



TALLINNA TEHNIKAÜLIKOOL
INSENERITEADUSKOND
Tartu kolledž

**MEDIPOST INFOVAHETUSTEENUSE
LIIDESTAMINE 5DVISION OÜ
TARKVARAPROJEKTIDEGA**

**DEVELOPING MEDIPOST INTEGRATION DATA MOUDLE
FOR 5DVISION SOFTWARE PROJECTS**

diplomitöö

Üliõpilane: Tanel Tiisler
Üliõpilaskood: 166148
Juhendajad: Kristjan Adojaan
Merik Meriste

Tartu 2022

AUTORIDEKLARATSIOON

Olen koostanud lõputöö iseseisvalt. Lõputöö alusel ei ole varem kutse- või teaduskraadi või inseneridiplomit taotletud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

20. mai 2022

Autor: / allkirjastatud digitaalselt /

Töö vastab bakalaureusetöö/magistritööle esitatud nõuetele

20. mai 2022

Juhendaja: / allkirjastatud digitaalselt /

Kaitsmisele lubatud

".....".....2022 .

Kaitsmiskomisjoni esimees

.....

/ nimi ja allkiri /

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks¹

Mina Tanel Tiisler (sünnikuupäev: 15.08.1997)

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „Medipost infovahetusteenuse liidestamine 5DVision OÜ tarkvaraprojektidega“, mille juhendajad on Kristjan Adojaan ja Merik Meriste ,

1.1 reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;

1.2 üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.

2.Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.

3.Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

¹Lihtlitsents ei kehti juurdepääsupiirangu kehtivuse ajal, välja arvatud ülikooli õigus lõputööd reprodutseerida üksnes säilitamise eesmärgil.

/allkirjastatud digitaalselt/

20. mai 2022 (kuupäev)

Taltech Tartu Kolledž

LÕPUTÖÖ ÜLESANNE

Üliõpilane: Tanel Tiisler, 166148 NDFR

Õppekava, peeriala: NDFR14/15 – Küberfüüsikaline süsteemitehnika

Juhendajad: Kristjan Adojaan ja Merik Meriste

Lõputöö teema:

Medipost infovahetusteenuse liidestamine 5DVision OÜ tarkvaraprojektidega

Developing Medipost integration module for 5DVision software projects

Lõputöö põhieesmärgid

1. Luua moodul andmevahetuseks Medipostiga
2. Testida seda moodulit andmebaasis

Lõputöö etapid ja ajakava

| Nr | Ülesande Kirjeldus | Tähtaeg |
|----|---|----------|
| 1 | Medipost dokumentatsiooni läbitöötamine | 28.02.22 |
| 2 | Lokaalne PHP põhine süsteem | 18.03.22 |
| 3 | Esmane versioon meditsiinilises andmebaasis | 25.04.22 |

Töö keel: Eesti keel

SISUKORD

| | |
|--|----|
| EESSÕNA | 6 |
| SISSEJUHATUS | 7 |
| 1 ANDMEVAHETUSPLATVORMID | 8 |
| 1.1 X-tee | 8 |
| 1.2 Medipost | 9 |
| 2 SUHTLUS MEDIPOSTIGA | 10 |
| 2.1 Teenused | 10 |
| 2.2 Tellimus | 10 |
| 2.3 Vastus | 11 |
| 2.4 Teade | 11 |
| 2.5 Üldine andmevahetus | 11 |
| 3 EESMÄRGID | 14 |
| 3.1 Funktsionaalsed Nõuded | 14 |
| 3.2 Mittefunktsionaalsed nõuded | 14 |
| 4. MEDIPOST KASUTAMINE ANDMEBAASIDES | 15 |
| 4.1 Sõnumite lugemine | 16 |
| 4.2 Tagastatav väärtus | 16 |
| 5. SÕNUMI KOOSTAMINE | 18 |
| 5.1 Struktuur | 18 |
| 5.2 Ühisosa | 20 |
| TESTIMINE JA TULEMUSED | 22 |
| XML | 22 |
| KOKKUVÕTE | 24 |
| SUMMARY | 25 |
| KASUTATUD KIRJANDUSE LOETELU | 26 |

EESSÕNA

See lõputöö teema on välja pakutud tarkvaraarendusettevõtte 5DVision OÜ poolt. Firma tegevusvaldkondade hulka kuulub mitme meditsiiniettevõtte tarkvarasüsteemide haldamine, mistõttu oli vajadus mõne projekti liidestamise võimekuseks Medipost andmevahetusteenusega firma pikemaajalistes plaanides sees.

Konkreetselt lõputöö projekti katsetati peamiselt Asper Biogene kliendihaldustarkvara arendusversioonis, kuid laiem eesmärk oli luua moodul nii, et seda saaks võimalikult lihtsalt kasutada erinevate projektide koosseisus.

Lõputöö valmis koos erialapraktikaga. Seejuures oli lõputöö eesmärk konkreetse projekti kavandamine ning arendamine PHP keskkonnas. Seda toetava erialapraktika eesmärk oli tööks ja selle testimiseks vajalike tarkvarade valimine ning andmevahetuse korrektsuse testimiseks sobilike algandmete leidmine.

SISSEJUHATUS

Tarkvaraprojektid vajavad töötamiseks eelnevalt programmeeritud koodibaasi ja juurdepääsu tööks vajalikele salvestavatele andmetele. Lihtsamatel juhtudel võivad andmed olla lihtsalt küpsised kasutaja arvutis, kuid üldiselt on selleks kasutusel eraldi andmebaas andmete pikaajaliseks salvestamiseks.

Konkreetselt meditsiiniettevõtetes on suurt hulka andmeid vaja näiteks nii patsientide olemasolevate terviseandmete kui ka vajaduspõhiselt tehtavate uute uuringute jaoks [1]. Seejuures võib erinevatel asutustel olla väga erinev võimekus andmete kogumisel. Mõned uuringud on piisavalt spetsiifilised, et neid on võimalik teha kas majanduslikel põhjustel või vastava kogemusega tööjõu puudumisel vaid üksikutes kohtades. Samas aga eeldab tänapäevane meditsiin, et diagnoosi tegemisel oleks kasutusel kogu võimalik info.

Seetõttu läheb vaja hajutatud andmesüsteemi, kus ettevõtted suudaksid vajaduspõhiselt olla ühenduses ka teiste asutuste andmebaasidega. See tähendab valmisolekut nii oma andmete jagamiseks teistega kui ka ise sobivate päringute tegemist. Tavaliselt ei ole aga erinevate ettevõtete poolt hallatavate tarkvaraprojektide vahel otseühendust. Samuti võib andmete vahetamisel tekkida probleeme, kui pole kokkulepitud ühest struktuuri: milline päring vastab millistele andmetele ja kuidas kontrollida teiste õigusi päringu tegemiseks.

Seetõttu on loodud spetsiifilisi andmevahetusserverid, mis vahendavad ja vajadusel ka valideerivad infot kahe või enama andmebaasi vahel. Selleks peavad olema sellistel serveritel olemas üsna selgelt etteantud nõuded: kes ja miks võib teenusega liituda ja milliseid andmeid/millises formaadis seal vahetada saab. Seetõttu nõuab mõne sellise serveri liidestamine olemasoleva projektiga suuremat eeltööd.

Selle lõputöö eesmärk on seotud konkreetselt Medipost infovahetamise teenusega. Täpsem vajadus on arendada ühekordselt välja selline klientprogramm, mida saaks hiljem vajaduspõhiselt kasutada erinevate tarkvaraprojektide liidestamiseks Medipost serveriga, ilma et see eeldaks projekti suuremat muutmist.

1 ANDMEVAHETUSPLATVORMID

Medipost on peamiselt Eesti meditsiinasutuste jaoks mõeldud informatsiooni jagamise teenus. Selle peamiseks ülesandeks on laborite ning muude meditsiinasutuste vaheliste analüüside tellimuste ning nende vastuste vahendamine[2]. Lisaks on neil olemas eraldi osa, mida kasutatakse e-arvete saatmiseks[3]. Sisuliselt on tegu sama teenusega, kuid neid käsitletakse eraldi, kuna kasutajaskond on mõnevõrra erinev. Selles lõputöö kontekstis räägitakse Medipostist kui analüüside vahendamise teenusest.

Selleks, et oleks võimalik hinnata, kas konkreetsel (meditsiini)ettevõttel oleks vaja Mediposti kasutada, on vaja teada, millised alternatiivsed keskkonnad on Eestis kasutusel ning mille poolest need erinevad.

1.1 X-tee

X-tee on peamiselt Eesti (riigi)asutuste vahelise andmevahetuse turvalisemaks muutmise teenus, mida kasutab üle 600 [4] ettevõtte. X-tee-l ei ole konkreetset kasutusvaldkonda, vaid peaks sobima kõikvõimalikuks infoedastuseks. Sellegipoolest on keskkonna surimad kasutajad just meditsiinivaldkonnas tegutsevad ettevõtted.

Sellega liitumiseks tuleb esitada põhjendatud taotlus ning hoolitseda oma andmebaasi turvalisuse eest. See tagab siiski vaid õiguse näha, kes veel antud teenust kasutavad. Reaalselt kellegi infosüsteemi päringu tegemiseks oleks vaja sõlmida konkreetse kasutajaga eraldi leping.

Veelgi keerulisem on olla ise andmete jagaja. Selleks tuleb paigaldada endale turvaserver [4] ja tõendada oma andmebaaside turvalisust läbi Riigi Infosüsteemide Haldussüsteemi menetluse. See võimaldab ka suurt osa võimalikest päringutest eelgenereerida. Päringute tegemise ajaks loodakse eraldi ajutine tunnel-ühendus andmebaaside vahel. Sellega saab tagada, et andmed ei satuks korrakski nende omanike valdusest välja.

Täiendavaks turvalisuseks on võimalik lisada igalt kasutajalt nõuda mitmetasandilist autentimist ning logida kõikide kasutajate kõiki tegevusi. Seetõttu sobib X-tee kõige rohkem just sellistele asutustele, kus kõige olulisem on andmete turvalisus. Samuti on neil paigas kindlad reeglid, kellelt tohib infot saada.

Nendel põhjustel on X-tee ajalooliselt olnud peamiselt kasutusel riigiasutustes. Siiski on aja jooksul loodud ka keskkonnaga liitumiseks loodud lisatarkvara MISP [5], mis

muudab ühinemise oluliselt lihtsamaks ettevõtetele, kellel pole ressursse oma tarvara vajalikul määral ümber ehitada.

X-tee abil on loodud näiteks tervise infosüsteem(digilugu). Kuna tegemist on tarkvaraga, mis sisaldab palju tundlikke isikuandmeid, ja seda kasutab palju inimesi, siis on mõistlik seda just X-tee kaudu arendada.

1.2 Medipost

Medipost on AS Medisoft poolt loodud infovahetusteenus meditsiini valdkonnas tegutsevatele ettevõtetele. Sellega ühinemiseks tuleks sõlmida Medisoftiga leping, mille tagajärel saadakse oma asutusele kasutajatunnus ja parool, mis kehtivad korraga kõigile sama asutuse töötajatele.

Medipost on mõeldud peamiselt laboritele ja nende klientidele elektroonilise uuringute ja analüüside tulemuste vahendamise teenusena.

Mediposti serverisse saavad autoriseeritud kasutajad sõnumeid saata, lugeda või kustutada. Kõik saadetud sõnumid salvestatakse serveri andmebaasi, kust neid saab vastavat ID numbrit teades alla laadida.

Mediposti idee on olla maksimaalselt kasutatav. See peaks olema kasutatav võimalikult paljudel erinevatel platformidel ja selle halduskulud peavad olema võimalikult väikesed. Seetõttu pole sellel nõuet turvaserveri kasutamiseks ja kogu infovahetus toimib üle tavalise HTTP protokolliga.

Seega on medipost mõeldud eelkõige sellistele asutustele, kus olulisimaks prioriteediks on teenuse kõrge kättesaadavus asutuse kõikidele töötajatele. Peamised kasutajad on meditsiinilaborid ja nende kliendid, kes vajavad elektroonilist analüüside tulemuste edastamise võimalust. Näiteks on Medipost kasutusel koos perearstide infosüsteemi ja veebiregistratuuriga koroonaviiruse testimises. [6]

Küll aga on ka Mediposti serveri kasutamise juures raskendav asjaolu see, et tegu pole terviklahendusega, vaid iga uus süsteemi kasutaja peab ise looma oma andmebaasiga ühilduva rakenduse, mis suudaks ka Medipost serveri poolt esitatud nõuetele vastata. Seetõttu on uute kasutajate lisamine tarkvara haldaja jaoks aeganõudev protsess.

2 SUHTLUS MEDIPOSTIGA

Selleks, et Mediposti kasutamise jaoks vajalikku klientprogrammi luua, on esmalt vaja täpselt teada, kuidas Medipost ise töötab – millistest päringutest see aru saab ja kuidas neile vastab. Selleks on Medipostis välja töötatud vastav dokumentatsioon. [7]

Medipost eristab nelja peamist sõnumimalli, mille kaudu tuleb kogu vajalik infovahetus realiseerida. Tegemist on XML formaadis sõnumitega, millel on täpselt ette määratud, millist infot ja millises järjestuses need sisaldama peavad.

Seega tuleb edasiseks tööks kõigepealt mõista iga sõnumitüübi vajalikkust ja kasutusvõimalusi. Seejärel saab kokku panna skeemi, mis kirjeldaks ainult lubatud sõnumitüüpe kasutades kogu andmevahetust kahe ettevõtte vahel.

2.1 Teenused

Selle sõnumi idee on võimaldada asutusel tutvustada enda tegevusi. Selleks koostatakse nimekiri kõikidest enda poolt hetkel pakutavatest teenustest koos infoga, mida on nende sooritamiseks vaja.

Parema ülevaate saamiseks on võimalik koondada sisult sarnaseid teenuseid uuringugruppideks. Iga grupp sisaldab hulka konkreetseid uuringuid. Ka iga uuring ise võib endas sisaldada ühte või mitut alamuuringut ehk testi.

Teenuste kaudu saavad teised asutused teada, milliseid teenuseid ja kellelt tellida saab, ja mida on selleks veel teha tarvis.

2.2 Tellimus

Tellimus on asutuse võimalus valida enda jaoks välja vajalikud uuringud. Erinevalt Teenustest on siin oluline ära märkida nii tellija kui ka pakkuja asutused. Samuti on oluline märkida konkreetselt tellimuse esitanud isik ja selle eest vastutaja(nt. pereõde ja perearst).

Vaja on ka ära märkida tellimuse olulisus (cito!) ja konfidentsiaalsus – kellel ja millal on lubatud hiljem tulemustega tutvuda. Lisaks on oluline ka kõikvõimalik täiendav informatsioon: esialgne diagnoos, patsiendi info ja saatekiri.

2.3 Vastus

Tegemist on labori vastusega saadetud uuringutellimusele. Selguse huvides peavad vastuses olevate uuringute ID- ja järjekorranumbrid kattuma tellimuse omadega. Küll aga võib vastus saabuda väiksemate osadena, mistõttu ei saa orienteeruda uuringugruppide kaudu. Lisaks on oluline lisada vastamisega seotud isikud, tellimuse olek (ootel/tehtud/tühistatud) ja uuringu läbiviimise aeg. Lisaks on võimalik lisada ka vastust täiendavaid dokumente PDF formaadis ning kommentaare.

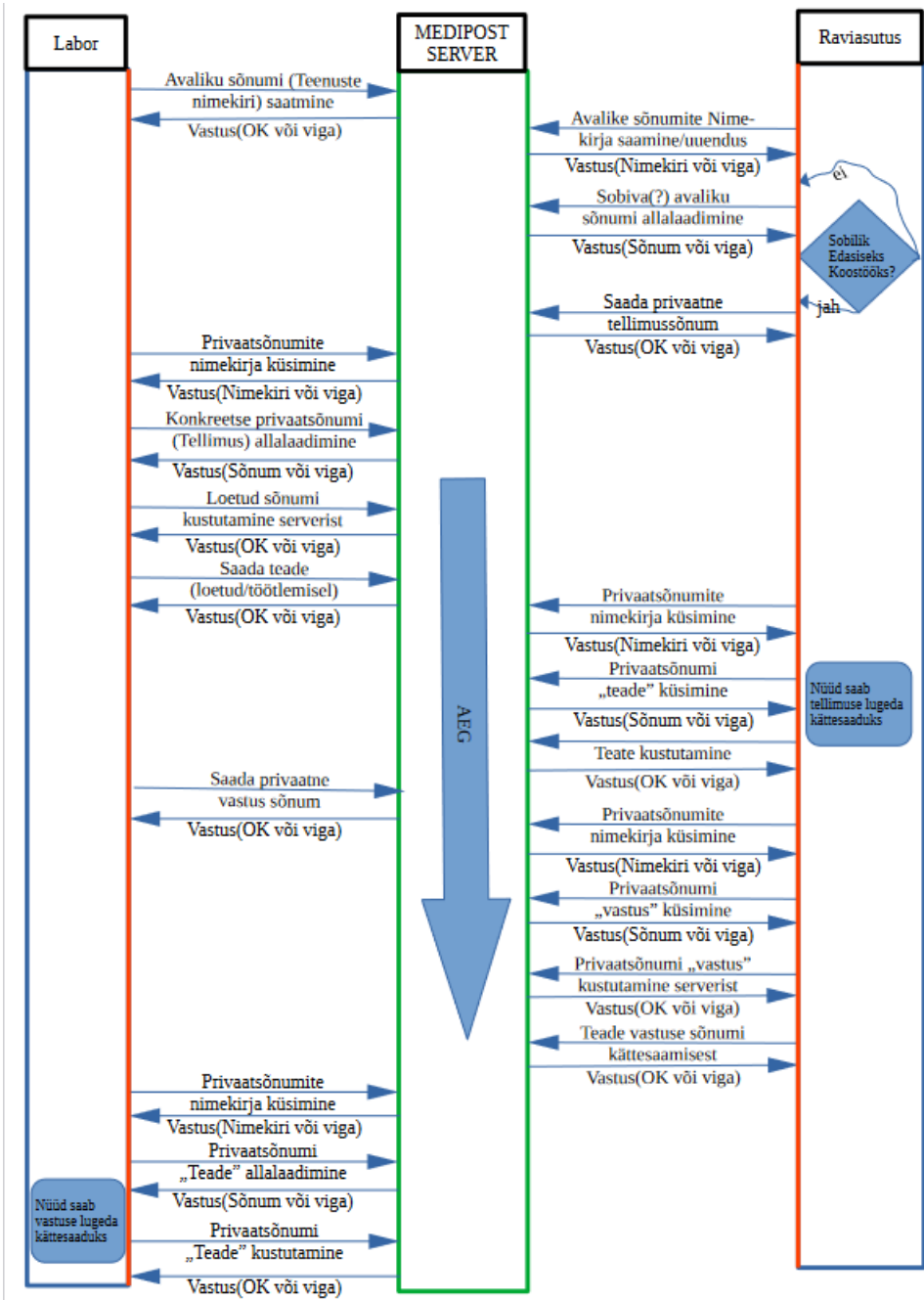
Konkreetse uuringu vastus sisaldab ka hulka tehnilisi parameetreid. Näiteks tulemuseks saadud vastuse suurusjärg (tavaliselt 1/100) ja normaalse vastusevahemiku kohta.

2.4 Teade

Täiendav sõnumitüüp, mida kasutatakse infovahetuse töötamise kontrolliks. See on seotud konkreetse tellimuse või vastusega. See saadetakse kinnituseks varem saabunud sõnumi kättesaamise ja arusaadavuse kohta. Vajadusel võib sisaldada ka saabunud sõnumi vigast osa kodeeritud kujul koos kommentaariga, mis seletab vea tüüpi.

2.5 Üldine andmevahetus

Lisaks sõnumitüüpidele on Medipostis olemas ka etteantud nimekiri võimalikest päringutest ja nende argumentidest, millega saab nimetatud sõnumitüüpe saata, või sobivate õiguste olemasolul ka lugeda või kustutada. Nende teadmiste kaudu saab lihtsustada kogu andmevahetuse kahe ettevõtte vahel üheks skeemiks. (Vt. Joonis 1)



Joonis 2.1 üldine andmevahetus

Joonisel postitab üks asutus oma teenuse või grupeeritud teenuste nimekirja Mediposti serverisse. Seejärel võib keegi teine selle endale alla laadida. Kui allalaaditud uuringute nimekirjas on mõni selline, mida asutus soovib kasutada, on neil läbi Mediposti võimalik esitada tellimus. Tellimused saadetakse tagasi esimesele asutusele, kes saab neid oma laboris sooritada ning hiljem tulemused kas ükshaaval või grupina tagasi saata.

Kui esimene asutus ei tegele mingil põhjusel enam nimekirjas loetletud uuringutega, saab ta minna protsessi algusesse tagasi ning saata välja uuendatud teenuste loetelu. Kui ühel asutusel on serveris salvestatud mitu nimekirja, siis kokkuleppeliselt arvestatakse vaid uusimaga.

Praktikas ei pruugi Medipost anda alati päringule soovitud vastust, vaid võib tagastada ka veateate. Sellisel juhul jääb andmevahetusprotsess ajutiselt pooleli ning vaja on inimkasutaja, kes suudaks vea põhjused tuvastada ja vajadusel ka parandada, sekkumist.

Samuti ei ole võimalik anda kogu ettevõttele ühte rolli, vaid on võimalik olla erinevate Medipost kasutajatega suheldes olla vastavalt vajadusele nii tellija kui ka teenuse pakkuja.

Seega tuleb eelneva joonise põhjal luua ülevaate tegevustest ning funktsionaalsustest, mida on klassil vaja realiseerida Medipost serveriga suhtlemiseks.

3 EESMÄRGID

Töö peamine eesmärk on luua programm, mille abil saaksid PHP keeles loodud tarkvaraprojektid vajadusel kasutada kõiki Medipost dokumentatsioonis kirjeldatud funktsionaalsusi. Selle jaoks oli testimiseks võimalus kasutada Asper Biotech geeniuuringute labori kliendihaldustarkvara arendusserveri koopiat.

3.1 Funktsionaalsed Nõuded

Nimekiri funktsionaalsusest (vt. tabel 3.1), mille puudumisel oleks oluline osa kasutamisevõimalustest puudu, mistõttu saaks töö lugeda poolikuks.

Tabel 3.1 funktsionaalsed nõuded

| Vajadus | Tegevused |
|-------------------------|--|
| Sõnumite lugemine | Ühildumine Medipostiga etteantud kasutajatunnuse kaudu, sõnumite nimekirja päring, sõnumi allalaadimine |
| Veatöötlus | Tagasiside, kas päring õnnestus või on vigane |
| Kasutajavormi puudumine | Sõnumite genereerimine otse andmebaasist, ei ole vahepealset tekstifaili tegemist ega faili üleslaadimise hüplikakent. |

3.2 Mittefunktsionaalsed nõuded

Ülesanded, mida on vaja kindlasti Mediposti korrektseks toimimiseks vaja, kuid mis ei anna otseselt kasutajale mingit lisaväärtust (vt tabel 3.2)

Tabel 3.2 mittefunktsionaalsed nõuded

| NÕUE | VAJALIKKUS |
|-------------------------------|---|
| HTTP protokoll kasutamine | Nii Mediposti server, kui ka enamik andmebaase omavad internetiühendust, mistõttu on HTTP üks kõige tõenäolisemalt töötavatest vahenditest(tulemüür laseb läbi). |
| XML genereerimine | Mediposti sõnumid eeldavad XML formaati, mille struktuur ja järjestus on määratud vastavate xsd failidega . Teoorias võib mõnes andmebaasis see juba olemas olla, kuid üldiselt on seal ainult lihtsalt SQL tabel. Seega tuleks XML andmetest jooksvalt genereerida |
| kodeeringud | UTF8 kõikidele parameetritele, eraldi koodistik määrata OID elementidele |
| Automatiseeritud funktsioonid | Oma vananenud sõnumite kustutamine, teated privaatse sõnumi kättesaamisest ning nende kustutamune oma baasi sakvestamisel |

4. MEDIPOST KASUTAMINE ANDMEBAASIDES

5DVisionis arendatavad andmebaasid on üldiselt loodud PHP programmeerimiskeeles, ning kasutavad firmas loodud 5D Basement platvormi, mistõttu võiksid need olla piisavalt sarnased, et sama moodulit peaks olema võimalik kasutada erinevates projektides. Selleks peaks moodul olema realiseeritud PHP klassina.

Samuti oleks kasulik, kui moodulist saadavad lisafunktsioonid järgiksid PHP-le omast objekt-orienteeritud stiili ning nende tähendus võiks olla juba nimetuse põhjal arusaadav. Seetõttu on otse medipostiga suhtlemiseks mõeldud meetodite nimetused samad, mis vastavad Medipost päringute nimetused.

Kuna mediposti eesmärk on olla maksimaalselt kasutatav, siis antakse kogu ettevõtte kasutajaskonna peale välja vaid üks kasutajatunnus. Seetõttu ei ole võimalik Medipostis kasutajate tegevusi täpselt logida ning seetõttu võib tegemist olla turvariskiga.

Seetõttu peaksid turaküsimused olema lahendatud seda kasutatava andmebaasi siseselt. Kuna minu töö tegeleb juba olemasolevate andmebaasidega, siis seda väga muuta võimalik ei ole ning ainus küsimus on selles, et kuidas salvestada kasutatav parool nii, et seda hiljem live versioonis igaühele näha ei oleks.

Mediposti päringu tegemiseks peab asutusele antud parooli iga päringuga kaasa andma, mistõttu tuleks see automaatselt iga päringu genereerimisel lisada. Seetõttu sai moodul realiseeritud PHP klassina, mille konstrueerimise funktsioon võimaldab autentimise parameetrid klassi loomisel salvestada. Samuti saab sama meetodiga salvestada URL aadressi, kuhu päringud saadetakse.

```
<?php
    function __construct($name=NULL, $pass=NULL){
        $this->User = $name;
        $this->Password = $pass;
        $this->URL = 'https://meditest.medisoft.ee:7443/Medipost/MedipostServlet' . '?user=' .
$this->User . '&password=' . $this->Password;
    }

function setUser($name){
    $this->User = $name;
    $this->URL = $this->URL =
'https://meditest.medisoft.ee:7443/Medipost/MedipostServlet' . '?user=' . $this->User . '&password=' .
$this->Password;
}

function setPassword($pass){
    $this->Password = $pass;
    $this->URL = $this->URL =
'https://meditest.medisoft.ee:7443/Medipost/MedipostServlet' . '?user=' . $this->User . '&password=' .
$this->Password;
}
?> Klassi loomine
```

Seega on mooduli lisamiseks olemasoleva projekti lehele vaja vaid järgmist koodi:

```
include_once('path_to_medipost.class.php'); - koodi füüsiliseks kopeerimiseks.
```

```
$mp = new medipost($kasutaja, $parool) – vajalike andmete kinnitamiseks.
```

4.1 Sõnumite lugemine

Mediposti andmebaas on põhimõtteliselt üks suur tabel, kuhu on salvestatud kõik sinna saadetud XML sõnumid ning nende metaandmed: saatja, saatmise aeg ja eralduseks ID-number. Samuti on kõik sõnumid jagatud vastavalt nende lugemisõigustele kas avalikeks (teenused) või privaatseteks (tellimus, vastus, teade).

Sõnumite lugemiseks tuleb teada selle ID-koodi. Seetõttu tuleb konkreetse sõnumi lugemiseks kasutada näiteks sellist käsklust:

```
$mp->getPublicMessageList($saatja);
```

Õnnestunud päringu puhul tagastatakse kõik kasutaja \$saatja avalike sõnumite päised. Valida tuleks kõige uuema sõnumi ID ning tellida selle sõnumi allalaadimine.

```
$mp->getPublicMessage(ID);
```

```
<?php
function getPublicMessageList($sender=NULL, $date=NULL, $type=NULL){
    $request = $this->URL . '&action=' . __FUNCTION__;
    if($date) $request.= '&lastchecked=' . $date;
    if($type) $request.= '&messagetype=' . $type;
    if($sender) $request.= '&sender=' . $sender;
    $context = $this->addOptions("GET");
    $xml = file_get_contents($request, false, $context);
    $this->validate($xml);
}
?> Get päringu näide
```

Selliste funktsioonide kutsumisel kogutakse kokku kõik funktsioonile kaasa antud argumendid ning samuti eelnevalt klassi loomisel salvestatud andmed. Nendest tuleb kokku panna URL kodeeringuga [8] päring ning saata see Mediposti serverile. Päringu põhimeetodi tuletatakse PHP maagilise konstandi [9] abil kasutatud funktsiooni enese nime küsides.

4.2 Tagastatav väärtus

Medipost võib päringule vastata kas soovitud sõnumiga või tagastada veateate. Seetõttu ei saa päringu vastust kohe andmebaasi lisada vaid tuleb vigased vastused välja filtreerida. Mõlemad vastusetüübid saadetakse XML struktuuriga tekstina.

Seetõttu sisaldab loodud PHP klass eraldi valideerimisfunktsiooni, mis salvestab Medipostist tulnud vastuse XML objektina ning kontrollib selle esimesi elemente. Juhul, kui esimene element on 'code' ja teine element ei ole 'ok' on tegemist veaga. Lihtsaim viis selle info edastamiseks kasutajale koos sõnumi sisulise poolega on vastuse salvestamine ühte kahest muutujast. Seega saab funktsiooni kasutaja seda ise kontrollida:

```
if($mp->errors){} - vigase sisendi korral  
else {} - võib vastused salvestada.
```

Selline lahendus tagab ka selle, et siis on tagastatav väärtus juba organiseeritud XML objekt, mistõttu saab sellest oluliselt lihtsamalt vajalikku informatsiooni välja filtreerida.

```
<?php  
private function validate($request){  
    $xml = simplexml_load_string($request);  
    if($xml->children()[0]->getName() == "CODE" && $xml->children()[1]!="OK"){ //  
veateate struktuuriga vastus  
        $this->errors = $xml->children()[1]; //veateade  
        $this->answer = ""; // vana päringu andmed kinni  
    }  
    else{  
        $this->answer = $xml; // salvestatav xml objekt  
        $this->errors = ""; //vana vea kustustus  
    }  
}  
?> Kontroll, kas õige vastus või veateade
```

5. SÕNUMI KOOSTAMINE

Päringute tegemiseks Medipost serverile läbi mooduli, on kasutusel PHP funktsioon `file_get_contents`. See funktsioon vajab oma argumentideks infot lugemise asukoha kohta (antud juhul Medipost URL aadress) ja lugemisviisi. Erinevalt sõnumite lugemisest, tuleb uute sõnumite üles laadimiseks kasutada multipart/form-data^[5] päringut. Seetõttu on enne päringu tegemist vaja tuvastada päringutüüp. See on eriti oluline just sõnumite saatmisel, kuna kogu päring tuleb genereerida manuaalselt.

```
case "POST": // multipart/form-data request
    define('MULTIPART_BOUNDARY', '-----'.microtime(true));

    $content = "--".MULTIPART_BOUNDARY."\r\n".
        "Content-Disposition: form-data; name=\"Action\"\r\n".
        "Content-Type: text/plain; charset=UTF-8\r\n".
        "Content-Transfer-Encoding: 7bit\r\n\r\n".
        $action. "\r\n";

    $content.= "--".MULTIPART_BOUNDARY."\r\n".
        "Content-Disposition: form-data; name=\"User\"\r\n".
        "Content-Type: text/plain; charset=UTF-8\r\n".
        "Content-Transfer-Encoding: 7bit\r\n\r\n".
        $this->User. "\r\n";
```

Fragment saatmispäringu koostamisest. Tuleb teha käsitsi, mitte faili laadimisega

Kõik Mediposti toetatud sõnumitüübid on XML failid. Teoreetiliselt – näiteks kui laetakse alla olemasolev sõnum uuendamiseks – võib ka andmebaasis olla saatmiseks sobilik sõnum, kuid üldjuhul on andmed salvestatud SQL tabeli(te)na.

Seetõttu on vaja eraldi funktsionaalsust, mis suudaks kokkulepitud formaadis andmetest – mida oleks lihtne otse andmebaasist saada – tekitada Medipost reeglitele vastava XML faili. XML failid koosnevad vaid etteantud järjekorras omavahel seotud elementidest ja nende elementide omadustest.

Seetõttu sobib XML loomiseks hästi PHP SimpleXML laiendus. See võimaldab lugeda algandmeid tsükliks ning liita neid vajalikus kohas XML-i sisse funktsiooniga `addAttribute()` või `addChild()`.

5.1 Struktuur

Ideaalis võiks andmebaas anda Mediposti moodulile ette kahemõõtmelise massiivi uuringuteks ja nendega seotud väärtustest. Medipostil on nõue, et iga saadetud uuring sisaldaks infot uuringule vastava LOINC [10] koodistiku kohta. Tegemist on üle-Eestilise

laboriuuringutes kasutatava standardiseerimisprotsessiga, mis aitab vältida eri asutustes sama uuringu valesti tõlgendamist sisestusvigade või erinevate käibelolevate asustusesiseste nimetuste tõttu. Seega võiks andmeid leida umbes sellise SQL päringuga: 'select * from uuringud where SendtoMedipost=1;'

Praktikas ei pruugi sellisest andmestruktuurist alati piisata, kuna suurema hulga uuringute puhul on soovituslik jagada sarnast tüüpi uuringud uuringugruppideks. Samuti võib iga individuaalne uuring sisaldada ise alamuuringuid.

Seega annab kasutaja ette ainult sõnumitüübi (teenused, tellimus või vastus) ja massiivi Mediposti postitavate andmetega. Moodul peab leidma andmed – eeldusega, et need on sisestatud õiges järjekorras ja lisama need XML'i.

```
<?php
function generate($data){ // array to XML
    $type=array_keys($data)[1];
$xml = new SimpleXMLElement('<?xml version="1.0" encoding="utf-8"?><' . $type . '></' . $type . '>');
    $teostaja = $xml->addChild("Teostaja", "");
    $asutus = $teostaja->addChild('Asutus','');
    foreach($data['Asutus'] as $key=>$value){ // asutuse elemendid peavad olema õiges
järjekorras
        $asutus->addChild($key, utf8_encode($value));
    }
    $nr=1; //counter

    foreach($data[$type] as $grupp => $testid){
        $uuringud = $teostaja->addChild('UuringuGrupp');
        $uuringud->addChild('UuringuGruppId', utf8_encode($grupp)); // Grupi id
andmebaasis
        $uuringud->addChild('UuringuGruppNimi',utf8_encode($grupp));
        $uuringud->addChild('UuringuGruppJarjekord', $nr); //grupi järjekorranr.
        $nr++;
        foreach($testid as $name => $elements){
            $uuring = $uuringud->addChild('Uuring');
            $uuring->addAttribute('tellitav', 'JAH');
            $element = $uuring->addChild('UuringuElement');
            $element->addChild('UuringIdOID', '2.16.840.1.113883.6.1'); // konstant,
LOINC kodeerimise versioon
            foreach($elements[0] as $test => $nimetus){ // uuringu eestistatud LOINC
raamistik
                $element->addChild($test, utf8_encode($nimetus));
                if($test=='KNimetus'){
                    if(next($elements[0])!='Uuringnimi'){
                        $element->addChild('UuringNimi',utf8_encode($nimetus));
//asustusesisese nime puudumisel kasutatakse LOINC oma
                    }
                }
                next($elements[0]);
            }
        }
        if($elements[1]){
            $element = $uuring->addChild('MaterjalideGrupp');
            $element->addAttribute('vaikimisi', 'JAH');
            $materjal = $element->addChild('Materjal');
            $materjal->addChild('MaterjaliTyypOID', '1.3.6.1.4.1.28284.6.2.1.244.3');
            foreach($elements[1] as $test => $nimetus){
                $materjal->addChild($test, utf8_encode($nimetus));
            }
        }
    }
    return $xml;
}
?> XML GENEREERIMINE
```

Ideaalis võiks sõnumite genereerimine toimida lihtsamalt: maksimaalselt kahekordne tsükkel, mis lisab ükshaaval vajalikud elemendid ja nende grupid utf8 kodeeringuga. Paraku on aga nõutud XML failis peaaegu igale elemenditüübile lisada OID koodistik[11]. Sama tüüpi elementidel on kasutusel sama string OID määramiseks. Samuti nõuab Medipost, et uuring kasutaks LOINC kodeerimiseks konstantset versiooninumbrit '2.16.840.1.113883.6.1'. Seetõttu ei ole mõistlik samu andmeid koos iga elemendiga andmebaasi salvestada, vaid tuli jätta esialgu koodi ülesandeks see õigetesse kohtadesse lisada.

5.2 Ühisosa

Iga sõnumitüüp Medipostis algab päisega, mis kirjeldab sõnumi saatja Medipost kasutajanime ning sõnumi saatmise aega. Kuigi see info võib ka andmebaasis olemas olla, ei ole selle saatmine iga kord koos päringu informatiivse osaga mõistlik. Selle asemel saab tekitada päisest ajutiselt uue XML fragmendi ning selle hiljem saatmiseks põhiosaga kokku liita. Kahjuks ei võimalda SimpleXML sellist liitmist, mistõttu tuleb info ajutiselt teisendada keerukamaks, kuid suurema funktsionaalsusega DOM-elementiks.

```
<?php
private function addHeader($type, $receiver=null){ // sõnumi päis
    $lookup = array("Teenused"=>"SL", "Tellimus"=>"OL", "Vastus"=>"AL", "Teade"=>"ME");
    $xml = new SimpleXMLElement('<?xml version="1.0" encoding="utf-
8"?><Saadetus></Saadetus>');
    $header = $xml->addChild('Pais', "");
    $version = $header->addChild('Pakett',$lookup[$type]);
    $version->addAttribute('version','20');
    $header->addChild('Saatja', $this->User);
    if($receiver) $header->addChild('Saaja', $receiver); // privaatsõnumites
    date_default_timezone_set('Europe/Tallinn');
    $header->addChild('Aeg', date("Y-m-d
    return $xml;
?> Päise loomine
```

```
<?php
private function sxml_append(SimpleXMLElement $to, SimpleXMLElement $from) {
    $toDom = dom_import_simplexml($to);
    $fromDom = dom_import_simplexml($from);
    $toDom->appendChild($toDom->ownerDocument->importNode($fromDom, true));
}
?> Päise ja sõnumi põhiosa kokkupanek
```

Samuti on enamiku sõnumitüüpide (v.a. teade) olemas tellija ja/või pakkuja asutus. Kuigi juba klassi loomisel salvestati ka Mediposti kasutaja info, ei ole tegelikult mingit garantiid, et asutuse kasutatav nimetus Medipostis on sama, mis nende päris äriregistri järgne nimetus. Seetõttu tuleb sõnumi loomiseks anda lisaks uuringute jms. infole kaasa ka konkreetse asutuse või allüksuse nimi.

TESTIMINE JA TULEMUSED

Loodud mooduli testimine toimus osaliselt ka erialapraktika koosseisus. Selleks tuli esmalt luua githubi versioonihaldustarkvara abil koopia Asperi tarkvaraprojektist ja see oma serveris tööle saada. Kuna antud projektis ei ole varem Mediposti teenuseid kasutatud, tegin sinna sinna uue andmebaasi Medipostist saabunud vastuse-elementide salvestamiseks.

Seejärel lisasin moodulis loodud tarvara Asperi projekti koosseisu ning sain teha lihtsa kontrolli: kui päringule vastavad andmed uute baasi salvestusid, võis pidada info pärimise osa töötavaks. Loomulikult tuleks mooduli lõplikuks integreerimiseks panna paika täpsem kava: milleks Medipostist saadud andmeid päriselt kasutada saab ning seega millistesse olemasolevatesse tabelitesse neid seega mõistlik salvestada.

XML

Keerulisem oli sõnumite saatmist valideerida. Vigase sisendi korral suudab Medipost öelda, kas viga on päringus endas või konkreetses sõnumis. Vigase päringu puhul tuli hoolega võrrelda etteantud näidispäringuid koodi poolt genereeritud päringuga ja koodi vajadusel täiendada.

Paraku oli korrektse sõnumi koostamine mõnevõrra keerulisem. Medipost loeb vigaseks nii iseenda dokumentatsioonis toodud näidissõnumid kui ka suure osa sinna teiste poolt varem sisestatud sõnumitega identselt koostatud sisuga sõnumitest. Seega on vigade tekkepõhjustest keeruline aru saada.

Seetõttu kasutasin XML genereerimise funktsiooni testimiseks avalikku Asperi teenuste nimekirja. [12] See nimekiri ei ole päriselt ajakohane – osa teenuseid võib olla aegunud või ajutiselt mitte kasutuses. Kuna selline sõnum osutus väga pikaks, siis tegin võimalike vigade tuvastamiseks ajutise reegli, et ühtegi uuringugruppi ei lisata üle kolme elemendi(Vt. Lisa1).

Seejärel katsetasin sama funktsiooni juba Asperi põhibaasis oleva infoga. Siiski on tegemist üsna keerulise tarkvaraga, mistõttu on raske oletada, millised uuringud päriselt

olulised on. Seega oleks praktikas vaja eelnevalt andmebaasi siseselt tuvastada, milline tähtsus konkreetsel uuringul hetkel on.

Lisaks saab sellisel kujul nimekiri olla vaid illustratiivne, kuna seda ei ole võimalik tellida. Tellimiseks tuleks iga uuringuga siduda korrektne info selle läbiviimiseks vajalike materjalide[13] ja nende hoiustamiseks mõeldud mahutite kohta.

Testimisel sain küll juhuslike materjaliandmetega kontrollida, et vajalikud elemendid lisatakse õigesse kohta ja et sõnum vastab Mediposti valideerimisreeglitele, kuid sellisena on info tegelikult kasutu.

Kuna Medipost lubab tellimussõnumeid saata ka iseenda kasutajale, siis oli võimalik ka selle struktuurilist õigsust kontrollida. Samuti sain tehtud näidistellimuste põhjal proovida ka vastuse sõnumi moodustamist.

Praktikas oleks aga vaja rohkem teadmisi sellest, kuidas konkreetsed uuringuandmed, materjaligrupid ning hinnakirjad omavahel seotud on. Samuti on ilma piisavate taustteadmisteta keeruline isegi reaalse tellimuse peale leida sobivat tüüpi andmeid, mida Asperis peaks pärast analüüside tegemist vastuse koostamiseks valida.

KOKKUVÕTE

Lõputöö tulemusena valmis PHP põhine moodul, mis on ühendatud olemasoleva andmebaasi arendusversiooni koopiaga. Seda klassi on võimalik kasutada Medipostis olemasolevate sõnumite lugemiseks ja vajadusel ka kustutamiseks. Samuti suudab klass eristada Mediposti päringute korrektseid vastuseid veateadetest ning võimaldab tagastatud väärtusi kuvada lihtsalt salvestatavas formaadis.

Samuti on võimalus ise genereerida kõiki Medipostis kasutatavaid sõnumitüüpe ja saata neid serverisse koos sellega kaasnevate tegevustega(aegunud sõnumite kustutamine, teavitus sõnumi kättesaamise kohta jms.). Samuti on tehtud esmased näidisfailid XML genereerimiseks vajaliku info leidmisest andmebaasist.

Paraku on projekt, mille peal Mediposti funktsionaalsust vahendavat klassi testiti, üsna keerulise ülesehitusega. Seetõttu pole mul täielikku ülevaadet sellest, kellele ja millal soovitakse Medipostis sõnumeid saata. Samuti pole piisavalt valdkonnaspetsiifilisi teadmisi, et siduda omavahel konkreetseid asutuselt tellitud uuringuid nende vastustega.

Seetõttu ei ole seda moodulit veel kindlasti live serveris kasutamisele võtmiseks piisavalt testitud ning suure tõenäosusega on selles veel ka edaspidi muudatusi vaja teha.

SUMMARY

The result of this thesis is an PHP module that is connected to a copy of an medical laboratory database. It can be used to exchange data with other medical databases via Medipost messaging service. The module is able to download and delete existing messages from the server. It is also able to differentiate faulty responses from correct.

The module is tested to be able to send all accepted types of messages to Medipost. In some cases, it has used laboratory' own database to generate correct XML messages.

However, as the underlying Medical database is pretty complex, it is not yet clear, which messages and to whom, is actually needed to send. Also, the module has not been tested on connecting assays and their results as connected messages.

Therefore, the module is not yet fully integrated to the database and might need additional testing/programming, before it can be used on the live version.

KASUTATUD KIRJANDUSE LOETELU

- [1] "Data Resources in the Health Sciences", Health science library University of Washington, [Võrgumaterjal]. Available: <https://guides.lib.uw.edu/hsl/data/findclin> [Kasutatud 20 Mai 2022].
- [2] „Index of /Failid/Klient/Medipost“, Medipost, [Võrgumaterjal]. Available: https://medisoft.ee/Failid/Klient/Medipost/Medipost_V2_9.zip [Kasutatud 20 Mai 2021].
- [3] E-arvete saatmine Mediposti Liidestusjuhend, Medipost, [Võrgumaterjal]. Available: https://medisoft.ee/Failid/Klient/Apteegi_infomaterjalid/E-arvete_saatmine_Mediposti.pdf [Kasutatud 20 Mai 2022].
- [4] „X-tee statistika“, x-tee, [Võrgumaterjal]. Available: <https://www.x-tee.ee/factsheets/EE/>. [Kasutatud 20 Mai 2022].
- [5] „Infosüsteemide andmevahetuskiht“, riigiteataja, [Võrgumaterjal]. Available: <https://www.riigiteataja.ee/akt/127092016004>. [Kasutatud 20 Mai 2022].
- [6] „MISP2“, Riigi infosüsteemi amet, [Võrgumaterjal]. Available: <https://www.ria.ee/et/riigi-infosusteem/x-tee/rakendused/misp.html> [Kasutatud 20 Mai 2022].
- [7] „Meist“, Medisoft, [Võrgumaterjal]. Available: <https://medisoft.ee/meist/> [Kasutatud 20 Mai 2022].
- [8] „Form-based File Upload in HTML“, ieft, [Võrgumaterjal]. Available: <https://www.ietf.org/rfc/rfc1867.txt> [Kasutatud 20 Mai 2022].
- [9] „Magic constants“, PHP, [Võrgumaterjal]. Available: <https://www.php.net/manual/en/language.constants.magic.php> [Kasutatud 20 Mai 2022].
- [10] „LOINC tööühm“ Eesti Laborimediitsiini Ühing, [Võrgumaterjal]. Available: <https://www.elmy.ee/tooruhmad/loinc/> [Kasutatud 20 Mai 2022]
- [11] „Standardite ja klassifikaatorite publitseerimiskeskuse OID-sektsioon“, infosüsteemide keskus, [Võrgumaterjal]. Available: <http://pub.e-tervis.ee/oids.py/> [Kasutatud 20 Mai 2022].
- [12] labori analüüsid, Tervise ja heaolu infosüsteemide keskus, [Võrgumaterjal]. Available: https://elhr.digilugu.ee/data/labori_uuringudList.html?laboriId=30 Kasutatud 20 Mai 2022]
- [13] „Proovimaterjali tüüp meditsiinilaborites“ Tervise ja heaolu infosüsteemide keskus, [Võrgumaterjal]. Available: <http://pub.e-tervis.ee/classifications/Proovimaterjali%20t%C3%BC%C3%BCp%20meditsiinilaborites/11> [Kasutatud 20 Mai 2022].

Lisa 1: genereeritud teenuste nimekirja xml

```
<Teenused>
<Teostaja>
<Asutus>
<Asutuseld>10504931</Asutuseld>
<AsutuseNimi>Asper Biotech Ltd. püüdlõä</AsutuseNimi>
<AsutuseKood>??</AsutuseKood>
</Asutus>
<UuringuGrupp>
<UuringuGruppId>TU</UuringuGruppId>
<UuringuGruppNimi>TU</UuringuGruppNimi>
<UuringuGruppJarjekord>1</UuringuGruppJarjekord>
<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>20990-8</UuringId>
<TLyhend>B-APC seq</TLyhend>
<KNimetus>Perekondlik adenomatoosne polüpoos (FAP) - APC geeni mutatsioonid (sekvenerimine)</KNimetus>
<UuringNimi>Perekondlik adenomatoosne polüpoos (FAP) - APC geeni mutatsioonid (sekvenerimine)</UuringNimi>
</UuringuElement>
</Uuring>
<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>21639-0</UuringId>
<TLyhend>B-BRCA1 seq</TLyhend>
<KNimetus>Pärilik rinna- ja munasarjavähk - BRCA1 geeni mutatsioonid (sekvenerimine)</KNimetus>
<UuringNimi>Pärilik rinna- ja munasarjavähk - BRCA1 geeni mutatsioonid (sekvenerimine)</UuringNimi>
</UuringuElement>
</Uuring>
<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>21731-5</UuringId>
<TLyhend>B-RB1 NGS</TLyhend>
<KNimetus>Retinoblastoom - RB1 geeni mutatsioonid (NGS)</KNimetus>
<UuringNimi>Retinoblastoom - RB1 geeni mutatsioonid (NGS)</UuringNimi>
</UuringuElement>
</Uuring>
<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>38531-0</UuringId>
<TLyhend>B-BRCA2 seq</TLyhend>
<KNimetus>Pärilik rinna- ja munasarjavähk - BRCA2 geeni mutatsioonid (sekvenerimine)</KNimetus>
<UuringNimi>Pärilik rinna- ja munasarjavähk - BRCA2 geeni mutatsioonid (sekvenerimine)</UuringNimi>
</UuringuElement>
</Uuring>
</UuringuGrupp>
<UuringuGrupp>
<UuringuGruppId>PH</UuringuGruppId>
<UuringuGruppNimi>PH</UuringuGruppNimi>
<UuringuGruppJarjekord>2</UuringuGruppJarjekord>
<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>21656-4</UuringId>
<TLyhend>B-CFTR seq</TLyhend>
<KNimetus>Tsüstiline fibroos (CF) - CFTR geeni mutatsioonid (NGS)</KNimetus>
<UuringNimi>Tsüstiline fibroos (CF) - CFTR geeni mutatsioonid (NGS)</UuringNimi>
</UuringuElement>
</Uuring>
<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>21704-2</UuringId>
<TLyhend>B-MT-ATP6 m.9176T&gt;C seq</TLyhend>
<KNimetus>Pärilik spastiline parapleegia - MT-ATP6 geeni m.9176T&gt;C mutatsioon (sekvenerimine)</KNimetus>
<UuringNimi>Pärilik spastiline parapleegia - MT-ATP6 geeni m.9176T&gt;C mutatsioon (sekvenerimine)</UuringNimi>
</UuringuElement>
</Uuring>
<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>21714-1</UuringId>
<TLyhend>B-MT-TL1 m.3243A&gt;G seq</TLyhend>
<KNimetus>MELAS sündroom - MT-TL1 geeni m.3243A&gt;G mutatsioon (sekvenerimine)</KNimetus>
<UuringNimi>MELAS sündroom - MT-TL1 geeni m.3243A&gt;G mutatsioon (sekvenerimine)</UuringNimi>
</UuringuElement>
</Uuring>
<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>21759-6</UuringId>
<TLyhend>B-FMR1 repeats</TLyhend>
<KNimetus>Fragiilse X-i sündroom (FXS) - FMR1 geeni trinukleotiidsed kordused</KNimetus>
<UuringNimi>Fragiilse X-i sündroom (FXS) - FMR1 geeni trinukleotiidsed kordused</UuringNimi>
</UuringuElement>
</Uuring>
</UuringuGrupp>
<UuringuGrupp>
<UuringuGruppId>FG</UuringuGruppId>
<UuringuGruppNimi>FG</UuringuGruppNimi>
<UuringuGruppJarjekord>3</UuringuGruppJarjekord>
<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>40425-1</UuringId>
<TLyhend>B-CYP2D6 genotype</TLyhend>
<KNimetus>CYP2D6 genotüüp</KNimetus>
<UuringNimi>CYP2D6 genotüüp</UuringNimi>
</UuringuElement>
</Uuring>
```

```

<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>57132-3</UuringId>
<TLyhend>B-CYP2C19 genotype</TLyhend>
<KNimetus>CYP2C19 genotüüp</KNimetus>
<UuringNimi>CYP2C19 genotüüp</UuringNimi>
</UuringuElement>
</Uuring>
<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>79714-2</UuringId>
<TLyhend>B-CYP2C19 met act</TLyhend>
<KNimetus>CYP2C19 metaboolne aktiivsus</KNimetus>
<UuringNimi>CYP2C19 metaboolne aktiivsus</UuringNimi>
</UuringuElement>
</Uuring>
<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>79715-9</UuringId>
<TLyhend>B-CYP2D6 met act</TLyhend>
<KNimetus>CYP2D6 metaboolne aktiivsus</KNimetus>
<UuringNimi>CYP2D6 metaboolne aktiivsus</UuringNimi>
</UuringuElement>
</Uuring>
</UuringuGrupp>
<UuringuGrupp>
<UuringuGruppId>KR</UuringuGruppId>
<UuringuGruppNimi>KR</UuringuGruppNimi>
<UuringuGruppJarjekord>4</UuringuGruppJarjekord>
<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>62343-9</UuringId>
<TLyhend>B-Submicroscopic chromosome aberrations</TLyhend>
<KNimetus>Submikroskoopilised kromosoomi aberratsioonid (ülegenoomne geenikiip)</KNimetus>
<UuringNimi>Submikroskoopilised kromosoomi aberratsioonid (ülegenoomne geenikiip)</UuringNimi>
</UuringuElement>
</Uuring>
</UuringuGrupp>
<UuringuGrupp>
<UuringuGruppId>HLA</UuringuGruppId>
<UuringuGruppNimi>HLA</UuringuGruppNimi>
<UuringuGruppJarjekord>5</UuringuGruppJarjekord>
<Uuring tellitav="JAH">
<UuringuElement>
<UuringIdOID>2.16.840.1.113883.6.1</UuringIdOID>
<UuringId>L-2972</UuringId>
<TLyhend>B-Coeliac panel</TLyhend>
<KNimetus>Tsöliaakia - HLA antigeenide paneel</KNimetus>
<UuringNimi>Tsöliaakia - HLA antigeenide paneel</UuringNimi>
</UuringuElement>
</Uuring>
</UuringuGrupp>
</Teostaja>
</Teenus>

```