustustoe tugisüsteemi ning peatükis 3 kirjeldatud integreeritavate süsteemide andmemudelite ühildamise kava.

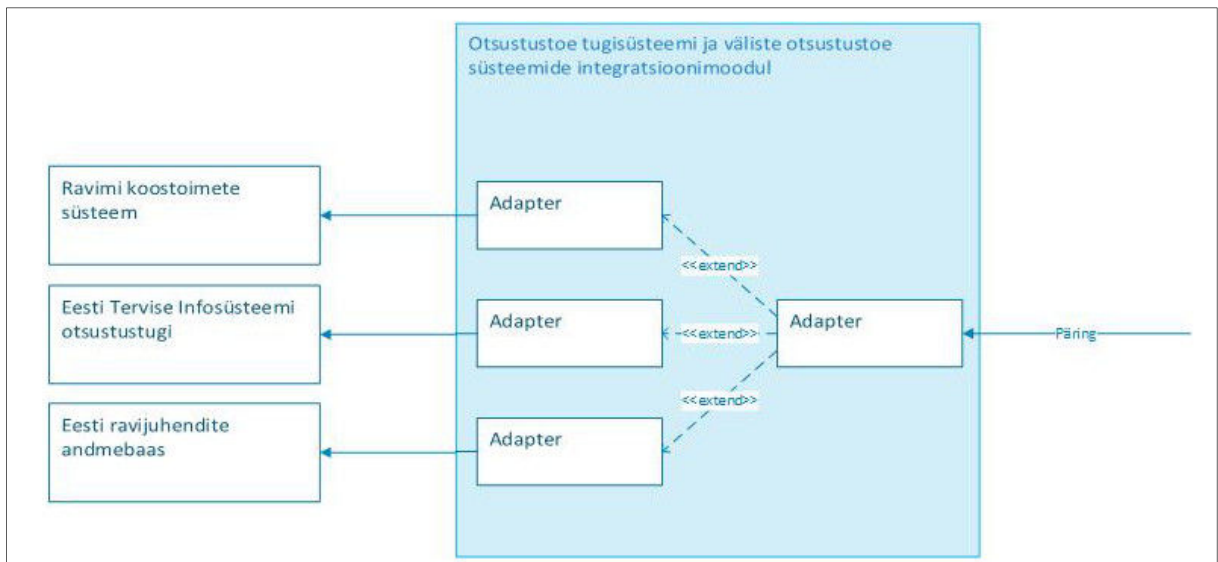
Seejärel kirjeldatakse kavandatava otsustustoe tugisüsteemi sisemise reeglibaasi andmemudel, mis peab muuhulgas võimaldama eesti ravijuhendite andmebaasi reeglite hoiustamist (peatükis 3.4 kirjeldatud mahus).

Viimaks kirjeldatakse kavandatava otsustustoe tugisüsteemi integratsioonisõnumite andmemudelid. Nende kaudu pärib ravitöö infosüsteem otsustustoe infot saab sobival kujul vastused. Integratsioonisõnumite andmemudel peab rahuldama sisemise reeglibaasi ja väliste otsustustoe andmekogude vajadusi nii päringute kui ka päringu vastuste andmekoosseisude osas.

4.4.1 Integreeritavate süsteemide andmemudelite ühildamise kava

Integratsioonimooduli loomisel on kavas kasutada tavapärasest integratsiooniplatvormi arhitektuuri: iga andmekogu jaoks luuakse spetsiifiline adapter, mis omab oskust selle andmekoguga suhelda ning andmed vajalikule kujule teisendada. Andmekogude adapteritel on ka ühine funktsionaalsus, mis sisaldab muuhulgas funktsionaalsust suhtlemaks päriava süsteemiga. Sisuliselt on tegemist teenusepõhise arhitektuuriga (SOA), kus iga andmekogu adapter pakub päriavale süsteemile andmekoguga suhtlemise teenust.

Eeltoodust tulenevalt on järgneval joonisel kujutatud otsustustoe tugisüsteemi ning väliste süsteemide vahel paikneva integratsioonimooduli arhitektuur, mis sisenevas päringus paikneva info väliste süsteemide jaoks sobivale kujule teisendab. Säärase funktsionaalsuse eesmärgiks on saavutada päritavate andmete osas süsteemide semantiline koosvõime - „tarkvarasüsteemide võime teistelt infosüsteemidelt saadud andmeid adekvaatselt kasutada (1 lk 25)“.



Joonis 10: Otsustustoe tugisüsteemi ja välise otsustustoe süsteemide integratsioonimooduli arhitektuur

Soovituslik andmeintegratsiooni arhitektuuri mall (Reference Architecture for Data Integration (18)) toob soovitusliku integratsiooniprotsessi juures välja järgmised etapid:

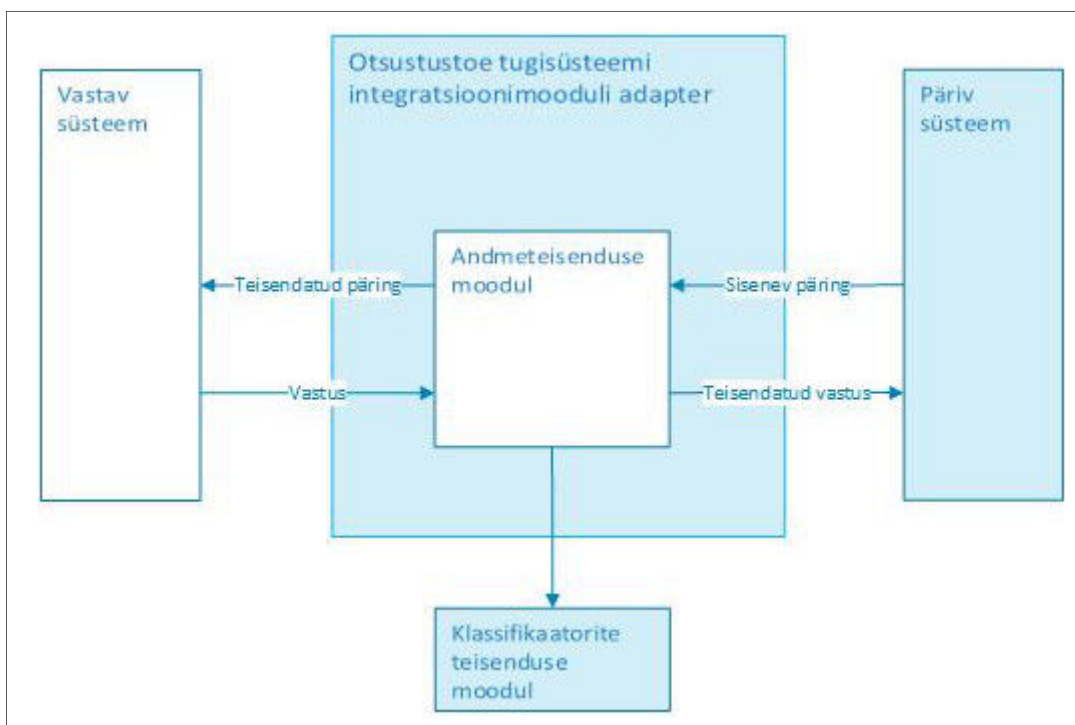
- lähteandmete pärimine ning salvestamine;
- lähteandmete andmekvaliteedi kontrollimine;
- lähteandmete puhastamine ning probleemide raporteerimine;
- vajalike andmeteisenduste tegemine;
- andmete pärija süsteemi vajadustest lähtuv ettevalmistus ning salvestamine.

Samas nenditakse, et etappide kasutamine peaks toimuma vastavalt konkreetsetele vajadustele. (18 lk 21)

Käesoleva töö autor leiab töö vajadustest lähtuvalt, et mõttekas on realiseerida vaid andmeteisenduse ning ettevalmistuse osa. Näiteks võib olla vajalik teisendada andmeid erinevate klassifikaatorite vahel. Samuti teisendada erinevate ühikutega kirjeldatud samasisulisi väärtuseid, et erinevad süsteemid infost samal viisil aru saaksid. Integratsiooniprotsessi sees tekkivate ajutiste andmete salvestamise vajadust hetkel ette näha ei ole. Kui päringumahud kasvavad ning tekivad enamkasutatavad päringute mustrid, võib osutada mõttekaks päringutulemuste ajutine vahemälu hoidmine.

Iga integratsioonimooduli adapter sisaldab spetsiifilist funktsionaalsust konkreetse andmekoguga suhtlemiseks ning kasutab ka välist funktsionaalsust, et erinevate süsteemide vahel liikuvad andmed sobivale kujule teisendada.

Järgneval joonisel on eelnevalt kirjeldatud integratsioonikava põhjal kujutatud ühe integratsioonimooduli adapteri arhitektuur.



Joonis 11: Otsustustoe tugisüsteemi integratsioonimooduli adapteri arhitektuur

Integratsioonimoodulina saaks kindlasti kasutada ka mõnda juba olemasolevat integratsiooniplatvormi. Kuid ka siis tuleks luua adapterid enamuste välise andmekogudega suhtlemiseks. Olemasolevat adapterilahendust saaks arvatavasti kasutada vaid Eesti Tervise Infosüsteemi otsustustoe pärimisel, kuna seal kasutatakse IHE XDS sõnumivahetuse standardit.

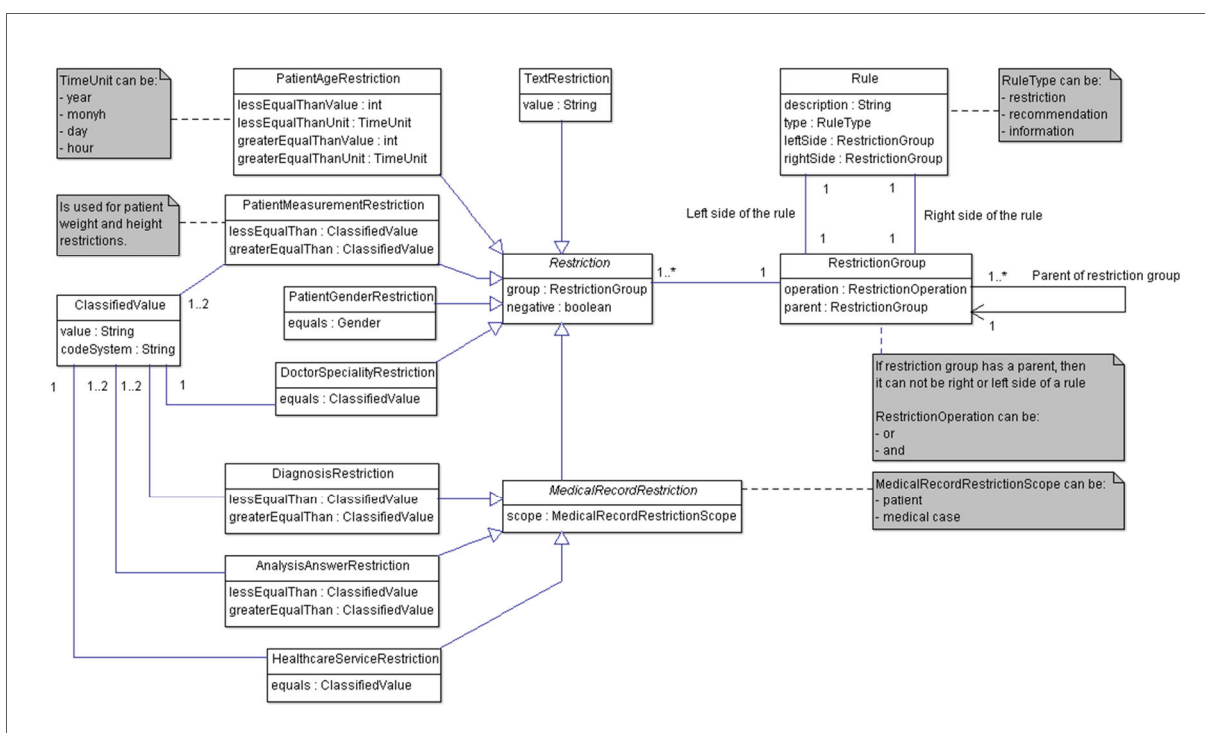
Kuna välise integratsiooniplatvormide juures on üheks argumendiks hind ning et käesoleva töö raames integreeritavaid väliseid süsteeme on vähe, siis käesoleva töö vajadustest lähtuvalt tundub töö autorile mõttekas luua kohalik integratsiooniplatvorm, mis sõnumivahetuses põhineb vajalikule andmemudelile. Välise andmeallikate lisandumisel võib aga tulevikus olla mõttekas kasutada ka teisi integratsioonilahendusi.

4.4.2 Sisemise reeglibaasi andmemudeli kirjeldus

Kavandatava otsustustoe tugisüsteemi reeglibaas peab võimaldama raviasutuse defineeritud reeglite hoiustamist. Kuna ühe andmeallikana kasutatakse ka eesti ravijuhendite andmebaasis kirjeldatud juhendeid, siis peab sisemine reeglibaas võimaldama ka seal kirjeldatud andmete sobival kujul hoiustamist.

Järgnevalt on joonisel 12 kujutatud sisemise reeglibaasi andmemudel, mis on koostatud vastavalt ravijuhendite andmebaasi parameetritele (kirjeldatud peatükis 3.4 tabelis 2).

Tingimuste väärtused määratakse nii väärtuse kui kasutatava klassifikatsiooni paarina (joonisel element „ClassifiedValue“). See võimaldab vajadusel kasutada erinevaid klassifikatsioone. Üldjuhul peaks ravitöö infosüsteemi ning kavandatava otsustustoe tugisüsteemi klassifikaatorid olema samad, kuid võimalik on ka erinevalt klassifitseeritud andmete teisendamine. Klassifikaatorite teisendamist kirjeldati täpsemalt peatükis 4.4.1.



Joonis 12: Sisemise reeglibaasi füüsiline andmemudel

Keskseks elemendiks on „reegel“ (joonisel „Rule“), mis omab nii vasakut (tingimuste) kui paremat (tulemuste) poolt. Mõlemad pooled on määratud tingimuste grupiga (joonisel „RestrictionGroup“), mis omakorda võivad sisaldada nii tingimusi kui ka tingimuste gruppe. Lisaks omab tingimuste grupp loogikatehet („jah“ või „või“), mis määrab tingimuse alla kuuluvate alamtingimuste või tingimuse gruppide kasutamise. Tingimused jagunevad juba vastavalt tingimuse klassile – raviteenuse, analüüsi vastuse, diagnoosi, arsti eriala, patsiendi soo ning patsiendi mõõdetavate omaduste (kaal, pikkus, vanus) määramiseks.

Enamike tingimuse väärtuste määramisel kasutatakse eelkirjeldatud „ClassifiedValue“ elementi. Analooget määramise viisi kasutatakse ka teiste sisemiste väärtuste määramisel.

4.4.3 Integratsioonisõnumite andmemudelite kirjeldus

Kavandatava tugisüsteemi integratsioonisõnumeid kasutatakse otsustustoe pärimiseks ning tulemuste tagastamiseks. Eesmärgiks on seatud, et väline otsustustuge tarbiv süsteem saab suhelda ühe, käesolevas töös kavandatava otsustustoe tugisüsteemiga, mis omakorda võib kasutada info leidmiseks mitut andmeallikat.

Käesoleva töö raames kasutab kavandatav otsustustoe tugisüsteem andmeallikatena sisemist reeglibaasi, Eesti Tervise Infosüsteemi otsustustoe allsüsteemi (EbMeDS) ning ravimi koostoimete andmebaasi (SFINX-PHARAO).

Kasutatavate interaktsioonisõnumite struktuur peab toetama kõigi kolme andmeallika vajadusi – päringud peavad sisaldama sobival kujul andmeid andmeallikatest otsustustoe leidmiseks ning vastus peab võimaldama erinevatest andmeallikatest saadud tulemuste ühtsel kujul tagastamist.

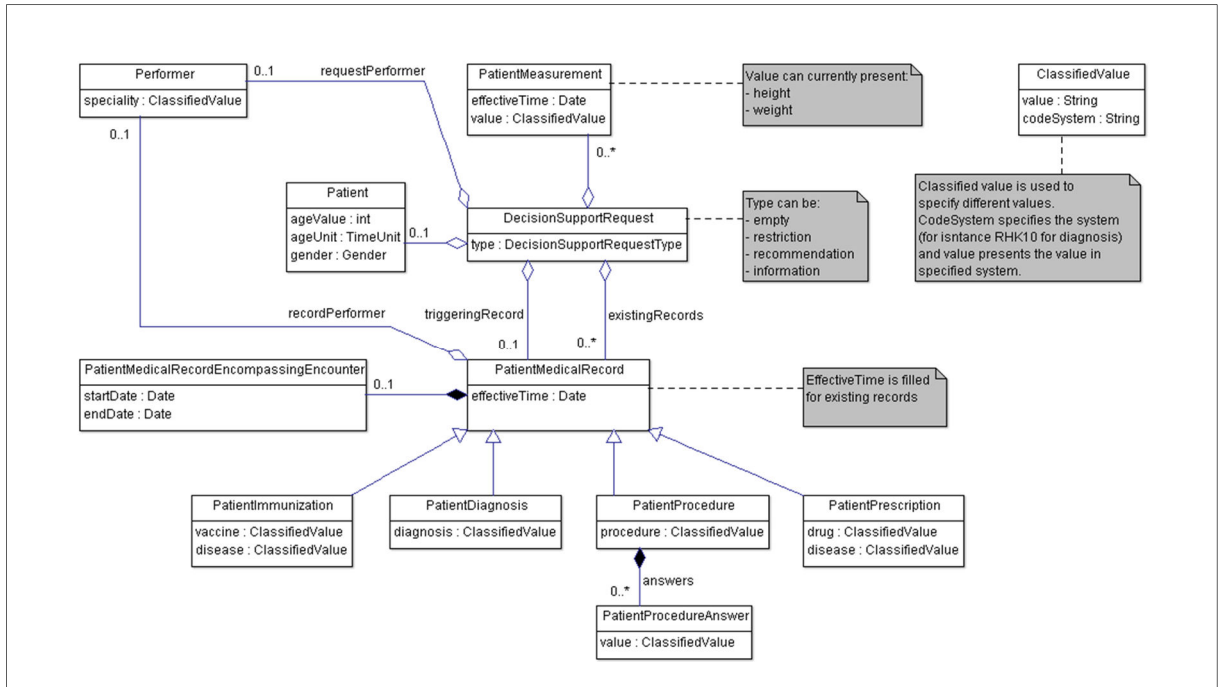
Interaktsioonisõnumite sisu defineerimisel vaatles käesoleva töö autor kolme andmeallika ning liidestatava ravitöö infosüsteemi andmemudeleid (kirjeldatud käesoleva töö peatükis 3) ning ravitöö infosüsteemi vajadusi. Leiti erinevate andmeallikate andmemudelites olev ühisosa, erisused, käesoleva töö raames olulised andmeobjektid ning defineeriti saadu põhjal käesoleva töö vajadustele vastavad andmemudelid.

Olulisuse hindamisel ravialaseid andmeid sõnumitest välja ei jäetud. Küll aga ei kavandata loodavas süsteemis andmeallikale spetsiifilise tehnilise lisainfo ravitöö infosüsteemile samal kujul tagastamist, nagu andmeallikas selle tagastab. Tehniline ja mittetehniline lisainfo tagastatakse pärivale süsteemile vajadusel üldise kirjeldusena, mida päriv süsteem saab vajadusel kasutada. Olulise tehnilise lisainfo saab loodava süsteemi juures sobiva tasemega logidesse kirjutada.

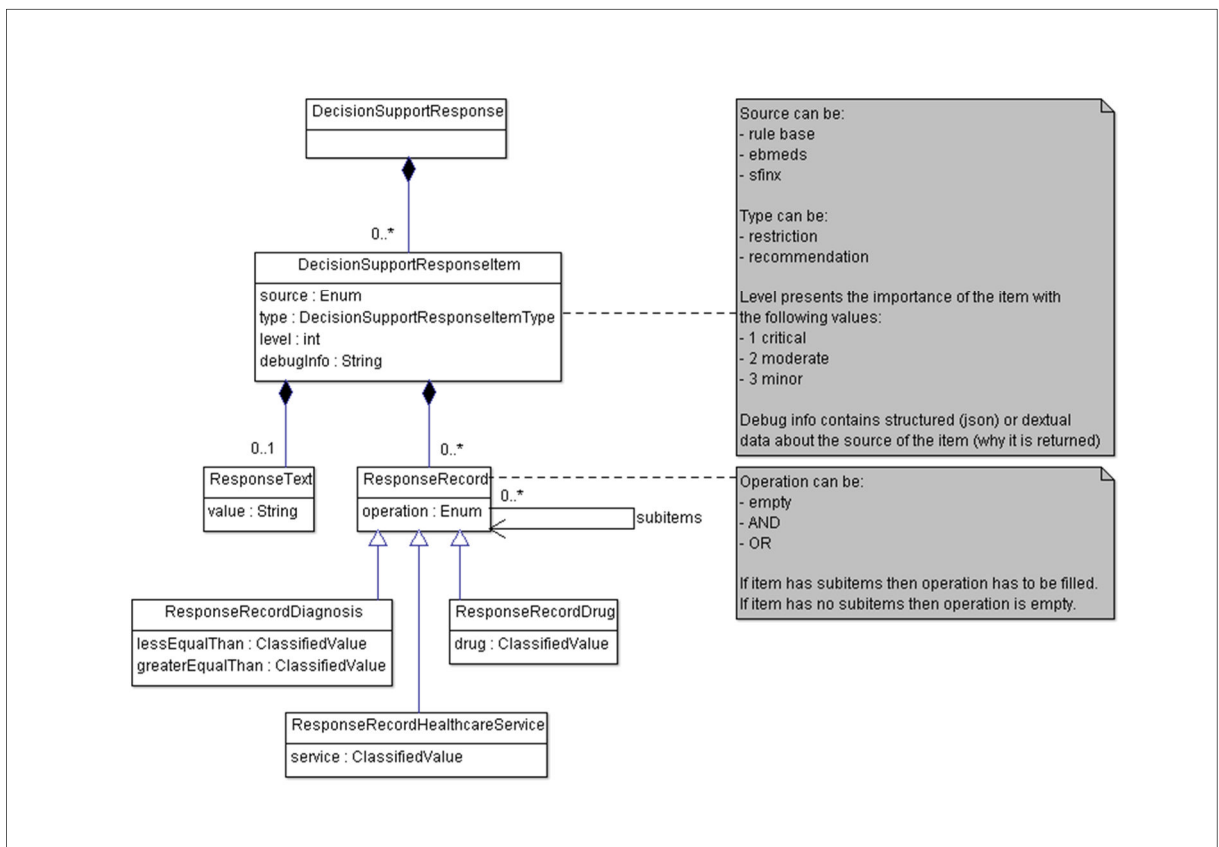
Samuti luuakse loodava süsteemi juurde väliste andmeallikatega suhtlemise veahaldus. Veasituatsioonide info kirjutatakse loodava süsteemi logidesse ning tagastatakse vastav info teksti kujul ka pärivale süsteemile (näiteks kui ühe andmeallikaga suhtlemine tehnilistel põhjustel ebaõnnestus). Veateadete kasutajale esitamise vajadus jääb päriiva süsteemi otsustada.

Järgneval kahel joonisel on näidatud kavandatava otsustustoe tugisüsteemi integratsioonisõnumite andmemudelid, mis võimaldavad pärimisel andmeallikatele vajalikul

kujul info edastamist ning vastamisel pärivale süsteemile vajalikul kujul info tagastamist. Joonised on koostatud peatükis 3 kirjeldatud väliste andmeallikate andmemudelite (joonised 2, 3, 4, 5, 6) ning ravijuhendite andmebaasi parameetrite (tabel 2) põhjal.



Joonis 13: Otsustustoe päringu andmemudel



Joonis 14: Otsustustoe vastuse andmemudel

Info edastamisel on päriaval süsteemil ning andmeallikatel võimalik andmeid edastada sobiva klassifikatsiooniga (joonistel 13 ning 14 olev element „ClassifiedValue“, mis sisaldab andmetena nii väärtust kui kasutatavat klassifikatsiooni). Eeldus on, et mõlemad pooled oskavad kasutatava klassifikatsiooniga andmeid kasutada. Seega peab kavandatav süsteem olema teadlik kõigi liidestatavate süsteemide andmemudelites kasutatavatest klassifikaatoritest ning võimalikest klassifikaatorivahelistest teisendustest. Klassifikaatorite ühildamise kava on täpsemalt kirjeldatud peatükis 4.5.1.

4.5 Järeldusmootori kirjeldus

Nagu peatükis 4.1 toodud otsustustoe tugisüsteemi arhitektuuri jooniselt 7 võib näha, on järeldusmootori ülesandeks pakkuda sisemise reeglibaasi integratsioonimoodulile ning reeglite haldamise moodulile teenuseid sisemise reeglite andmebaasiga suhtlemiseks ning reeglites paiknevat infot sobival kujul leidmiseks.

Sisemise reeglibaasi integratsioonimoodul soovib järeldusmootorilt saada järeldamise teenust, mis reeglibaasis paiknevad reeglid ning integratsioonimooduli edastatud faktid kokku viib ning reeglites paiknevad tulemused tagastab. Seejärel saab integratsioonimoodul reeglite tulemused sobivale kujule teisendada ning otsustustoe moodulile tagastada.

Reeglite haldamise moodul soovib järeldusmoodulilt samuti järeldamise teenust, kuna reeglite halduril on vajadus defineeritud reeglite testimiseks - haldur soovib kontrollida faktide kunstliku etteandmisega õigete reeglite rakendumist nende rakendumist ning korrektsete tulemuste saavutamist.

Tugisüsteemi vajadust järeldusmootori järele saab rahuldada mitmel viisil. Üheks võimaluseks on see luua ise. Kõige lihtsam järeldusmootori sisemine loogika oleks kõik reeglid järjest läbi käia, kontrollida nende tingimuse (vasaku) poole vastavust faktidele ning sobivusel määrata tõseks reegli tulemuse (paremal) poolel paiknevad faktid. Tsüklit tuleb korrata seni, kuna ühtegi reeglit enam tõseks ei õnnestu lugeda, kuna positiivse tulemuse saanud reegli poolt lisatud faktid võivad tõseks muuta mõne uue reegli. Kirjeldatud loogikat nimetatakse andmepõhiseks järeldamiseks (1 lk 30). Kuna käesolevas töös kavandatava süsteemi omadused vajavad tulemuste poolel vaid faktide lisamist ning puudub vajadus näiteks väliste süsteemide protsesside käivitamiseks, siis selline lahendus loogiliselt sobiks. Samuti saaks otse kasutada süsteemi enda andmemudelit ning puuduks vajadus selle

teisendamiseks mõne välise süsteemi oodatavale kujule (näiteks edaspidi kirjeldatava reeglismootori „Drools“ poolt kasutatavaks „drl“ andmestruktuuriks).

Samas muutub reeglite hulga kasvades kirjeldatud lahendus väga ressursimahukaks, kuna reeglite kontrollimisi tekib väga palju (19). Et antud töös kavandatav lahendus peab tulevaste kasutajate usalduse võitmiseks end juurutamisel kasutusmugavuse poolest heast küljest näitama, siis ei saa riskida võimalusega, et reeglite lisamisel selgub süsteemi suutmatust nendega toime tulla. See annaks tugeva tagasilöögi ning ohustaks süsteemi edukat rakendamist.

Efektiivsem viis andmepõhise järeldamise teostamiseks on kasutada mõnda selleks otstarbeks kohandatud algoritmi. Näiteks Rete algoritm loob reeglibaasi reeglite põhjal otsingupuud, kus puu „võras“ esitatakse reeglite vasaku poole tingimused ning „lehtedeks“ on reeglid. Sellega saab samadest faktidest sõltuvad reeglid grupeerida ning faktide lisandumisel käivitada vaid sellised reeglid, mis lisandunud faktidest ka reaalselt sõltuvad. (19)

Kuna jõudlusprobleemide tekke oht ei võimalda käesolevas töös kavandatava süsteemi juures eelnevalt kirjeldatud lihtsat andmepõhise järeldamise algoritmi rakendada, kavandatakse järelalusmootori realiseerimine Rete algoritmi baasil.

Rete algoritmil baseeruvad paljud tasulised ning ka tasuta saadaolevad reeglisüsteemid. Seega ei ole mõttekas käesolevas töös kavandatava süsteemi juures hakata uut järelalusmootorit looma, vaid mõistlik on kasutada mõnda olemasolevat lahendust. Võimalike vajalike andmemudeli teisendustele (näiteks reeglite viimine reeglisüsteemi poolt oodatavale kujule) kuluv töömaht jääb autori hinnangul kindlasti väiksemaks, kui see oleks Rete algoritmil baseeruva süsteemispetsiifilise järelalusmootori loomisel.

Kuna käesolevas töös kavandatava süsteemi vajadused välisele reeglisüsteemile on võrdlemisi väikesed (kasutatakse järelalusmootori osa) ning ka rahalisi ressursse ei ole meditsiiniastutustel selleks kuigi palju panustada (vastavalt peatükis 2.2 kirjeldatud eesti tervishoiu IT-rahastuse tasemele), võetakse esmasesse reeglisüsteemi kandidaatide nimekirja tasuta saadaolevad ning väiksema hinnaga variandid.

Eeltoodust tulenevalt kirjeldatakse järgnevalt järelalusmootorile esitatavad täpsemad funktsionaalsed ning mittefunktsionaalsed nõuded. Seejärel vaadeldakse kolme esmasesse valikusse võetud reeglisüsteemi täpsemalt, võrreldakse nende omadusi, vastavust nõuetele

ning teostatakse lõplik valik. Viimaks kirjeldatakse valitud lahenduse rakendamist käesolevas töös kavandatava tugisüsteemi juures.

4.5.1 Järeldusmootorile esitatavad nõuded

Järgnevas tabelis kirjeldatakse järeldusmootorile esitatavad funktsionaalsed nõuded ning nende kaalud vastavalt käesoleva töö autori hinnangule.

Nõude ID	Kirjeldus	Kaal
RQ_RE_F_1	Järeldusmootor peab võimaldama relatsioonilises andmebaasis paiknevate reeglite kasutamist, mis on kirjeldatud rakendatava süsteemi omadustest lähtuvalt (reeglite andmemudel on süsteemile spetsiifiline).	1
RQ_RE_F_2	Järeldusmootor peab võimaldama etteantud tingimustele vastavate reeglite leidmist (otsingut reeglite seast).	0,5
RQ_RE_F_3	Järeldusmootor peab võimaldama reeglibaasi reeglite põhjal otsingupuu kuvamist.	0,5
RQ_RE_F_4	Järeldusmootor peab sobivate reeglite otsingul rakendama efektiivseid ning selleks otstarbeks sobivaid algoritme (nt Rete algoritm).	1

Tabel 8: Järeldusmootorile esitatavad funktsionaalsed nõuded

Järgnevas tabelis kirjeldatakse järeldusmootorile esitatavad mittefunktsionaalsed nõuded.

Nõude ID	Kirjeldus	Kaal
RQ_RE_NF_1	Järeldusmootor peab olema päriava süsteemiga liidestatav SOA (Service-Oriented Architecture) põhimõtetest lähtuvalt.	1

Nõude ID	Kirjeldus	Kaal
RQ_RE_NF_2	Järeldusmootor peab olema täisfunktsionaalses mahus tasuta kättesaadav (vähemalt prooviversioonina). Soovitav on süsteemi võimalikult madal hind.	1
RQ_RE_NF_3	Järeldusmootor peab olema kasutatav MySQL või PostgreSQL andmebaasimootoritega.	1

Tabel 9: Järeldusmootorile esitatavad mittefunktsionaalsed nõuded

Lisaks kirjeldatud nõutele on käesoleva töö autori kogemuste põhjal soovituslik, kui järeldusmootori liidestamine väliste süsteemidega (reeglibaas, päriv süsteem) on võimalik Java programmeerimiskeelt kasutades.

4.5.2 Vaadeldud järeldusmootorite ülevaade

Nagu peatükis 4.6 kirjeldati, on üheks valikukriteeriumiks järeldusmootori võimalikult soodne hind. Funktsionaalse poole pealt on vajalik efektiivse otsingualgoritmi kasutamine ning võimalikult lihtne liidestatus loodud rakendustega, mis järeldusmootori pakutavaid teenuseid kasutama hakkavad.

Selles tulenevalt teostati esmane valik ning valiti välja kolm kandidaati, mida järgnevalt täpsemalt kirjeldatakse.

Drools

Drools on Java programmeerimiskeeles kirjutatud ärireeglisüsteem (BRMS), mis on alates 2005 aastast osa „JBoss Enterprise BRMS“ nimelisest paketist. Reeglibaasi kasutamisel rakendab ning laiendab Drools Rete algoritmi ning võimaldab nii andmepõhist kui ka sihipõhist järeldamist. (20 lk 3)

Drools koosneb viiest põhimoodulist, mida on võimalik rakendada vastavalt vajadustele koos või ka individuaalselt. Nendeks on:

- jBPM- äriprotsesside haldamise moodul;
- Drools Expert – järeldamise moodul;

- Drools Fusion – andmeanalüüsi ning situatsioonituvastuse moodul;
- Drools Guvnor – reeglibaasi haldamise moodul;
- Drools Planner – automaatse ressursiplaneerimise moodul. (20 lk 34)

Drools on avatud lähtekoodiga ning litsentseeritud „Apache Software License 2.0“ põhjal, mis võimaldab Drools'i tasuta kasutada ning modifitseerida ka tasulistes ning suletud lähtekoodiga tarkvaratoodetes. (21)

Käesolevas töös planeeritava tugisüsteemi vajadustest lähtuvalt oleks mõttekas Drools'i viiest moodulist kasutada järeldamise moodulit „Drools Expert“. Kuna reeglibaasi haldus on kavas teostada omatehtud vahenditega, siis ei ole vajadust Drools'i pakutava reeglibaasi haldamise vahendi järele. Kuid see oleks vajadusel võimalik rakendada reeglibaasi reeglitest parema ülevaate saamiseks, kui „Drools Guvnor“ seda võimaldab.

OpenRules

OpenRules on ulatusliku funktsionaalsuse ning avatud lähtekoodiga ärireeglisüsteem. OpenRules'i veebileht (22) omab laialdast infot süsteemi võimaluste kohta ning ka paljusid näiteid. Veebilehelt leitud info põhjal on ka koostatud järgnev süsteemi kirjeldus.

Reeglite hoiustamiseks kasutatakse OpenRules'is Excel'i „xls“ faile, kuid käesoleva töö autori arvates ei tasu selle põhjal järeldada, et süsteem on ajale jalgu jäänud. Excel'i asemel saab kasutada ka näiteks Google Docs'i ning seeläbi reegleid „pilves“ hoida. Samuti saab reegleid lugeda näiteks andmebaasitabelist (siiski „xls“ faili struktuuri järgi). Reeglite defineerimine tuttavas tabelarvutussüsteemis on jõukohane ka äripoolle inimestele ning ei vaja ka eraldi haldusliidese loomist ega tundmaõppimist. Seeläbi on ka kogu reeglisüsteemi õppimiskõver kasutajasõbralikum.

OpenRules sisaldab palju erineva otstarbega komponente. Paljud neist on huvipakkuvad ka käesolevas töös kavandatava tugisüsteemi omadustest lähtuvalt. Nendeks on:

- Rule Engine – järeldamise moodul;
- Rule Validator – reeglite kontrollmoodul (kontrollib reeglite süntaksit);
- Rule Testing – reeglite loogilise poole testimismoodul (võimaldab „xls“ failide kaudu testandmete loomist ning reeglite testimist);

- Rule Deployment – järelusmootori rakendamise moodul (võimaldab järelusmootorit käivitada SOA põhimõtetest lähtuvalt, Java rakenduse koosseisus läbi API, enda kasutajaliidest omava veebirakendusena);
- Rule Learner – andmekaeve ning reeglite õppimise moodul (õppeandmete peal masinõppe algoritme rakendades püütakse ekspertteadmiste abiga leida uusi seoseid ning luua uusi reegleid);
- Rule Compressor – olemasolevate reeglite grupeerimis- ja üldistusmoodul (võimaldab paljud sama asja kohta käivad reeglid viia vähendatud kuid siiski samasisulisele kujule)

OpenRules'i vähendatud funktsionaalsusega versiooni saab prooviversioonina alla laadida tasuta.

Avatud lähtekoodiga rakendustes saab OpenRules'i kasutada GPL litsentsi alusel litsentsitasuga 99 USA dollarit installatsiooni kohta. GPL litsents nõuab, et GPL'iga litsentseeritud toodet kasutav toode oleks samuti GPL litsentsi tingimustega vastavuses ning avatud lähtekoodiga.

Suletud lähtekoodiga rakendused saavad OpenRules'i kasutada litsentsitasuga, mis käesoleva töö kirjutamise hetkel on 3695 USA dollarit installatsiooni kohta.

Käesolevas töös planeeritava tugisüsteemi vajadustest lähtuvalt võiks kaaluda OpenRules'i „Rule Engine“ ning „Rule Deployment“ komponentide rakendamist. See annaks võimaluse reeglite põhjal tehtavate järeluste tegemise delegeerida OpenRules'ile ning liidestamine teostada kas SOA põhimõtetest lähtuvalt või läbi vastava Java API.

Väga huvipakkuvad on ka „Rule Learner“ ning „Rule Compressor“ komponendid, mida tulevikus saaks kavandatava süsteemi juures rakendada.

„Rule Learner“ võimaldab kasutada masinõppe algoritme ning rakendusi uute reeglite loomiseks. Hetkel on olemas liides WEKA kasutamiseks ning nähakse ette ka teisi võimalusi.

„Rule Compressor“ annaks võimaluse olemasolevaid reegleid rakenduse jaoks optimeerida.

Miinuspoolena OpenRules'i kasutamisel võib käesoleva töö kontekstis näha selle jäika põhinemist tabelarvutussüsteemi struktuuril (Excel'i „xls“ failidel), mida kasutatakse reeglite kirjeldamisel. Kuigi äripoole inimestele on see arusaadavam kui mõni programmeerimiskeel,

tooks see käesolevas töös vajaduse reeglibaasi teisendusteks andmebaasi tabelitest „xls“ kujule. Juhul kui tulevikus rakendada ka OpenRules'i reeglite õppimise moodulit (mille lõpptulemus on samuti „xls“ struktuuris), tuleks luua ka teisendusalgorithm „xls“ kujul reeglite käesolevas töös defineeritud struktuuriga andmebaasitabelitesse kandmiseks.

Jess

Jess on Java programmeerimiskeeles kirjutatud ärireeglisüsteem, mis nagu ka eelkirjeldatud rakendab oma töös Rete'i algoritmi. Jess'i esimene versioon on kirjutatud aastal 1995 kui CLIPS'i analoog, kuid sealt edasi on Jess'i eraldiseisvalt edasi arendatud. Rohkelt infot on võimalik leida Jess'i veebileheküljelt (23) ning Jess'i looja Ernest Friedman-Hill'i kirjutatud raamatust „Jess in Action“ (24).

Jess võimaldab nii sisemise struktuuri järgi kirjeldatud kui ka Java's defineeritud faktibaasi kasutamist. Seeläbi saab seda Java rakendustega võrdlemisi hõlpsasti siduda. Reeglid tuleb defineerida kas Jess'i enda keeles võiks xml'ina. Käesolevas töös kavandatava süsteemi vajadustest lähtuvalt oleks mõttekam xml, kuna seda saab andmebaasis paiknevatest reeglitest genereerida.

Jess'i töö jälgimiseks (millised faktid põhjustavad milliste reeglite käivitumise) on süsteemis samuti vahendid olemas.

Jess'i on lubatud akadeemilises tegevuses tasuta kasutada. Kommertstoodangus kasutamiseks tuleb soetada tasuline litsents, mille hind selgub läbirääkimiste tulemusena.

4.5.3 Järeldusmootori valik

Järgnevas tabelis on esmase valiku reeglisüsteeme võrreldud käesolevas töö vajadustest lähtuvalt. Välja on toodud võrdlus kasutustingimuste, reeglite ning faktide kirjeldusviisi ning järeldusmootori kasutamise osas.

	Drools	OpenRules	Jess
Kasutamise tingimused	Tasuta (Apache Software License 2.0).	Avatud lähtekoodiga rakendustes praktiliselt tasuta (GPL litsents), muidu tasuline.	Akadeemilises tegevuses tasuta, kommertstegevuses tasuline.

	Drools	OpenRules	Jess
Reeglite defineerimise viis	Drools'i defineeritud struktuur. Võimalik lugeda andmebaasist või DRL failidest.	Tabelarvutusprogrammide omane struktuur (nt XLS fail).	Jess'i enda keel või XML struktuur.
Faktibaasi defineerimise viis	Drools'i enda keel, Java, XML	Java	Jess'i enda keel, Java.
Järeldusmootori kasutamine	Läbi Java API või SOA põhimõtete järgi läbi veebiteenuse.	Läbi Java API või SOA põhimõtete järgi läbi veebiteenuse.	Jess'i enda keeles või läbi Java API.
Lisainfo	Ulatuslik kasutajate kogukond ning pidevate uuenduste lisandumine. Ühilduvus Spring raamistikuga.	Võimaldab reeglibaasi uuendamist töö kestel. Suurte töömahtudega testitud. Ühilduvus WEKA'ga.	Kõige uuem versioon anti välja aastal 2008.

Tabel 10: Valimis osalenud järeldusmootorite omaduste võrdlus

Järgnevas tabelis on näidatud valikusse võetud järeldusmootorite vastavus peatükis 4.5.1 esitatud nõuetele.

Nõude ID	Nõude kirjeldus	Drools vastavus	OpenRules vastavus	Jess vastavus
RQ_RE_F_1	Järeldusmootor peab võimaldama relatsioonilises andmebaasis paiknevate reeglite kasutamist, mis on kirjeldatud rakendatava süsteemi omadustest lähtuvalt (reeglid ei ole kirjeldatud reeglumootori defineeritud struktuuri kohaselt).	Jah, luues andmebaasi ning reeglibaasi vahele teisendusliidese (andmebaas -> DRL struktuur).	Jah, luues andmebaasi ning reeglibaasi vahele teisendusliidese (andmebaas -> XLS struktuur).	Jah, luues andmebaasi ning reeglibaasi vahele teisendusliidese (andmebaas -> XML või Jess struktuur).
RQ_RE_F_2	Järeldusmootor peab võimaldama etteantud tingimustele vastavate reeglite leidmist.	Jah.	Jah.	Jah.
RQ_RE_F_3	Järeldusmootor peab võimaldama reeglibaasi reeglite põhjal otsustuspuu kuvamist.	Rete otsingupuud on võimalik näha Eclipse arendusvahendi lisa abil DRL failide põhjal.	Ei (käesoleva töö autorile teadaolevalt).	Võimaldab Java graafilise kasutajaliidese aknas Rete otsingupuud kuvamist.
RQ_RE_F_4	Järeldusmootor peab sobivate reeglite otsingul rakendada efektiivseid ning selleks otstarbeks sobivaid algoritme (nt Rete algoritm).	Jah, Rete algoritm.	Jah, Rete algoritm.	Jah, Rete algoritm.

Nõude ID	Nõude kirjeldus	Drools vastavus	OpenRules vastavus	Jess vastavus
RQ_RE_NF_1	Järeldusmootor peab olema päriiva süsteemiga liidestatav SOA (Service-Oriented Architecture) põhimõtetest lähtuvalt.	Jah.	Jah.	Ei, kuid võimalik luua ise vajalik liides.
RQ_RE_NF_2	Järeldusmootor peab olema täisfunktsionaalses mahus tasuta kättesaadav (vähemalt prooviversioonina).	Jah, täielikult tasuta kättesaadav.	Täisfunktsionaalne prooviversioon tasuta kättesaadav, ärilisel otstarbel kasutatav versioon tasuline.	Akadeemiliseks kasutuseks mõeldud versioon tasuta kättesaadav, ärilisel otstarbel kasutatav versioon tasuline.
RQ_RE_NF_3	Järeldusmootor peab olema kasutatav MySQL või PostgreSQL andmebaasimootoritega.	Jah (läbi „RQ_RE_F_1“ rahuldamiseks loodud liidese).	Jah (läbi „RQ_RE_F_1“ rahuldamiseks loodud liidese).	Jah (läbi „RQ_RE_F_1“ rahuldamiseks loodud liidese).

Tabel 11: Valimis osalenud järeldusmootorite vastavus esitatud nõuetele

Järgnevalt on eelmises tabelis esitatud järelusmootorite nõuetele vastavus esitatud arvuliselt (0 – ei vasta, 0,5 – vastab osaliselt, 1 – vastab täielikult). Samuti on juurde toodud peatükis 4.5.1 kirjeldatud nõuete kaalud ning arvutatud nende põhjal kokku lõplik punktisumma.

Nõude ID	Kaal	Drools		OpenRules		Jess	
		Tulemus	Summa	Tulemus	Summa	Tulemus	Summa
RQ_RE_F_1	1	0,5	0,5	0,5	0,5	0,5	0,5
RQ_RE_F_2	0,5	1	0,5	1	0,5	1	0,5
RQ_RE_F_3	0,5	0,5	0,25	0	0	0,5	0,25
RQ_RE_F_4	1	1	1	1	1	1	1
RQ_RE_NF_1	1	1	1	1	1	0,5	0,5
RQ_RE_NF_2	1	1	1	0,5	0,5	0,5	0,5
RQ_RE_NF_3	1	1	1	1	1	1	1
Kokku:			5,25		4,5		4,25

Tabel 12: Valimis osalenud järelusmootorite punktisummad

Kõige rohkem punkte kogus „Drools“, kuna on täielikult tasuta kättesaadav ning võimaldab teatud viisil ka reeglite põhjal koostatud otsingupuud kuvamist. Lisaväärtusena võib veel välja tuua laia arendajate kogukonna ning aktiivse arendustegevuse, mis teisel kahel kandidaadil nii aktiivne ei olnud.

Seega vastas käesoleva töö autori hinnangul töös esitatud nõuetele kõige lähedasemalt reeglisüsteem „Drools“, mis vastab kõigile olulistele esitatud nõuetele ning mida on mõttekas käesolevas töös kavandatava tugisüsteemi juures rakendada.

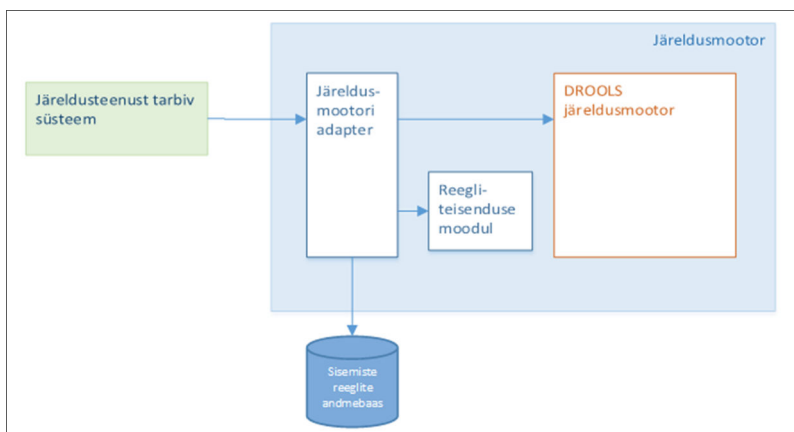
Järgnevalt kirjeldatakse järelusmootori rakendamise kava lähemalt.

4.5.4 Järeldusmootori rakendamise kirjeldus

Valituks osutunud reeglisüsteemi „Drools“ järeldusmootor „Drools Expert“ (edaspidi „järeldusmootor“) võimaldab Java programmeerimiskeeles loodud liidese kaudu reeglite ning faktide järeldusmootorile etteandmist, etteantud faktibaasi peal reeglite rakendamist ning saadud tulemuste kasutamist.

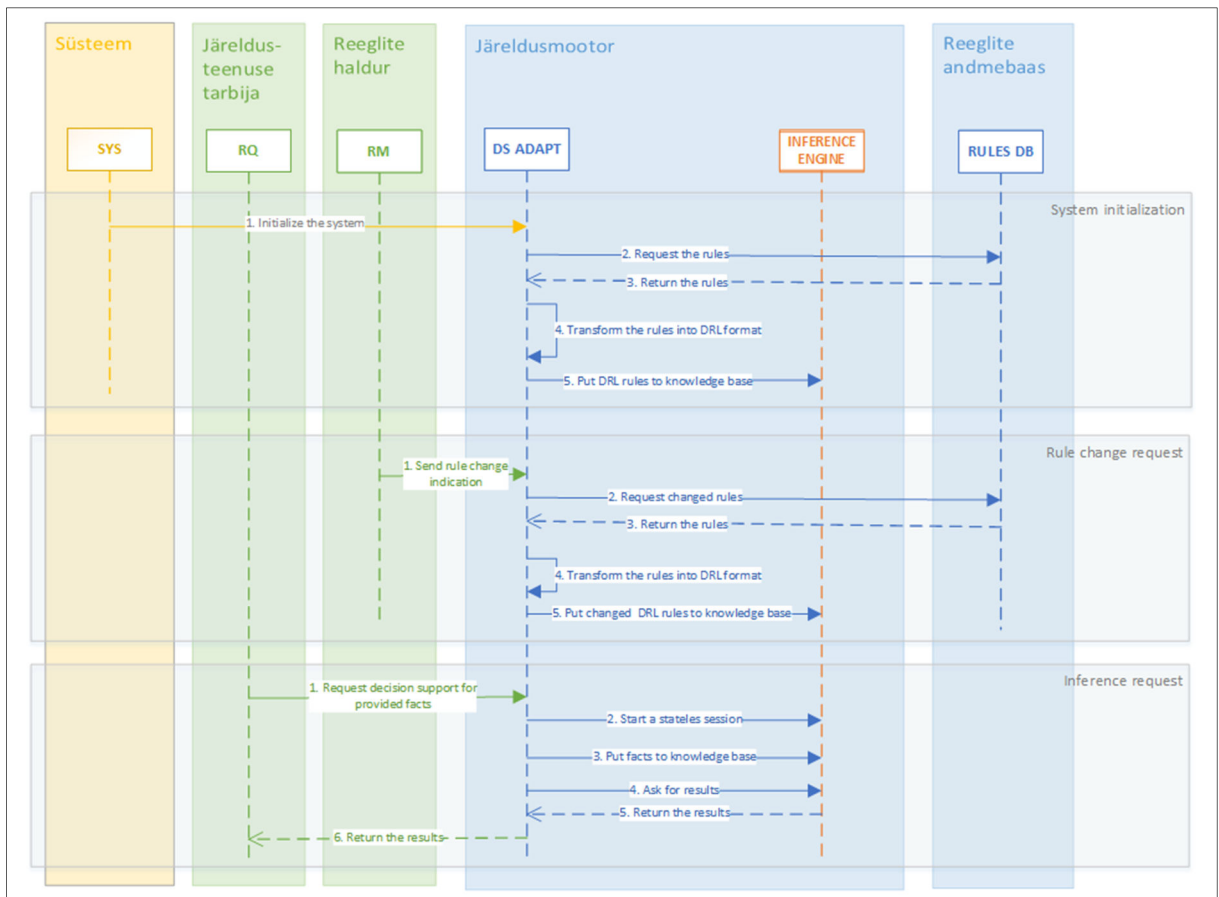
Reegleid on võimalik lugeda nii failisüsteemist, relatsioonilisest andmebaasist kui ka veebiaadressilt, kuid reeglite struktuur peab vastama kindlatele nõuetele. Et käesolevas töös kavandatava tugisüsteemi juures realiseeritakse reeglid süsteemile spetsiifilise andmemudeli kohaselt, tuleb luua funktsionaalsus reeglite sobivale kujule teisendamiseks.

Vastavalt otsustustoe tugisüsteemi arhitektuurile (joonis 7) ning valitud järeldusmootori omadustele on järgneval joonisel 15 kujutatud otsustustoe tugisüsteemi juures kavandatava järeldusmootori arhitektuur, mis koosneb Drools'i poolt pakutavast järeldusmootorist ning süsteemispetsiifilisest adapterist järeldusmootori ning reeglibaasiga suhtlemiseks. Nimetatud adapteri kompetentsi hulka kuulub ka reeglite sobivale kujule teisendamise oskus. Samuti võtab adapter vastu järeldusteenuse päringud ning tagastab saadud tulemused.



Joonis 15: Järeldusmootori arhitektuur

Järeldusmootori kasutamisel näeb käesoleva töö autor kolme kasutusjuhtu: reeglibaasi initsialiseerimine, reegli muudatus juba initsialiseeritud reeglibaasis ning järeldusteenuse päring. Kirjeldatud kasutusjuhtude käigus toimuvad protsessid on näidatud täpsemalt alloleval joonisel 16, mille järel on kasutusjuhte ka täpsemalt kirjeldatud.



Joonis 16: Järeldusmootori protsesside jadadiagramm

Süsteemi initsialiseerimine

Reeglibaasi järeldusmootori mallu lugemine ning Rete otsingupuu koostamine on järeldusmootori jaoks tömahukas protsess. Seetõttu ei tehta seda iga päringu puhul uuesti, vaid paigutatakse otsingupuu süsteemi vahemällu ning kasutatakse päringute puhul korduvalt.

Süsteem edastab järeldusmootori adapterile käivitusteate, misjärel loeb adapter reeglibaasist reeglid, teisendab need järeldusmootorile sobivale kujule, käivitab Rete otsingupuu loomise ning paigutab tulemuse vahemällu.

Reeglite muutmine

Reeglite haldur saab reegleid muuta ka süsteemi töö käigus ning kui muudatused on tehtud, soovib ta need koheselt ka tugisüsteemi juures rakendada. Ei oleks mõttekas hakata seetõttu

kogu järelsmootori otsingupuud algväärtustama, sest see koormaks süsteemi ning häiriks seisakuga kasutajaid. Mõistlik oleks uuendada otsingupuus vaid neid reegleid, mis realselt muutusid ning Drools seda ka võimaldab.

Reeglihaldur edastab järelsmootori adapterile reeglite muutumise indikatsiooni ning annab kaasa ka muutunud reeglite identifikaatorid. Adapter loeb muutunud eksisteerivad reeglid reeglite andmebaasist uuesti sisse ning uuendab vahemälus paiknevat järelsmootori reeglibaasi (uuendades või lisades eksisteerivad reeglid ning eemaldades kustutatud).

Järeldamise päring

Järeldamise käigus rakendab järelsmootor oma reeglibaasi reegleid etteantud faktide peal ning tagastab käivitatunud reeglite poolt loodud tulemused.

Drools võimaldab järeldamist läbi viia kahel viisil: olekuga („stateful“) ning ilma olekuta („stateless“) sessiooni kaudu. (25 lk 112)

Olekuga sessioon võimaldab faktide lisamist ning järelsmootori kasutamist mitme iteratsiooni käigus. Sellist lähenemist saab kasutada näiteks diagnostika või aktsiaturgude analüüsi vallas, kus pidevalt lisandub uusi teadmisi mida järeldamisel arvestada. (25 lk 112)

Olekuta sessiooni puhul antakse kõik faktid iga kord uuesti järelsmootorile ette ning eelmisi fakte ei arvestata. See on lihtsam variant mida võib vaadelda tavalise funktsioonina, kus sisendiks on etteantavad faktid, ärioloogikaks reeglid ning tulemuseks reeglite rakendamisel saadud tulemused. (25 lk 112)

Käesolevas töös kavandatava süsteemi juures realiseeritakse esmalt olekuta sessiooni kasutamine, kuna iga päringu kontekst on erinev. Edaspidi võib analüüsida ka olekuga sessioonide rakendamise mõttekust, kuna ka patsiendi haigusjuhtum on pidevalt lisanduvate faktidega andmestik. See aga tähendaks, et omaette sessiooni tuleks hoida iga patsiendi haigusjuhtumi kohta eraldi.

Kindlasti saaks olekuga sessioone kasutada aga suuremamahulise seire teostamiseks. Näiteks saaks määratleda reeglite kujul mingi kogukondliku ohu (näiteks gripiviiruse) ilmnemise sümptomid, koguda olekuga sessiooni keskkonnas toimuvaid fakte ning seeläbi tuvastada

otsitava sündmuse toimumine. Kuigi sellised kasutusjuhud jäävad käesolevas töös kavandatava süsteemi skoobist välja, võimaldavad realiseeritavad komponendid ka selliste stsenaariumite tuvastamist.

Käesoleva töös kavandatava süsteemi vajadustest lähtuvalt edastab järeldusteenuse soovija järeldusmootori adapterile päringus nimekirja faktidest. Adapter paigutab need järeldusmootori faktibaasi, käivitab reeglite rakendamise ning tagastab järeldusteenuse soovijale reeglite rakendamisel saadud tulemused.

Seda kasutusjuhtu kasutatakse nii ravitöö infosüsteemist otsustustoe pärimisel kui ka reeglite haldusmooduli juures reeglite testimisel.

4.6 Edasiarendamise võimalused

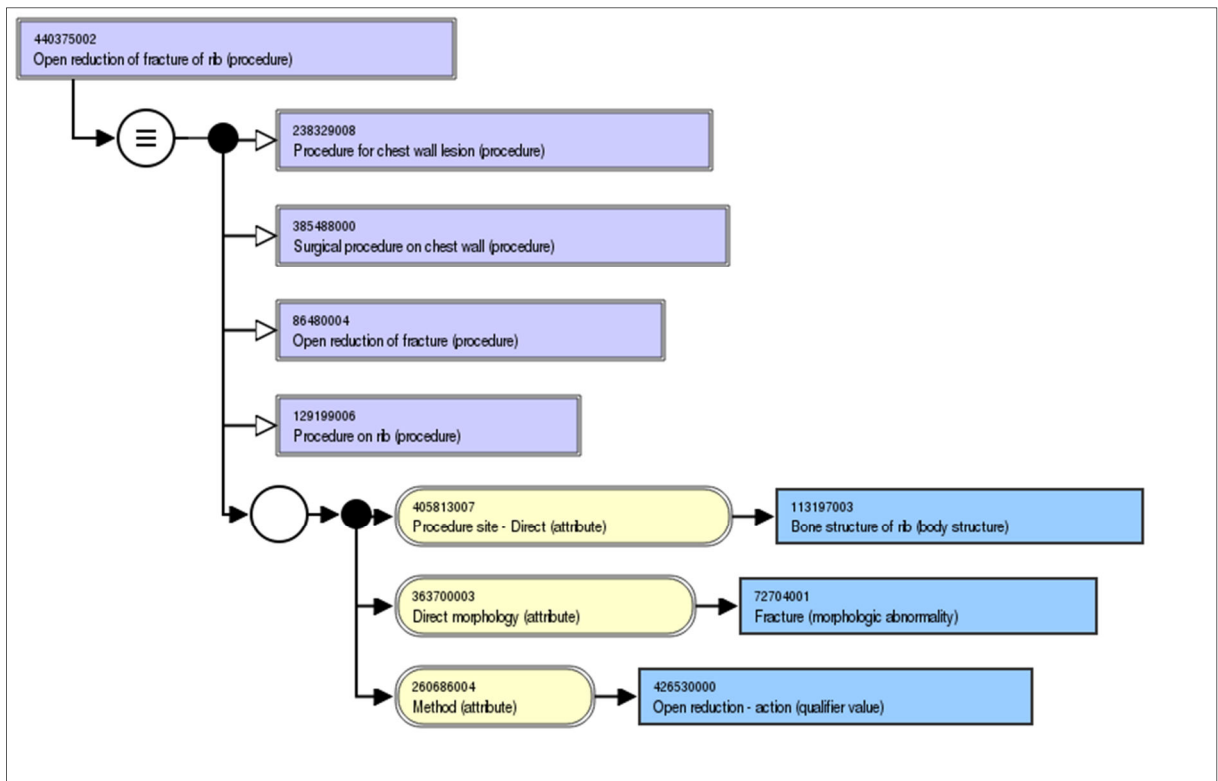
Käesolevas töös kavandatav tugisüsteem on mõeldud otseste ravitöö protsesside toetamiseks ning abistava info pakkumiseks raviotsuste tegemisel. Otsustustoe pakkumiseks vajalik info saadakse eelnevalt defineeritud välistelt andmeallikatelt ning sisemisest järeldusmootoriga seotud reeglibaasist.

Süsteemi edasiarendamiseks on palju võimalusi, mis puudutavad nii informatsiooni omavahelisi seoseid ning tähendust kui ka süsteemi võimekust olemasolevate andmete pealt ise seaduspärasusi otsida ning reegleid luua. Järgnevalt kirjeldatakse kahte käesoleva töö autori arvates potentsiaalset arengusuunda lähemalt.

4.6.1 SNOMED CT terminoloogia rakendamine

Kaugemas tulevikus näeb käesoleva töö autor vajadust kavandada otsustustoe tugisüsteemi andmemudel SNOMED CT terminoloogiast lähtuvalt. SNOMED CT seosed ning andmete erinevad kontekstist sõltuvad tähendused annavad võimaluse seeläbi tekkinud informatsiooni efektiivsemaks kasutamiseks. Samuti tekib võimalus reeglite loomulikumaks kirjeldamiseks

Järgneval SNOMED CT dokumentatsioonist pärineval joonisel on kujutatud üks näide SNOMED CT terminist, mis kujutab protseduuri "ribiluu murru avatud reduktsioon" omadusi (26).



Joonis 17: SNOMED CT termini omadused

Nagu näha, on lisaks protseduuri enda koodile ning nimetusele sellega seostatud ka rida teisi termineid:

- tähenduslik seos: käesolev protseduur on ka rindkere seina kahjustuse protseduur;
- tähenduslik seos: käesolev protseduur on ka kirurgiline rindkere seina protseduur;
- tähenduslik seos: käesolev protseduur on ka murru avatud reduktsioon;
- tähenduslik seos: käesolev protseduur on ka ribiluule teostatav protseduur;
- atribuut: käesoleva protseduuri paige on ribiluu;
- atribuut: käesoleva protseduuri morfoloogiline tunnus on luumurd;
- atribuut: käesoleva protseduuri ravimeetod on avatud reduktsioon.

Kirjeldatud seoste olemasolu võimaldab reegleid defineerida erinevate tunnuste alusel ning erineva detailsusega, kuna süsteem on SNOMED CT terminite kaudu andmeseostest teadlik.

SNOMED CT terminoloogiat haldab ning arendab IHTSDO, mille liige on ka Eesti. Liikmelisus tähendab, et riigis on võimalik SNOMED CT-d ilma litsentsitasudeta rakendada. Samuti saab luua terminoloogia rahvusvahelise versiooni alla riigi kohandatud ning tõlgitud versiooni.

Käesoleva töö kirjutamise hetkel ei ole autorile teadaolevalt veel SNOMED CT Eesti jaoks kohandatud (s.h. tõlgitud) versiooni loodud. Samuti ei kasuta SNOMED CT-d konkreetne ravitöö infosüsteem, mille juures kavandatavat otsustustoe tugisüsteemi on pilootprojekti raames kavas rakendada. Seega lähtuti käesolevas töös otsustustoe tugisüsteemi kavandamisel konkreetse ravitöö infosüsteemi andmemudelist ning seal hetkel kasutatavatest terminoloogiatest (nagu ICD-10 diagnooside puhul jne). Kui edaspidi saab vajalikuks ka SNOMED CT kasutamine, kohandatakse otsustustoe tugisüsteemi andmemudel sellele vastavaks.

4.6.2 Raviandmete põhjal reeglite tuletamine

MeditSiiniasutuse ravitöö infosüsteem sisaldab informatsiooni esinenud haiguste ning nendega seotud andmete vahel. Seotud andmeteks on näiteks varasemad diagnoosid, patsiendi omadused (vanus, sugu, pikkus, kehakaal, elukoht), patsiendi vanemad ning nende omadused ja haigused, haigusjuhtumi käigus kogutud uuringute ja proovide vastused ning palju muud.

Kirjeldatud andmeid oleks võimalik kasutada erinevate andmekaeve meetodite abil assotsiatsioonireeglite leidmiseks, mida siis sobivusel reeglibaasi lisada ning kasutama hakata. Näiteks kui meditsiiniandmete põhjal leitakse, et mingi patsiendi vanusevahemiku, allergia ning diagnoosi koos eksisteerimisel esineb tihti ka mõne teise haiguse oht, saab selle põhjal luua reegli, mis tõseks osutumisel meditsiinitöötajat vastavast ohust teavitab.

Et meditsiin on vastutusrikas valdkond ning vigadel võib olla kõrge hind, tuleks arvutuslikku intelligentsi kombineerida ekspertteadmistega. Samas ei piisa tõelise otsustustoe pakkumiseks ka ainult ekspertteadmistest, kuna inimestel puudub võimekus väga suuri andmemahte regulaarselt analüüsida. Arvutitel on see võimekus olemas.

Reeglite tuletamise kasutusjuhte näeb käesoleva töö autor sellisena, kus erinevaid algoritme kasutades leitakse ravitöö infosüsteemi andmestiku pealt uute reeglite kandidaadid. Need vaatab üle ekspertteadmisi omav isik ning kinnitab sobivad, misjärel kandidaatidest saavad reaalselt kasutatavad reeglid.

Lisavõimalusena saaks otsustustoe tarbijale anda võimaluse reeglite hindamiseks. Näiteks reeglist tuleneva info saamisel saaks saaja lihtsal viisil (kuna keerulist viisi ei hakkaks koormatud meditsiinitöötajad kasutama) määrata, kas see tulemus antud kontekstis oli sobiv

või mitte. Sel moel saadud tagasisidet saaks kasutada näiteks reeglitele prioriteetide määramisel.

Kirjeldatud reeglite tuletamise riskide poole pealt võib suurima riskina välja tuua väärade reeglite tekkimise ohu, mis võivad hakata võimendama erinevate edaspidiste ravivigade teket. Riski esinemise tõenäosus ei ole kuigi suur, kuid mõju võib olla ulatuslik (kui väärade reeglile tuginedes on tehtud uusi väärade raviotsuseid). Riski saaks vähendada ekspertteadmistega isikute poolt pideva süsteemi reeglibaasi haldamise ning jälgimisega.

Kirjeldatud viisil assotsiatsioonireeglite leidmist võimaldab näiteks „Apriori“ nimeline algoritm, mille ühe sarnase kasutuskohana võib välja tuua mudeli koostamise ühes India ülikoolis läbiviidud projektis, millega püütakse varajasesel diagnoosida suus levivate kasvaja esinemist. Vaadeldud on selle haiguse esinemise juhtumeid ning patsiente ning leitud seoseid haiguse ja riskifaktorite vahel (nagu suutubaka kasutamine, suitsetamine, alkoholi tarbimine, hammaste puudumine, kehakaal jms). (27)

4.7 Kokkuvõte

Käesolevas peatükis kirjeldati kavandatavale otsustustoe tugisüsteemile esitatavad nõuded ning viisid nõuete täitmiseks.

Vaadeldi süsteemi kavandatavat arhitektuuri ning komponente ning kirjeldati iga komponendi omadusi ning tööülesannet täpsemalt. Põhilisteks komponentideks olid:

- sisemine reeglibaas oma haldusliideselega;
- integratsioonimoodulid suhtlemaks ravitöö infosüsteemi ning väliste otsustustoe süsteemidega;
- järelmootor sisemise reeglibaasi reeglite põhjal järeluste tegemiseks.

Sisemise reeglibaasi ning haldusliidese osas kirjeldati sisemise reeglibaasi andmemudel ning haldusliidese omadused ja funktsionaalsus. Loodi haldusliidese esmane funktsionaalne prototüüp.

Integratsioonimoodulite kirjelduses esitati sõnumivahetuse protsess ning sõnumites kasutatavad andmemudelid.

Järeldusmootori kirjelduses valiti esmalt kolm võimalikku järeldusmootori kandidaati ning vaadeldi nende omadusi lähemalt. Kirjeldati antud töö vajadustelt lähtuvalt järeldusmootorile esitatavad nõuded ning määrati nõuetele kaalud. Seejärel hinnati kandidaate nõuetest ja kaaludest lähtuvalt ning valiti rakendatavaks järeldusmootoriks ärireeglisüsteemi „Drools“ koosseisu kuuluv mootor „Drools Expert“. Viimaks kirjeldati valitud mootori rakendamist käesolevas töös kavandatavas süsteemis.

Viimaks kirjeldati mõnda käesolevas töös kavandatava süsteemi edasiarendamise võimalust, mis antud töö autori hinnangul võiksid realiseeruda.

5. Otsustustoe tugisüsteemi ning ravitöö infosüsteemi integratsiooni kirjeldus

Kavandatava otsustustoe tugisüsteemi kasutuselevõtuks on vajalik see integreerida ravitöö infosüsteemiga, et pakkuda ravitöö infosüsteemi toetatavate tööprotsesside juures kasutajale otsustustuge.

Käesolevas peatükis kirjeldatakse üldisel tasemel mõningaid ravitöö infosüsteemi toetatavaid tööprotsesse ning võimalusi nende juures otsustustoe tugisüsteemi kasutamiseks.

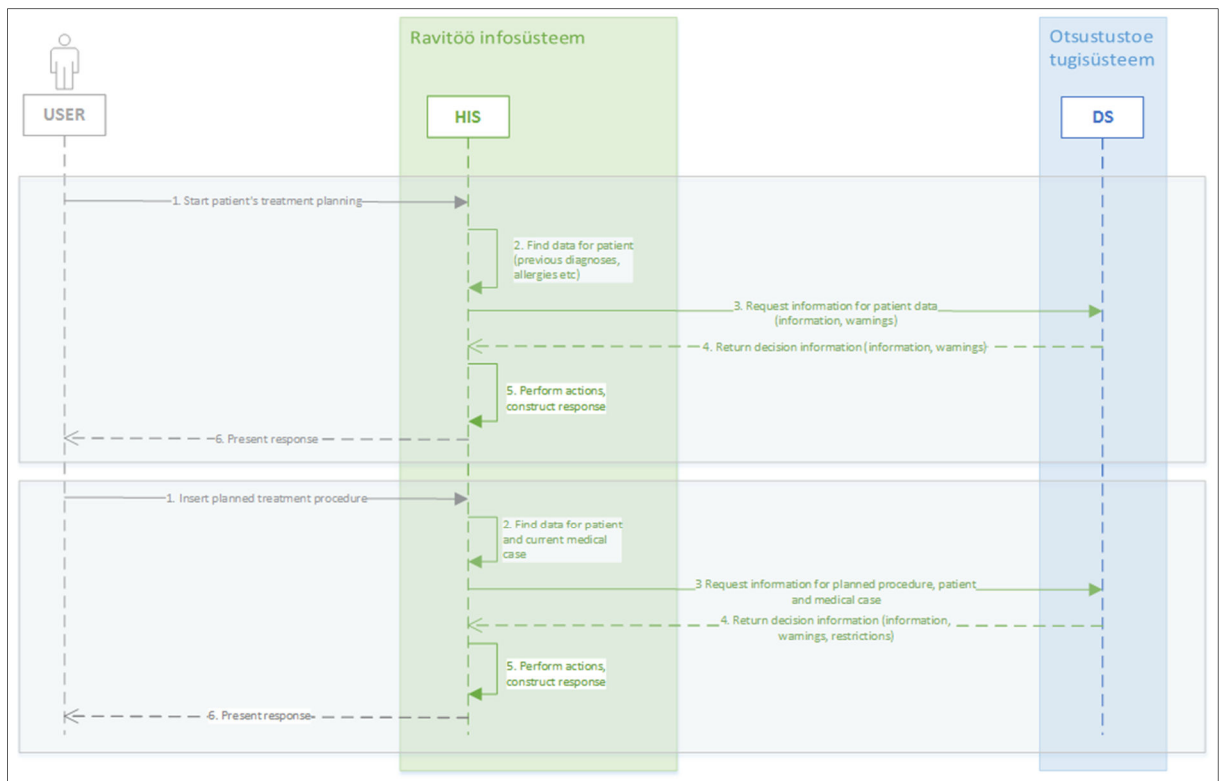
Kindlasti on äärmiselt oluline ka viis, mil moel otsustustoe tugisüsteem ravitöö infosüsteemi juurde integreerida. Seda eeskätt just kasutajakogemuse vaatepunktist. Eesmärgiks peaks olema, et kasutajad saaksid aru otsustustoe rakendamisest saadavast kasust ning oleksid ise huvitatud selle rakendamisest. Otsustustoe tugisüsteemi integreeritus ei tohiks kasutajale tööd juurde tuua. Otsustustoe päringud peaksid käivituma ärioloogiliselt sobival hetkedel ning tulemuste esitamine peaks olema kasutaja jaoks loogiline ning mitte liialt häiriv. Samuti on oluline, et juurutusfaasis oleks võimalik saada kasutajatelt võimalikult palju tagasisidet.

Käesoleva töö autori hinnangul oleks vajalik esmases rakenduskohas (piloohaiglas) luua otsustustoe tugisüsteemi juurutamise töörühm, kuhu kuuluksid tarkvaraarendaja, haigla IT-osakonna ning otsustustoe rakendamisest huvitatud meditsiinipersonali esindajad. Töörühma ülesandeks oleks määratleda otsustustoe integreerimise etapid ning järjekord, analüüsida kasutajatelt tulevat tagasisidet ning kavandada võimalikke vajalikke muudatusi.

Juurutusprotsessi ei ole käesolevas töös detailsemalt käsitletud, kuid selle kavandamine ning läbimõeldud rakendamine on käesoleva töö autori hinnangul süsteemi eduka kasutuselevõtu kohalt äärmiselt suure tähtsusega.

Järgnevalt on kirjeldatud kaks ravitöö infosüsteemi abstraktset kasutusjuhu näidet, mille juures on kasutatud otsustustoe tugisüsteemi päringuid. Keskendutud on just käivituvatele protsessidele. Otsustustoe tugisüsteemi protsesside sisemist ärioloogikat on detailsemalt kirjeldatud käesoleva töö peatükis nr 4. Ravitöö infosüsteemi sisemise ärilogika detailsem kirjeldus jääb käesoleva töö skoobist välja, kuid ravitöö infosüsteemil on võimalus saadavat otsustustoe infot rakendada vastavalt kasutaja vajadustele ning sobivaimal viisil.

Käesolevas peatükis kirjeldatud kasutusjuhtude protsesse on kujutatud ka alloleval jadadiagrammil. Kirjeldatud kasutusjuhud ning protsesside skeem (joonis 18) on koostatud lähtudes süsteemide arhitektuurilisel skeemil (joonis 1) toodud interaktsioonidest ning käesoleva töö autori kogemustest ühe konkreetse ravitöö infosüsteemi tööprotsesside osas.



Joonis 18: Otsustustoe tugisüsteemi ning ravitöö infosüsteemi integratsiooniprotsesside jadadiagramm

Patsiendi ravi planeerimise alustamise kasutusjuht

Ravi planeerimise käigus koostab ravi planeeriv meditsiinitöötaja patsiendi ravijuhtumi plaani – määrab ajateljel vajalikud diagnostilised ning võimalusel ka juba ravitoimingud. Ravi plaani täiendatakse ravijuhtumi käigus korduvalt.

Ravi planeerimise alustamisel on oluline olla teadlik patsiendi eelnevatest terviseprobleemidest ning iseärasustest. Seetõttu pärib ravitöö infosüsteem kasutajalt vastava korralduse saamisel oma andmestikust vajalikud patsiendi varasemad andmed ning edastab need otsustustoe päringus otsustustoe tugisüsteemile. Otsustustoe tugisüsteem leiab sisemist reeglibaasi ning integreeritud väliseid tugisüsteeme kasutades faktidele vastava tulemuse (otsustustoe info) ning tagastab selle päringu vastusena ravitöö infosüsteemile. Ravitöö

infosüsteem töötleb seejärel saadud tulemust endale vajalikul viisil ning tõenäoliselt esitab selle ka kasutajale.

Antud kasutusjuhuses tagastatava otsustustoe päringu vastuses võib olla näiteks:

- hoiatus patsiendi allergiate kohta;
- hoiatus hetkel aktiivselt manustatavate oluliste ravimite kohta;
- info võimalike ravimi koostoimete kohta;
- info ravi- ning diagnostikaprotseduuride soovitude kohta;
- patsiendi varasematest diagnoosidest sõltuv info antud diagnoosiga enamjaolt kaasnevate raviprotseduuride kohta.

Planeeritud raviprotseduuri sisestamise kasutusjuht

Konkreetsel raviprotseduuri sisestamisel valib kasutaja protseduuride loendist ühe elemendi ning määrab vajadusel teostamisega seonduvat lisaandmed (teostaja, teostamise aja, kommentaari jms).

Ka siin saab olla vajadus otsustustoe info järele, kuna sisestatava protseduuriga seoses võib otsustustoe tugisüsteemi andmekogudes olla infot, millest kasutajal võib abi olla.

Peale kasutajapoolset raviprotseduuri valikut leitakse sarnaselt eelmisel kasutusloole info patsiendi muude raviandmete kohta ning edastatakse see info koos valitud protseduuriga otsustustoe tugisüsteemile. Otsustustoe tugisüsteem leiab enda andmekogudest päringus saadud andmetele vastavad vastused ning tagastab need pärivale süsteemile. Ravitöö infosüsteem töötleb saadud infot ning kasutab seda sobival viisil (näiteks informeerib kasutajat või keelab antud protseduuri määrata).

Kirjeldatud kasutusjuhuses tagastatava otsustustoe päringu vastuses võib olla näiteks:

- info valitud protseduuridega tihti koos esinevate teiste protseduuride kohta;
- hoiatus valitud protseduuri ning patsiendi diagnoosiga kaasneva võivate tüsistuste kohta;
- keeld patsiendile antud protseduuri määramiseks (näiteks püütakse meessoost patsiendile määrata rasedusega seonduvat protseduuri);
- valitud protseduuriga seonduvat ravisovitusi.

6. Kokkuvõte

Käesoleva magistritöö eesmärgiks on töötada välja kava otsustustoe tugiinfo kasutuselevõtuks ühe ravitöö infosüsteemi juures, rakendades selleks otsustustoe tugisüsteemi. Sealjuures on vajalik analüüsida süsteemile esitatavaid nõudeid, kavandada süsteemi arhitektuur koos tööprotsessidega ning kirjeldada viisid nõuetele vastava süsteemi loomiseks ja vajalike tööprotsesside toetamiseks. Samuti on vajalik vaadelda olemasolevaid süsteeme ning nende kasutamise võimalusi.

Ravitöö infosüsteemina on vaadeldud ühte konkreetset Eesti meditsiinasutustes kasutatavat infosüsteemi, millega töö autoril on olnud tööalased kogemused, kuid käesoleva töö tulemused on tööprotsesside sarnasuse tõttu kasutatavad ka teiste samalaadsete ravitöö infosüsteemide juures.

Eesti tervishoiu IT-arenduste rahastus jääb alla Euroopa komisjoni soovitatud tasemest. Sellest johtuvalt on olemasolevate otsustustoe terviklahenduste kasutuselevõtt nende hindu arvestades keeruline. Samuti on Eesti riik asunud mõningaid otsustustoe andmebaase lokaliseerima ning riiklike vajadustega vastavusse viima, mistõttu on Eestis rakendatava süsteemi puhul vajalik ka liidestusvõimekus konkreetsete süsteemidega. Lisaks eksisteerib vajadus raviastutuse tasemel otsustustoe reeglite määramiseks, mida konkreetse asutuse tööprotsesside juures rakendada. Seetõttu on autor analüüsi tulemusena jõudnud järeldusele, et otstarbekas on luua kohalikele vajadustele vastav otsustustoe tugisüsteem, mis võimaldab nii välise otsustustoe andmeallikate kui ka sisemise, meditsiinasutuse enda hallatava reeglibaasi kasutamist.

Käesoleva töö põhitulemus on otsustustoe tugisüsteemi analüüsi projekt, milles on vastavalt nõuetele:

- kirjeldatud välise otsustustoe andmeallikate andmemudelid;
- kirjeldatud sisemise reeglibaasi andmemudel ning haldusliidese prototüüp;
- loodud andmeallikate andmemudelite ühildamise kava;
- kirjeldatud süsteemi arhitektuur, põhiprotsessid ning sõnumivahetuse andmemudelid;
- teostatud sisemise reeglibaasi järeldusmootori valik ning kirjeldatud järeldusmootori rakendamise kava;

- kirjeldatud otsustustoe tugisüsteemi ning ravitöö infosüsteemi integreerimise põhimõtted.

Töö eesmärgi võib lugeda saavutatuks, kuna nõuetele vastava otsustustoe tugisüsteemi kirjeldus on loodud. Töö autor koostas isiklike teadmiste, materjalidest leitud informatsiooni ning tööalaste kogemuste põhjal otsustustoe tugisüsteemi kirjelduse ning andis ettepanekud kirjeldatud süsteemi ravitöö infosüsteemiga integreerimiseks. Ettepanekutele vastav integratsioon on ühe Eestis kasutatava ravitöö infosüsteemi juures kavas realiseerida.

Töös kirjeldatud teemat saab edasi arendada mitmel suunal.

Üheks neist saaks olla kirjeldatud otsustustoe tugisüsteemi juurutusprotsessi analüüs, mille eesmärk on otsustustoe tugisüsteemi edukas kasutuselevõtt, kasutajate positiivne tagasiside ning parendusettepanekute saamine.

Teiseks suunaks saaks olla süsteemile iseõppimisvõime lisamine, kasutades sisemise reeglibaasi reeglite loomiseks erinevaid olemasoleva meditsiiniinfo peal töötavaid algoritme (näiteks assotsiatsioonireelgite leidmist).

Summary

IT-service has been part of the medical institutions in Estonia for more than 20 years. During this time a considerable amount of information has been stored in the databases of those institutions. This information can be used to support medical personnel in their everyday work.

Also, with the IT-world evolving rapidly, the medical personnel has to deal with increasing amount of information overload, from which they need to gather information to make the right medical decisions. IT-services could help them – to process data and filter out only the concerned information.

The goal of the present thesis is to provide a way to use clinical decision support information in supporting the process of making medical decisions. A description of the medical information support system is created to provide this possibility. The described support system is intended to be integrated with a clinical information system already used in a medical institution.

As a result of this thesis, an analysis project was created, which:

- describes the data models of the external decision information databases used;
- describes the data model of the internal rule database and its management system;
- describes the plan to integrate different data models of the systems concerned;
- describes the architecture, the business processes and the models of integration messages of the planned system;
- conducts the comparison and selection of the inference engine to be used with the inner base of rules as well as the plan to integrate the selected inference engine with other parts of the system;
- describes the integration of the planned decision support system in the processes of the clinical information system.

The goal of the thesis is achieved as the analysis project has been created to allow the decision support system to be used within a medical institution, by implementing a decision support

system. The actual implementation is also being planned within one medical institution in Estonia.

This topic can be expanded further in separate directions.

One of them is to plan and analyze the implementation process of the decision support system. The aim of this plan would be to conduct a successful implementation of the new system and to receive the much-needed feedback of the actual medical personnel using the system.

The other is to develop the system further and introduce machine-learning algorithm for creation of the rules into the systems internal rule base, by using the information gathered in the database of the clinical information system. This allows to take advantage of the knowledge and experiential information gathered over years by the use of the clinical information system.

Kasutatud kirjandus

1. **Tepandi, Jaak.** *Intelligentsed Süsteemid, loengukonspekt.* s.l. : TTÜ Informaatikainstituut, 2014.
2. **IHTSDO.** Snomed CT - Starter Guide. [Võrgumaterjal] 2015. a. www.ihtsdo.org/resource/resource/41.
3. **Wikipedia.** Application programming interface. [Võrgumaterjal] 2016. a. https://en.wikipedia.org/wiki/Application_programming_interface.
4. **HL7 International.** Health Level Seven International. [Võrgumaterjal] 2016. a. <http://www.hl7.org/about/index.cfm?ref=nav>.
5. *Kuidas kiirendada Eesti tervishoid infoajastusse.* **Ain Aaviksoo, Janek Saluse, Gerli Paat.** 6, 2010. a., Praxis Poliitikaanalüüs.
6. **Eesti E-Tervise Sihtasutus.** Dokumentide statistika. [Võrgumaterjal] 2015. a. <http://www.e-tervis.ee/index.php/et/dokumentide-statistika>.
7. **eMeditsiini Laboratoorium, Tehnomeedikum, Tallinna Tehnikaülikool.** *Eestikeelse kliiniliste otsuste tugisüsteemi välja arendamise kava tervise infosüsteemide jaoks.* Tallinn : s.n., 2014.
8. **Inc, Truven Health Analytics.** Micromedex Clinical knowledge. [Võrgumaterjal] 2012. a. http://micromedex.com/portals/1/assets/ClinicalKnowledge_LOW.pdf.
9. **Archimedes Clinical Analytics, Inc.** Archimedes IndiGO. [Võrgumaterjal] Archimedes Clinical Analytics, Inc., 2015. a. <https://archimedesmodel.com/indigo>.
10. **Elsevier.** Clinical Key. [Võrgumaterjal] 2015. a. <http://www.clinicaldecisionsupport.com/solutions/clinical-reference/clinical-key/>.
11. **Medicine, Map of.** Map of Medicine. [Võrgumaterjal] 2015. a. <http://mapofmedicine.com>.

12. **Duodecim Medical Publications Ltd.** EBMeDS. [Võrgumaterjal] Duodecim Medical Publications Ltd., 2015. a. www.ebmeds.org/web/guest/home?lang=en.
13. **Medbase Ltd.** Sfinx. [Võrgumaterjal] Medbase Ltd., 2015. a. <http://www.medbase.fi/en/professionals/sfinx>.
14. **Sihtasutus, Eesti E-Tervise.** *Otsustustoe liidese spetsifikatsioon.* 2010. a.
15. **Healthcare, Celsius.** *Ravimi koostoitete liidese spetsifikatsioon.* 2015. a.
16. **Ravijuhend.** Ravijuhend. [Võrgumaterjal] 2015. a. <http://www.ravijuhend.ee>.
17. **Eesti Haigekassa.** Eesti 2. tüüpi diabeed juhend. [Võrgumaterjal] 2008. a. https://www.haigekassa.ee/uploads/userfiles/Diabeedijuhend_33%20Rajasalu%2013%2007%2008.pdf252007%252008.pdf&usg=AFQjCNEls9Dx.
18. **Giordano, Anthony David.** *Data Integration Blueprint and Modelling.* s.l. : IBM Press, 2011.
19. **Wikipedia.** Rete algorithm. [Võrgumaterjal] 2016. a. https://en.wikipedia.org/wiki/Rete_algorithm.
20. **Ary, Jeremy.** *Instant Drools Starter.* s.l. : Packt Publishing, 2013.
21. **The Apache Software Foundation.** Apache License, Version 2.0. [Võrgumaterjal] 2016. a. <http://www.apache.org/licenses/LICENSE-2.0>.
22. **OpenRules Inc.** Open Rules. [Võrgumaterjal] 2016. a. <http://www.openrules.com/components.htm>.
23. **Sandia National Laboratories.** Jess Rule Engine. [Võrgumaterjal] 2015. a. <http://www.jessrules.com/jess/>.
24. **Friedman-Hill, Ernest.** *Jess in Action, Rule-Based Systems in Java.* s.l. : Manning, 2003.
25. **Bali, Michal.** *Drools JBoss Rules 5.0 Developer's Guide.* 2009.
26. **IHTSDO.** SNOMED CT Browser. [Võrgumaterjal] <http://browser.ihtsdotools.org/?perspective=full&conceptId1=440375002&edition=en->

edition&release=v20150731&server=https://browser-aws-
1.ihtsdotools.org/&langRefset=900000000000509007.

27. **Neha Sharma, Hari Om.** Extracting Significant Patterns for Oral Cancer Detection Using Apriori Algorithm. [Vörgumaterjal] 2014. a.
<http://www.scirp.org/journal/PaperInformation.aspx?paperID=43830>.

28. **Jérôme Boyer, Hafedh Mili.** *Agile Business Rule Development*. 2011.

29. **Amador, Lucas.** *Drools Developer's Cookbook*. s.l. : Packt Publishing, 2012.

Lisa 1 - Sisemise reeglibaasi haldusliidese ekraanivormide prototüübid

Restriction
KEELD rasedustoimingud raseduseks sobimatule

OR

- not ((patient gender) = FEMALE)
- not (10 <= [patient age in years] <= 90)

OR

- [healthcare service] = 384
- [healthcare service] = 2930L
- [healthcare service] = 383
- not (([healthcare service] = 011104)
- [healthcare service] = 041102
- [healthcare service] = A41
- not (([healthcare service] = 9140)
- [healthcare service] = 3780
- [healthcare service] = 378N
- [healthcare service] = 7947
- [healthcare service] = V41

Sisemise reeglibaasi haldusliidese reeglite nimekirja vaate prototüüp

Edit rule

Type: Restriction

Descr: Recommendation

KEEL Restriction

rased Information

Save

Sisemise reeglibaasi haldusliidese reegli vaate prototüüp

Edit restriction group

Operation: AND

Save AND OR

Sisemise reeglibaasi haldusliidese reegli tingimuste grupi vaate prototüüp

Edit restriction

Negative

Gender equals: Naine

Save

Sisemise reeglibaasi haldusliidese isiku soolise tingimuse vaate prototüüp

Edit restriction

Negative

Years greater or equal to: 10

Years less or equal to: 90

Save

Sisemise reeglibaasi haldusliidese isiku vanuselise tingimuse vaate prototüüp

Edit restriction

Negative

Scope: Patient

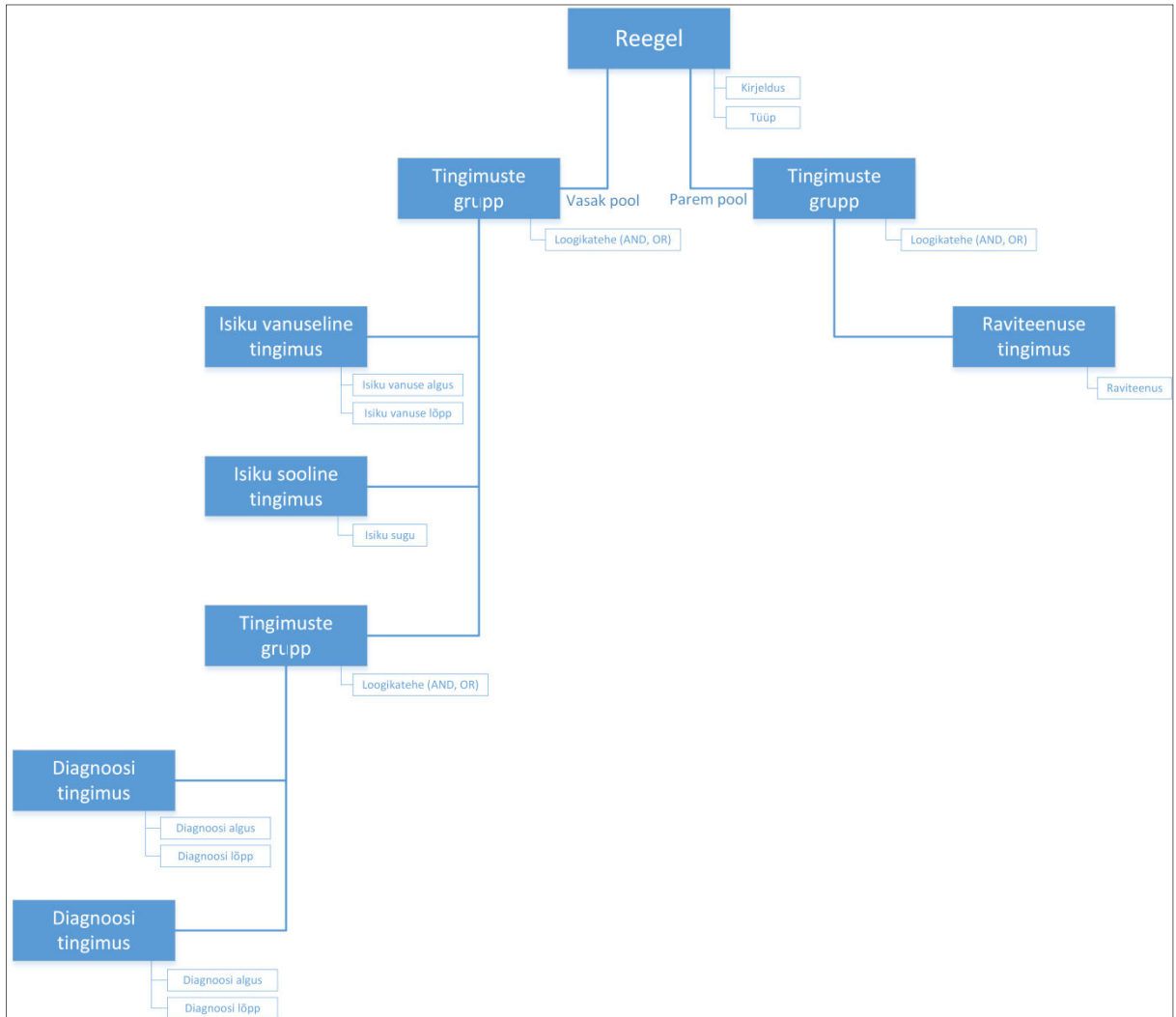
Diagno: Patient or equal to S23 Medical case

Diagnosis code less or equal to: S33

Save

Sisemise reeglibaasi haldusliidese diagnoosi tingimuse vaate prototüüp

Lisa 2 - Sisemise reeglibaasi reegli andmekoosseisu näide



Kui sellise reegli tüübiks oleks "soovitus", siis võiks reeglit lugeda järgmiselt: "kui vasaku poole tingimused osutuvad tõesteks, siis soovita ühte konkreetset raviteenust".