

TALLINNA TEHNIKAÜLIKOOL  
Infotehnoloogia teaduskond  
Informaatikainstituut

Miina Strom 120838IABB

**RAHVASTIKUREGISTRI X-TEE TEENUSTE  
TESTIMINE ARENDUSKESKKONNAS  
SOAPUI JA JMETER NÄITEL**

Bakalaureusetöö

Juhendaja: Inna Švartsman  
MSc  
Lektor

Tallinn 2017

## **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.

Autor: Miina Strom

03.01.2017

## Annotatsioon

Käesoleva töö „Rahvastikuregistri X-tee teenuste funktsionaalsuse testimine arenduskeskkonnas soapUI ja JMeter näitel“ põhiliseks eesmärgiks on testida rahvastikuregistri X-tee teenuste funktsionaalsust vastavalt spetsifikatsioonile ning välja selgitada, millise tarkvaraga on efektiivsem teste läbi viia.

Töö käigus kirjeldatakse X-tee teenuste olemust, tuuakse välja veebiteenuste testimise põhimõtted ning selgitatakse, mis on funktsionaalne testimine. Samuti näidatakse, kuidas teste soapUI ja JMeter'iga läbi viia.

Antud töö tulemustest selgus, et soapUI-ga on testimine ajaliselt kiirem. Üheks põhjuseks võib tuua, et päringu järelpärimise ning päringu vastuse tekst on soapUI-s silmale paremini loetavam. Teiseks põhjuseks on *XPath Assertion* aknas olev nupp *Expected Results*, mis annab oodatud tulemuse automaatselt. Kolmandaks põhjuseks tooks *Assertion* i lisamise, mis on soapUI-s palju mugavam.

Arvutamise tulemus näitas, et kogu rahvastikuregistri X-tee teenuste testimiseks kuluks soapUI-ga ligikaudu 4 tundi vähem aega.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 37 leheküljel: 6 peatükki, 13 joonist, 4 tabelit.

## **Abstract**

### **Testing of population register X-Road services in development environment based on an example of soapUI and JMeter**

The main goal of this research paper „Testing of population register X-Road services in development environment based on an example of soapUI and JMeter“ is to test functionality of the Population Register X-Road network services in compliance with the specification and investigate with which software it is more efficient to do the tests.

In the course of the research, the author describes the essence of the ‘X-Road network services, introduces the principles of the web services testing and explains what is the meaning of functional testing. In addition, the examples of soapUI and JMeter tests are given.

The research results showed that testing with soapUI is faster in terms of time. One of the reasons may lie in a fact that a request for information and a text of answer to request in soapUI is better readable for the eyes. Another reason is a button *Expected Results*, which appears in a window *XPath Match Assertion* and which gives the expected result automatically. And the third reason, that the author can mention is adding the *Assertion* which is much more convenient in soapUI.

The results of the calculation demonstrated that the total time spend on testing of the entire network services of the population register in soapUI appeared to be approximately 4 hours shorter.

The thesis is in Estonian and contains 37 pages of text, 6 chapters, 13 figures, 4 tables.

## Lühendite ja mõistete sõnastik

ALAMSÜSTEEM	Tehnoloogiliselt ja organisatoorselt piiritletud X-tee liikme infosüsteemi osa andmeteenuse osutamiseks või kasutamiseks
ANDMETEENUS	X-tee liikme teenus, mille kaudu toimub internetipõhine andmevahetus
ANDMETEENUSE KASUTAJA	X-tee liige, kes kasutab andmeteenust
ANDMETEENUSE OSUTAJA	X-tee liige, kes osutab teistele liikmetele andmeteenus
FTP	<i>File Transfer Protocol</i> , failide edastamine ühest arvutist (serverist) teise
JMS	<i>Java Message Service</i> , rakendusliides, mis võimaldab luua, saata ja lugeda sõnumeid
JSSE	<i>Java Secure Socket Extension</i> , vahend, mille abil saab viia läbi ohutut Interneti kommunikatsiooni
JTIDY	Aitab kontrollida mitesobivat süntaksit HTML'is
JVM	<i>Java virtual machine</i> , tarkvara, mis kompileerib Java lähtekoodi baitkoodi
LDAP	<i>Lightweight Directory Access Protocol</i> , lihtsustatud kataloogisirvimise protokoll, mis on mõeldud kataloogiteenustele ligi pääsemiseks ja nende haldamiseks üle IP-võrgu
PÄÄSUÕIGUS	Andmeteenuse kasutamise võimaldamine X-tee tarkvaras
SOA	<i>Service-oriented architecture</i> , teenuse orienteeritud arhitektuur
TURVASERVER	Tarkvaraline lahendus, mis järgib X-tee baasprotokollistikku
UDDI	<i>Universal Description, Discovery and Integration</i> , veebipõhine jagatud kataloog, mis võimaldab ettevõtetel loetleda end Internetis ja avastada üksteisi
XTEE KESKUS	Riigi Infosüsteemi Amet, kes vastutab X-tee haldamise ja arendamise eest
X-TEE LIIGE	Asutus või isik, kes on liitunud X-teenuga

## Sisukord

1 Sissejuhatus .....	9
2 X-tee tutvustus .....	11
2.1 X-tee turvalisus .....	12
2.1.1 X-tee liikmete kohustused .....	12
2.1.2 X-tee turvaserver .....	13
2.2 Näiteid X-tee V5 keskkonna teenustest .....	14
3 Veebiteenuse .....	15
3.1 Mis on veebiteenus? .....	15
3.2 Veebiteenuste testimine .....	15
4 Tarkvara testimine .....	17
4.1 Funktsionaalne testimine .....	17
4.2 JMeter .....	18
4.3 SoapUI .....	21
4.4 SoapUI ja JMeter'i võrdlus .....	25
4.5 SoapUI ja JMeter'i tehniline võrdlus .....	28
5 Testimine .....	29
5.1 Päringu RR76AndmIK spetsifikatsiooni kirjeldus .....	29
5.2 Päringu RR76AndmIK testjuhtumid .....	29
5.3 Testide läbiviimine .....	30
6 Kokkuvõte .....	34
Kasutatud kirjandus .....	36
Lisa 1 – spetsifikatsioon RR76 .....	38

## Jooniste loetelu

Joonis 1. X-tee on nagu nähtamatu seeneniidistik, mille toel kerkivad üha uued e-teenused. [5].....	12
Joonis 2. Riigi infosüsteemi haldusüsteemi lehel on kirjeldatud kõik 145 rahvastikuregistri X-tee teenust nii neljandas- kui ka viiendas keskkonnas. [8].....	14
Joonis 3. Harude grupp ( <i>Thread Group</i> ) vaade.....	19
Joonis 4. Testitav element ( <i>Sampler</i> ) „SOAP/XML-RPC Request“ vaade. ....	20
Joonis 5. Kuulajad ( <i>Listeners</i> ) „View Results Tree“ vaade. ....	20
Joonis 6. <i>XPath Assertion</i> vaade.....	21
Joonis 7. SoapUI elementide diagramm. [20] .....	22
Joonis 8. Testkogumi ( <i>TestSuite</i> ) lisamine.....	23
Joonis 9. Testjuhtumi ( <i>TestCase</i> ) lisamine.....	23
Joonis 10. Testsammude ( <i>TestSteps</i> ) vaade. ....	24
Joonis 11. <i>XPath Assersation</i> deklareerimine. ....	25
Joonis 12. Testimistulemused kuue teenuse lõikes minutites.....	31
Joonis 13. Kuue teenuse testimistulemus minutites .....	32

## **Tabelite loetelu**

Tabel 1. SoapUI ja JMeter võrdlus .....	27
Tabel 2. SoapUI ja JMeter tehniline võrdlus .....	28
Tabel 3. Testjuhtum 1 .....	30
Tabel 4. Testjuhtum 2 .....	30



# 1 Sissejuhatus

Rahvastikuregister on andmekogu, mis koondab Eesti kodanike, oma elukoha Eestis registreerinud Euroopa Liidu kodanike ja Eestis elamisloa või elamisõiguse saanud välismaalaste peamisi isikuandmeid. Registrit haldab ja arendab vastutava töötlejana Siseministeerium. Rahvastikuregistris olevatele andmetele pääsevad avaliku ülesande täitmiseks ja õigustatud huvi korral ligi nii asutused kui füüsilised ja juriidilised isikud. Üheks võimaluseks saada juurdepääs andmetele on infosüsteemide andmevahetuskiht ehk X-tee. [1]

2. novembril 2015 algas üleminek X-tee versioonile 6. Üleminek on plaanitud lõpetada 2017. aasta keskpaigaks. Teenuse pakkujad peavad 2016. aasta lõpuks avaldama oma andmeteenused X-tee versiooni 6 platvormil toodangukeskkonnas. Andmeteenuste tarbijad peaksid infosüsteemid toodangukeskkonnas üle viima versiooni 6 platvormile 2017. aasta märtsikuu jooksul. [2] Hetkel on Rahvastikuregistril rohkem kui 140 teenust, mis tuleb kõik X-tee versioonis 6 testida.

Töö autor töötab Siseministeeriumi infotehnoloogia- ja arenduskeskuses (edaspidi SMIT) rahvastikuregistri arendusmeeskonnas tarkvara testijana.

Bakalaureusetöö eesmärgiks on selgitada X-tee olemust, funktsionaalse testimise põhitõdesid ning näidata, kuidas soapUI-ga ja JMeter'iga teste läbi viia. Viimsed said valitud, kuna toetavad funktsionaalset testimist, *XPath Assersation* kasutamist ning on avatud lähtekoodiga. Lõputöö põhiliseks eesmärgiks on välja selgitada, kumma tarkvaraga on testimine efektiivsem. Eesmärgi saavutamiseks testitakse 6. teenust, mis valiti juhuslikult. Esmalt tutvub töö autor teenuste spetsifikatsioonidega ning koostab testjuhtumid. Samuti kontrollitakse, kas teenuste andmeväljad tagastavad oodatud tulemuse, mida kontrollitakse *XPath Assertion*'i abiga. Pärast teenuste testimist võrreldakse mõlema tarkvara tugevaid ja nõrku külgi. Testimistulemused esitab töö autor diagrammidena. Lõpuks arvutatakse teoreetiline aeg, mis kulub JMeter'iga ning soapUI-ga 145 teenuse testimiseks.

Töö tulemused on eelkõige vajalikud SMIT'ile kuna antud töö põhjal saab otsustada, kas organisatsioon peaks teenuste testimiseks kasutama tarkvara soapUI või JMeter. Samuti aitab antud töö teisi testijaid, kes soovivad testida päringute funktsionaalsust, aga ei oska valida selleks sobivat tarkvara.

Antud bakalaureuse lõputöö on kirjutatud 2017. aasta talvel Tallinna Tehnikaülikooli lektori Inna Švarstmani juhendamisel. Lõputöö teema ning selle tegemiseks vajalik informatsioon on saadud Siseministeriumi infotehnoloogia- ja arenduskeskuselt.

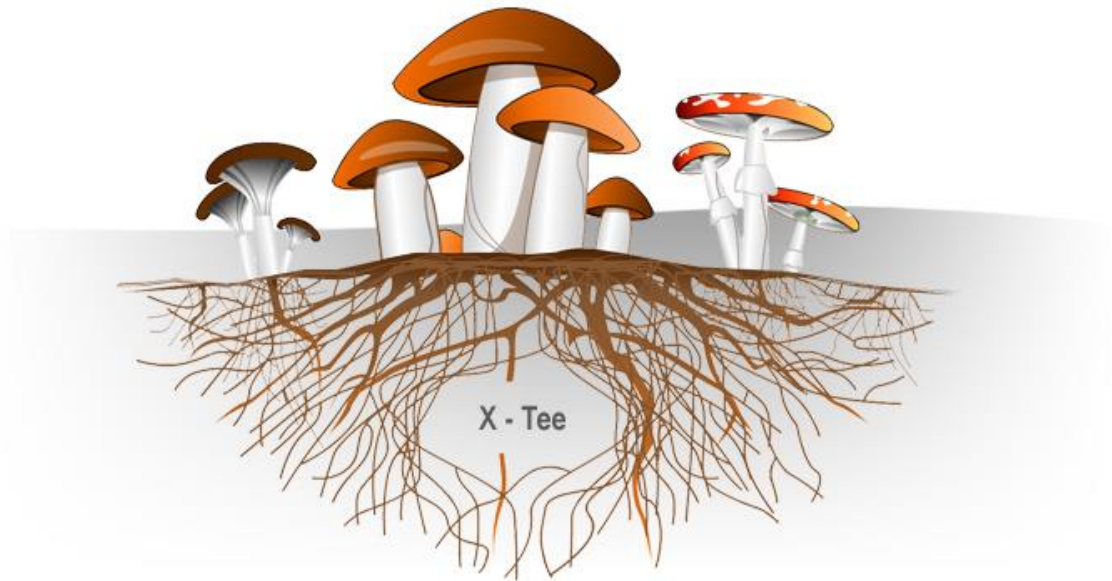
## 2 X-tee tutvustus

Riigi infosüsteemide andmevahetuskiht X-tee on tehniline ja organisatsiooniline keskkond, mis võimaldab korraldada turvalist internetipõhist andmevahetust riigi infosüsteemide vahel. X-tee võimaldab asutustel/inimestel turvaliselt andmeid vahetada, samuti korraldada isikute juurdepääsu riigi andmekogudes säilitatavatele ja töödeldavatele andmetele. Andmekogud pakuvad andmeid X-tee teenuste abil. [3]

Riigi Infosüsteemi Ameti poolt koostatud X-tee kirjeldus toob välja mitmeid põhjuseid, miks X-tee kaudu on liikmetel turvaline andmevahetust pidada. [3]

X-tee tagab liikmele andmevahetuse juures: [4]

- autonoomsuse – X-tee liige defineerib ise, milliseid andmeteenuseid ta soovib pakkuda ning kellele teenuste kasutamise pääsuõigusi anda;
- konfidentsiaalsuse – info jõuab X-tee vahendusel vaid nende osapoolteni, kellele on selleks õigus antud;
- tõestusväärtuse – digitaalallkirja kasutamine võimaldab tõestada, kellelt laekunud andmed pärinevad;
- koosvõime – sõltumata liikme kasutatavast tehnoloogiast või arhitektuurist räägivad kõik X-tee liikmed üksteisega ühises keeles.



Joonis 1. X-tee on nagu nähtamatu seeneniidistik, mille toel kerkivad üha uued e-teenused. [5]

## 2.1 X-tee turvalisus

Vastavalt määrusele „Infosüsteemide andmevahetuskiht“ peavad kõik X-tee liikmed järgima teatud abinõusid, et X-tee andmevahetus oleks turvaline.

Kõik X-tee liikmed peavad rakendama andmekaitse abinõusid ning asjakohaseid füüsilisi, organisatsioonilisi ja infotehnilisi turvameetmeid. Turvalisusega seotud riskide maandamiseks tuleb rakendada andmete käideldavust, terviklust ja konfidentsiaalsust tagavaid meetmeid. Andmeteenust osutatakse ja kasutatakse vastavalt X-tee liikmete vahelisele andmeteenuse kasutamise kokkuleppele. [6]

### 2.1.1 X-tee liikmete kohustused

Andmeteenuse osutaja on kohustatud täitma nõudeid, et tema andmeteenus oleks kasutuskõlblik. Andmeteenus tuleb registreerida koos andmeteenuse tehnilise kirjeldusega turvaserveris ja hoida andmeteenuse kirjeldust nii turvaserveris kui ka RIHA-s asja- ja ajakohasena. [6]

Liige peab rakendama andmekaitse abinõusid ning turvalisusega seotud riskide käsitlemiseks tuleb rakendada andmete terviklikkuse, konfidentsiaalsuse ja käideldavuse meetmeid. Samuti tuleb esitada X-tee keskusele vajalik teave, et hinnata turvaserveri turvalisust. Andmeteenuse kasutamise kokkuleppes määratakse kindlaks andmeteenuse kasutamiseks vajalikud infoturbemeetmed ning teenustaseme tingimused. Liige on

kohustatud määrama kindlaks töö- ja ametikohad, kellel on õigus alamsüsteemi ja seeläbi alamsüsteemile võimaldatud andmeteenuseid kasutada. Kui liikmel peaks esinema X-tee kasutamisega probleem, mis mõjutab X-tee keskust, siis tuleb sellest koheselt teada anda X-tee keskusele. Samuti tuleb X-tee keskust teavitada turvaintsidentidest ja nende vahetust ohust. [6]

Andmeteenuse kasutamine on samuti võimalik X-tee liikmete alamsüsteemides, millele on antud konkreetse andmeteenuse kasutamiseks pääsuõigus. Nii andmeteenuse osutaja kui ka kasutaja peavad järgima andmeteenuse kasutamise kokkulepet ning kohustatud ajatembeldama laekunud sõnumid turvaserveris andmete töötlemiseks vajalikus tempos. [6]

### **2.1.2 X-tee turvaserver**

X-tee liige on kohustatud X-tee turvalise andmevahetuskanali loomise võimaldamiseks paigaldama turvaserveri tarkvara ning registreerima keskuses nõuetekohase turvaserveri autentimissertifikaadi. X-teel on lubatud kasutada vaid sellist turvaserveri tarkvara, mis järgib keskuse kinnitatud X-tee baasprotokollistikku. X-tee liige peab majutama turvaserverit Eesti Vabariigi territooriumil. Keskuse loal võib turvaserveri paigaldada ka väljaspool Eesti Vabariiki. X-tee liige peab oma turvaserveri teistele X-tee liikmetele kasutamise võimaldamisel kasutama krüpteeritud ühendust ning kahepoolset autentimist turvaserveri ja alamsüsteemi ühendamiseks. [7]

Päringulogide osas on X-tee liikmetel järgmised kohustused: [7]

- töötada välja päringulogi arhiveerimise protseduur, mis sisaldab arhiveerimise sagedust ja arhiveeritava informatsiooni loetelu,
- määrata isikud, kes ja millistel tingimustel saavad juurdepääsu turvaserveri arhiveeritud päringulogidele, ning
- arhiveerida päringulogisid viisil, mis tagab arhiveeritud sõnumite käsitlemisel samad turvalisuse nõuded, mis on nõutud andmeteenuse kasutamiseks.

## 2.2 Näiteid X-tee V5 keskkonna teenustest

Riigi infosüsteemi haldusüsteemi lehel on kirjeldatud kõik Rahvastikuregistri 145 X-tee teenust. Tabelis on välja toodud teenuse nimi, teenuse kood, versioon ja kirjeldus. Samuti saab vaadata päringu WSDL'i. Päringute täpsem kuvamine on näha Joonisel 2.

### X-tee V5 keskkonna teenused

☰ Ava detailinfo | ☰ Sulge detailinfo

Näitan: 1-50 | Kokku: 145 | Näita kõiki

Nimi	Kirjeldus	Tegevus
isikEpilet	RR58 - Isiku põhielukoha valla koodi ja isikukoodi päring isikukoodi järgiRR58 - Teenus tagastab isikukoodile vastava isiku isikukoodi ja isiku põhielukoha valla/linna EHAK koodi	WSDL+
> Teenuse kood: rr.isikEpilet > Versioon: v1		
isikEpilet_vers2	RR409 - Isiku põhielukoha valla koodi, kehtiva dokumendi ja isikukoodi päring isikukoodi järgiRR409 - Teenus tagastab isikukoodile vastava isiku isikukoodi ja isiku põhielukoha valla/linna EHAK koodi ning kehtiva ID-kaardi, elamisloakaardi või Eesti passi andmed.	WSDL+
> Teenuse kood: rr.isikEpilet_vers2 > Versioon: v1		
isikKodanik	Kodaniku päring iseenda kohta rahvastikuregistrist. Vastavalt rahvastikuregistri seaduse peatükk 12.-le on isikul õigus saada teavet oma alaealiste laste andmete kohta rahvastikuregistris. Laste andmete vaatamiseks peavad rahvastikuregistrisse olema kantud seosed laste ja vanemate vahel. Nimetatud seosed on olemas alates 01.01.2002 sündinud laste ja vanemate andmete vahel. Varem sündinud laste ja vanemate andmete vahel võivad isikuseosed puududa ja nende laste andmeid vanemale päringu tulemusena ei kuvata.	WSDL+
> Teenuse kood: rr.isikKodanik > Versioon: v2		
isikPRIA	RR48 - Isiku põhiantmete ja elukoha päring isikuandmete järgi infosüsteemi jaoks. RR48 - Teenus tagastab isiku registreeritud põhielukoha andmed. Kui isikul puudub RR-s elukoha aadress, tagastatakse tühi string. Kui isiku elukohaks on välisriik, siis väljastatakse riigi nimetus - tänava väärtuseks on aadress tekstina ning maakonna, valla, asula, maja ja korteri väärtuseks on tühi string.	WSDL+
> Teenuse kood: rr.isikPRIA > Versioon: v1		

Joonis 2. Riigi infosüsteemi haldusüsteemi lehel on kirjeldatud kõik 145 rahvastikuregistri X-tee teenust nii neljandas- kui ka viiendas keskkonnas. [8]

## **3 Veebiteenus**

Veebiteenused võimaldavad teenuse pakkujatel ning edasimüüjatel müüa oma teenust läbi Interneti. Veebiteenuste eelisteks on teenuste liideste rakenduste eraldatus ja võime teenust dünaamiliselt siduda eri keeltega. Need eelised pärinevad XML liidesest ja võimaldavad ligipääsu määratud WSDL kirjelduse kaudu. Veebiteenused võivad tuua kokku erinevad rakendused eri platvormidelt, lubades andmebaasi informatsiooni vahetust ning rakenduste kasutamist, mis olid algselt mõeldud sisevõrgus kasutamiseks, kasutada läbi Interneti. [9]

### **3.1 Mis on veebiteenus?**

Veebiteenused on Internetil põhinevad modulaarsed rakendused teenustepõhise arhitektuuri (SOA) kasutamiseks. Veebiteenuseid leitakse UDDI abil ning kirjeldatakse WSDL'is. Süsteemid suhtlevad veebiteenustega SOAP abil ja viisil, mis on määratud selle kirjelduses. Need SOAP sõnumid (nagu ka kõik teised veebiteenuste tehnoloogiad) põhinevad XML-formaadil ja antakse tavaliselt edasi HTTP abil. [10]

### **3.2 Veebiteenuste testimine**

Testimine on väga tähtis faas tarkvara arendustükis. Veebiteenuste testimisel on palju väljakutseid. Teenuste puhul ei tohi testida vaid funktsionaalsust, vaid tuleb testida ka majutust, saadavust erinevatele platvormidele, seadmete ühilduvust, teenuse laadimist jms. [11]

Veebiteenuste populaarsuse tõusu põhjuseks on asjaolu, et saame juba olemasolevaid tarkvararakendusi kasutada ja integreerida viisil, mis laseb luua uusi äriteenuseid. Veebiteenuste testimiseks kasutatakse nii avatud kui ka suletud lähtekoodiga testimisvahendeid, mis erinevad üksteisest mitmete eriomaduste ja funktsioonide poolest. Sisuliselt on testimisvahend programm, mis teostab erinevaid testimise ülesandeid. [12]

Tänapäeval on saadaval on palju veebiteenuste testimisvahendeid. Antud töös kasutatakse soapUI ja JMeter tarkvara. SoapUI on avatud lähtekoodiga tarkvara, mida saab kasutada nii funktsionaalsel- ja mittefunktsionaalsel testimisel. Veebiteenuste testimisel on soapUI võimeline täitma nii kliendi ja teenuse rolli. SoapUI'1 on olemas samuti tasuv versioon soapUIPro, mis annab kasutajale rohkem testimis võimalusi. [13] JMeter on ka avatud lähtekoodiga tarkvara, mis võimaldab koormusteste, jõudlusteste ning funktsionaalteste. Algselt oli JMeter loodud testimaks veebirakendusi, kuid hiljem on laiendanud testimis võimalusi. [14] Kuigi nende vahendite põhieesmärgid on sarnased, erinevad nad üksteisest funktsionaalsuse, eriomaduste, kasutatavuse ja koostalitlusvõimelisuse poolest.



## **4 Tarkvara testimine**

Tarkvara testimine on tegevus, mille käigus käivitatakse programm, et veenduda selle funktsionaalsuses ja õigsuses. Tarkvara testimist viivad enamasti läbi programmeerijad ja testijad ning see on oluline ja keeruline osa tarkvara arendamise protsessist. Testimise eesmärgiks on üles leida probleemid ning tarkvara kvaliteedi parandamiseks need lahendada. [13]

Tarkvara testimise meetodid saab jagada kahte gruppi. Üheks on manuaalne testimine ning teiseks automatiseeritud testimine. Manuaalne testimine on tegevus, mille käigus testimise protsesse viib läbi testija manuaalselt ja tavaliselt järgib testimisplaani, mis on koostatud testjuhtumite põhjal. Testijal on soovituslik kasutada manuaalseks testimiseks musta - või valge kasti tehnikaid. Manuaaltestid aitavad avastada ja raporteerida vigu või mitte oodatud käitumist tarkvara funktsionaalsusest. [13]

Automatiseeritud testide puhul kirjeldatakse testid programmina ning arvuti täidab programmi ning kontrollib tulemusi. Automaattestide olemasolu muudab tarkvara testimise efektiivsemaks ja kiiremaks. Kui on automaattest loodud, siis seda saab käivitada ükskõik millisel ajahetkel. Valideerides mõnda parandust või uuendust tarkvaras ja luues sellele automaatne test, muudab järgnevate muudatuste tegemise lihtsamaks. Automaattestid on sobilikud regressiooni testimiseks, kus testitakse, et juba arendatud funktsioonides ei esine vigu arenduse jätkudes. [14]

### **4.1 Funktsionaalne testimine**

Funktsionaalse testimise eesmärk on leida kinnitust, et tarkvaratoode vastab oma spetsifikatsioonide nõuetele, tarkvara süsteem on kasutatav ning on valmis kliendile edastamiseks. Tarkvara keerukus ja suurus kasvavad pidevalt ning see tähendab, et ka testimine muutub keerukamaks. Tarkvara testimiseks on pakutud erinevaid mooduseid, kuid tarkvara testimise põhiprobleemiks on kõige efektiivsema mooduse valimine. [15]

Üks funktsionaalse testimise viise on musta kasti testimine, kus puudub igasugune teadmine rakenduse sisemisest struktuurist. Funktsionaalsus on selle summa, mida toode, näiteks tarkvararakendus, saab kasutaja heaks teha. [12]

Tavaliselt musta kasti testimisel sisestab testija süsteemi kasutajaliidesesse sisendeid ja vaatleb väljundeid, ilma et ta teaks, kuidas ja kus sisendeid töödeldakse. [14]

Funktsionaalne testimine rõhutab funktsioonide ellu viimist ja sisendite ning väljundite andmete vaatlemist. Testija kohtleb testitavat tarkvara kui musta kasti – vaid sisendid, väljundid ja spetsifikatsioonid on nähtavad, funktsionaalsuse määrab sisenditele vastavate väljundite vaatlemine. Testides kasutatakse mitut sisendit ja väljundeid võrreldakse spetsifikatsioonidega, et kinnitada nende õigsust. Kõik testjuhtumid tuletatakse spetsifikatsioonidest. Ühtegi koodi kasutamise üksikasja ei arvestata. Ilmselgelt leitakse rohkem probleeme, mida rohkem sisendeid kasutatakse ja seega saab tarkvara kvaliteedis kindlam olla. Ideaalis testiksime sisendruumi ammendavalt. Aga nagu eelnevalt mainitud, kehtivate sisendite kombinatsioonide ammendav testimine on enamiku programmide puhul võimatu, rääkimata kehtetutest sisenditest, ajalisest järgnevusest ja ressursse puudutavatest muutujatest. [16]

Raskendatud asjaoluks on spetsifikatsiooni õigsuse ja täielikkuse usaldusväärsus. Spetsifikatsioonides kasutatava keele (tavaliselt loomuliku keele) piirangute tõttu on mitmetähenduslikkus sageli vältimatu. Isegi kui kasutame mingit formaalset või piiratud keelt, ei pruugi meil õnnestuda spetsifikatsiooni kirja panna kõiki võimalikke juhtumeid. Vahel võib spetsifikatsioonist endast saada ületamatu probleem: pole võimalik täpsustada igat olukorda, mis võib esineda kasutades piiratud sõnu. Ja inimesed suudavad harva täpsustada, mida nad soovivad – tavaliselt saavad nad öelda, kas prototüüp vastab nende soovidele alles pärast selle valmimist. Spetsifikatsioonide probleemid moodustavad umbes 30% tarkvarade vigadest. [16]

## **4.2 JMeter**

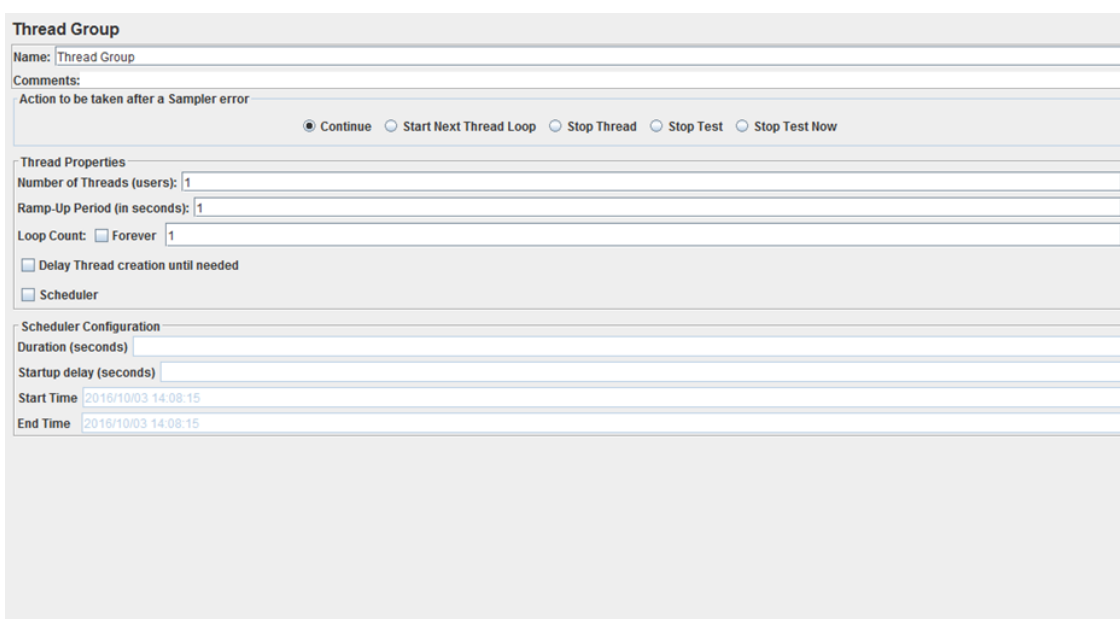
JMeter on avatud lähtekoodiga testimisvahed, mille arendajaks on Apache Software Foundation. Seda levitatakse Apache litsentsi all. Esialgu kavandati see veebirakenduste testimiseks, kuid selle tegevusala laiendati teiste testimisfunktsioonideni. JMeter'i põhifunktsioon on teha koormusteste klient/server rakendustele, aga seda saab kasutada

ka jõudluse mõõtmiseks ning funktsionaalsuse testimiseks. Lisaks sellele, JMeter on kasulik regressioontestimisel, kus tõeste väidetega aitab JMeter kaasa test-skriptide loomisele. Sel viisil saadakse kinnitust, et rakendus annab ootuspärased tulemused. [13]

Kõige lihtsam testimisplaan koosneb tavaliselt järgmistest elementidest: [17]

- Harude grupp (*Thread group*)
- Testitav element (*Sampler*)
- Kuulajad (*Listeners*)

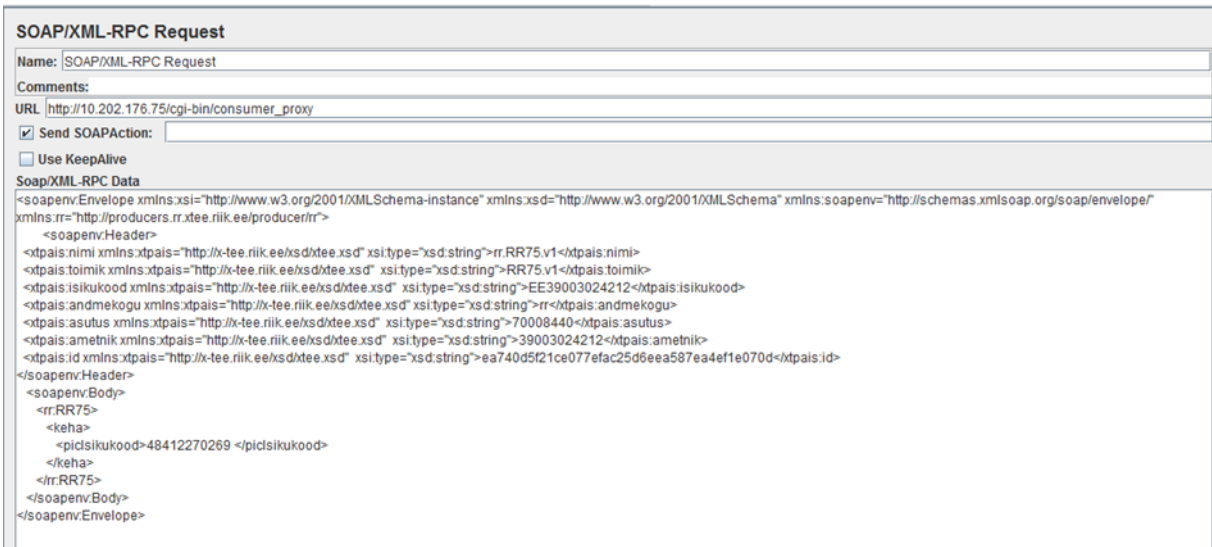
Harude grupp (*Thread group*) on elemendid, mis täpsustavad jooksvate harude arvu ning käivitusperioodi. Iga haru simuleerib kasutajat ning käivitusperiood täpsustab aega, mis kulub kõikide harude loomiseks. Näiteks 5 haruga ja 10 sekundilise käivitusperioodiga on iga haru loomise vahel 2 sekundit. Tsüklite arv määrab haru jooksmise aja. Graafik laseb sul määrata ka jooksmise algus- ning lõpuaja. [17] Joonisel 3 on määratud 1 haru, käivitusperiood 1 sekund ja test käivitatakse 1 kord.



The image shows the 'Thread Group' configuration window in JMeter. It includes fields for Name, Comments, and an 'Action to be taken after a Sampler error' section with radio buttons for Continue, Start Next Thread Loop, Stop Thread, Stop Test, and Stop Test Now. The 'Thread Properties' section contains fields for Number of Threads (users) set to 1, Ramp-Up Period (in seconds) set to 1, and Loop Count set to 1 with the 'Forever' checkbox selected. There are also checkboxes for 'Delay Thread creation until needed' and 'Scheduler'. The 'Scheduler Configuration' section has fields for Duration (seconds), Startup delay (seconds), Start Time (2016/10/03 14:08:15), and End Time (2016/10/03 14:08:15).

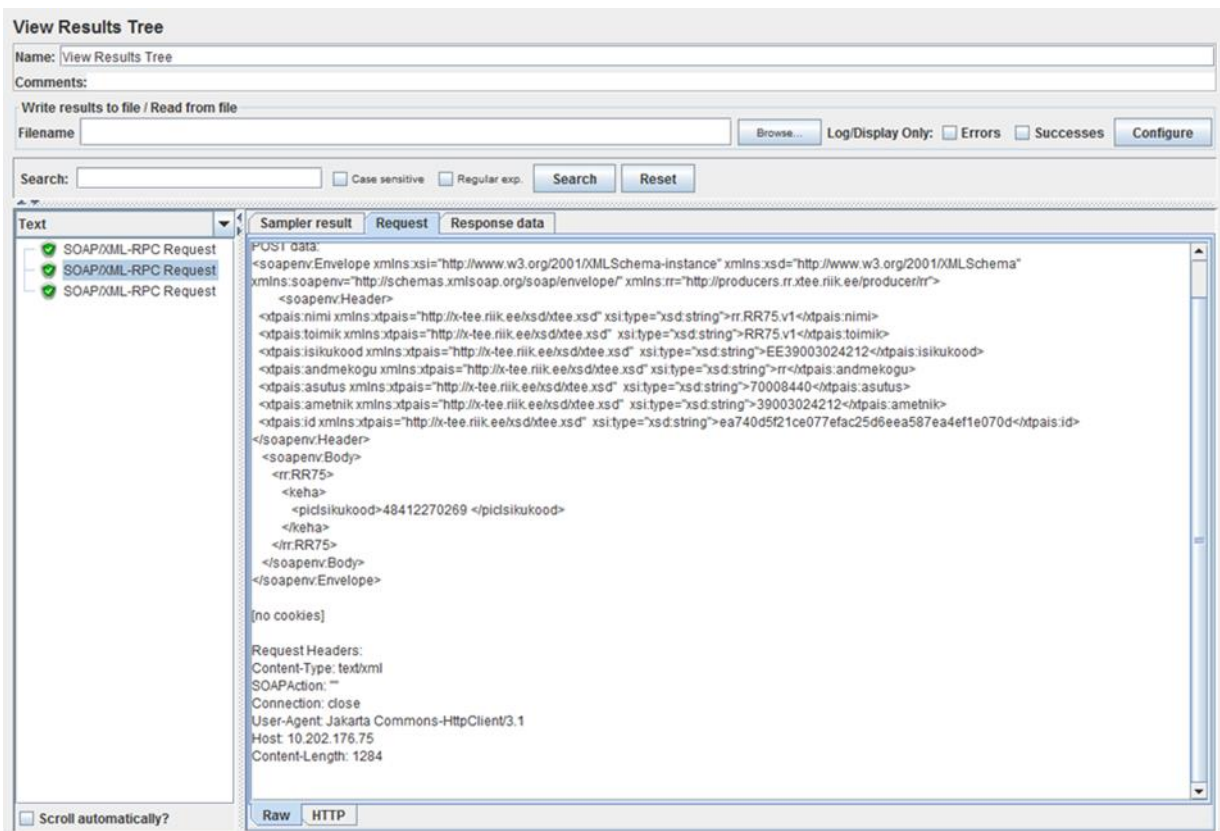
Joonis 3. Harude grupp (*Thread Group*) vaade.

Testitav element (*Sampler*) konfigureerib päringuid serverile – HTTP, FTP või LDAP päringud. [17] Joonisel 4 on SOAP päring. Määratud on päringu nimi, turvaserveri URL ja päring RR75 andmed.



Joonis 4. Testitav element (*Sampler*) „SOAP/XML-RPC Request“ vaade.

Kuulajad (*Listeners*) on elemendid, mis järeltöötlevad päringu andmeid. Näiteks saab andmeid faili juurde salvestada või tulemusi tabelina kujutada. Hetkel ei paku JMeter'i tabel väga palju konfiguratsiooni võimalusi, kuid see on laiendatav ja alati on võimalus lisada mõni visuaal või andmetötlusmoodul. [17] Joonisel 5 on kasutatud „View Results Tree“ vaadet. Roheline kilbi ikoon annab märku, et test on läbitud edukalt.



Joonis 5. Kuulajad (*Listeners*) „View Results Tree“ vaade.

*XPath Assertion* testib dokumendi vormistust töödeldes seda läbi *JTidy* ja testides *XPath*'i. Kui *XPath* eksisteerib, siis sel juhul on „*Assertion*“ „*true*“. Kasutades „/“, saab vastuse igasse hästi vormistatud dokument. [18]

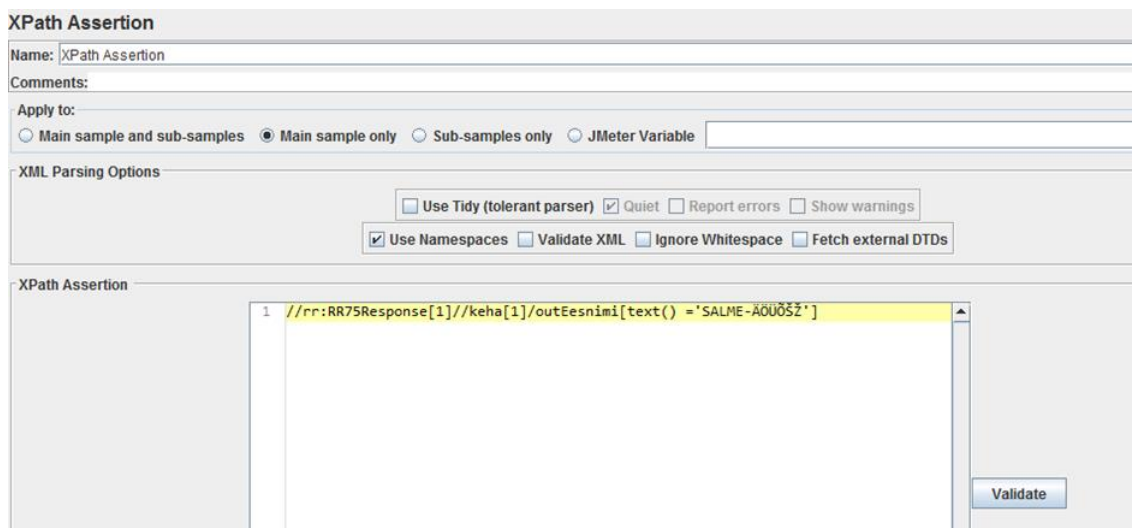
Antud töös kasutati ainult *XPath Assertion*'it.

Mõned näited:

```
//rr:RR75Response[1]/keha[1]/outAK[text() = '176']
```

```
//rr:ParingHaridusametileResponse[1]//keha[1]/Isik[1]/item[1]/Isik.Surmakuup[1]
```

Joonisel 6 otsitakse eesnime oodatud tulemuseks. Kindlasti tuleb antud aknas märkida linnuke „Use Namespaces“, et JMeter saaks kasutada nimeruume, mis on määratud failis „Jmeter.properties“.



Joonis 6. *XPath Assertion* vaade.

JMeter on suure laiendatavusega, kuna kasutab lisatavaid komponente. Nende lisatavate komponentide hulka kuuluvad taimerid, testitav element (*samplerid*) ja visuaalide pluginad. JMeter pakub arvukalt statistilisi raporteid kui ka graafilist analüüsi. [13] Hetkel on kõige uuem versioon 3.0.

### 4.3 SoapUI

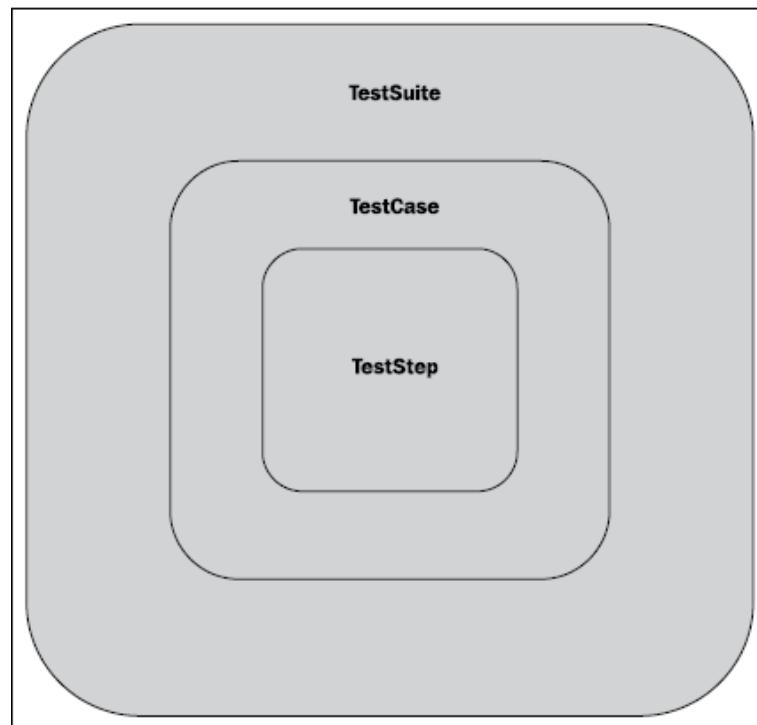
SoapUI on avatud lähtekoodiga teenuste põhise arhitektuuri (SOA) ja veebiteenuste testimisvahend. Selle arendajaks on SmartBear Software. SoapUI aitab kiirelt luua arenenud jõudluste ja läbi viia automatiseeritud funktsionaalsuse teste. SoapUI pakutavad eriomadused aitavad hinnata veebiteenuste jõudlust. Testitulemuste analüüs annab võimaluse teenuste ja rakenduste kvaliteeti parandada. SoapUI pakub

kasutajasõbralikku graafilist kasutajaliidest ja suudab läbi viia erinevaid teste. [13] Hetkel on kõige uuem versioon 5.3.

SoapUI funktsionaalne test koosneb kolmest võtme elemendist: [19]

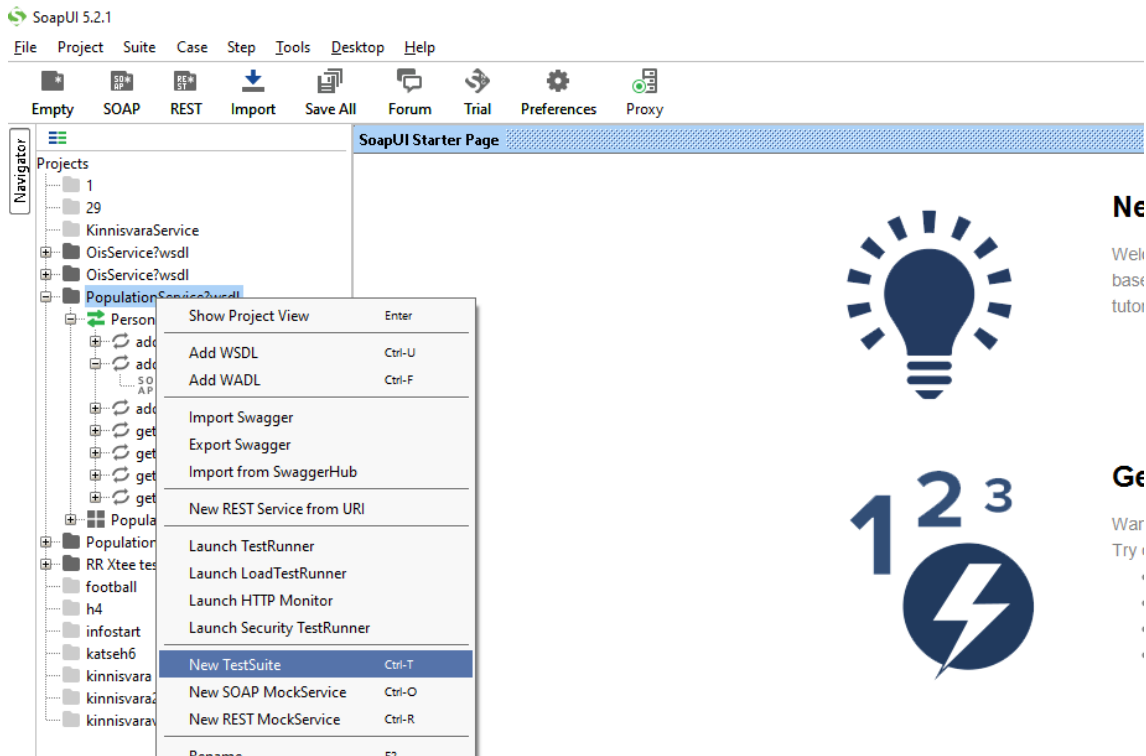
- Testkogum (*TestSuite*)
- Testjuhtum (*TestCase*)
- Testsammud (*TestStep*)

Elementide struktuuri soapUI projektis saab esitada järgneva diagrammina joonis 7.



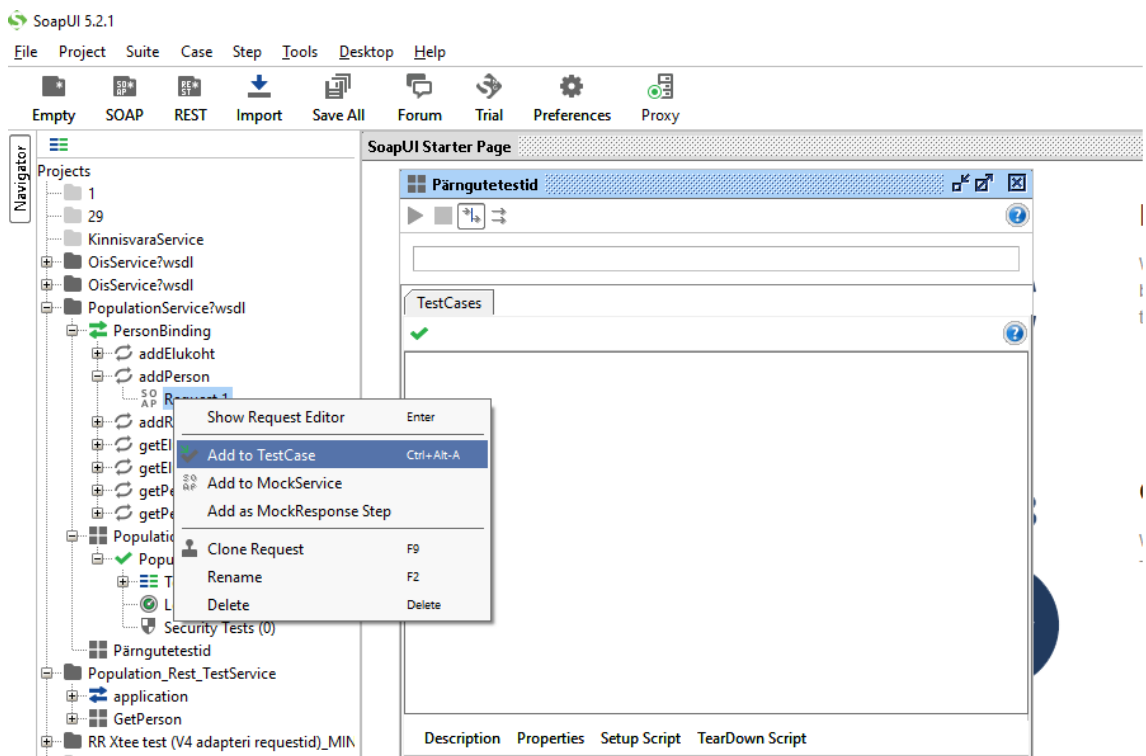
Joonis 7. SoapUI elementide diagramm. [20]

Testkogum (*TestSuite*) on kollektsoon testjuhtumitest, et grupeerida funktsionaalseid teste loogilisteks üksusteks. SoapUI projekti saab lisada lugematu arv testkogumeid, et viia läbi mahukaid testimisprotsesse. [19] Joonisel 8 lisatakse projektile uus testkogum.



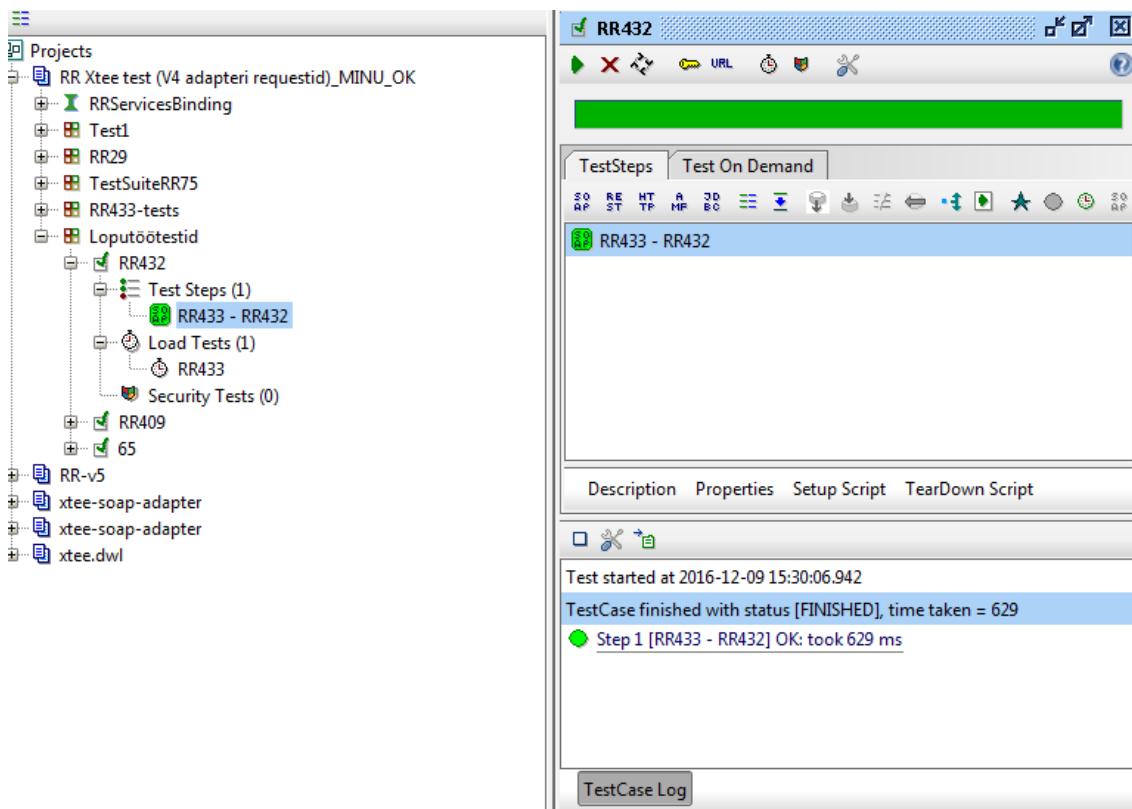
Joonis 8. Testkogumi (*TestSuite*) lisamine.

Testjuhtum (*TestCase*) on kollektsoon testsammudest (*TestStep*), mis on kokku pandud, et testida spetsiifilise aspektiga teenust. Võimalik on lisada lugematu arv testjuhtumeid. [19] Joonisel 9 lisatakse päringule testjuhtum.



Joonis 9. Testjuhtumi (*TestCase*) lisamine.

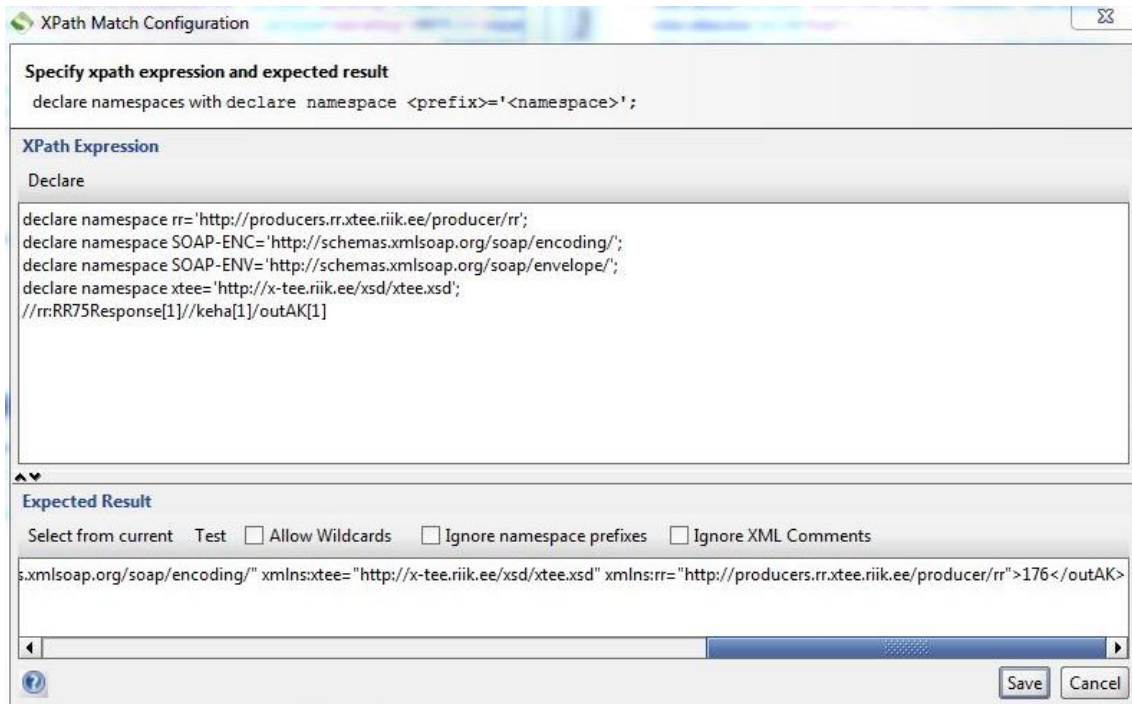
Testsammud (*TestStep*) on "põhiosad" soapUI funktsionaalses testimises. Need lisatakse testjuhtumi ning kasutatakse käivitamise ja valideerimise funktsionaalsuse testimise kontrolliks. [19] Joonisel 10 on testsammude vaade. Antud testis on üks testsamm.



Joonis 10. Testsammude (TestSteps) vaade.

*XPath Assertion* kasutatakse, et võrrelda *XPath expression* tulemust eeldefineeritud väärtusega. Kindlasti tuleb deklareerida nimeruumid, mida kasutakse *expression*'is. Nimeruumid deklareeritakse automaatselt, kui vajutada nuppu „*Declare*“. Samuti kuvatakse oodatav väärtus, kui vajutada nuppu „*Expected Result*“. [20] Joonisel 11 on määratud nimeruumid vajutades nuppu „*Declare*“. Kasutatud on teenust RR75. Joonisel on kuvatud oodatud tulemus, mis on saadud „*Select from current*“ nupu vajutamiseiga.





Joonis 11. XPath Assersation deklareerimine.

## 4.4 SoapUI ja JMeter'i võrdlus

SoapUI ja JMeter on konkureerivad veebiteenuste testimise tarkvarad. Mõlemad on avatud lähtekoodiga. Selleks, et nende pakutavaid võimalusi paremini mõista, tuleb neid erinevate näitajate alusel võrrelda. Tarkvarade võrdluseks kasutatud näitajad on valitud, arvestades mõlema tarkvara ühiseid jooni. Kuna lõputöö autor on testinud veebiteenuseid mõlema tarkvaraga, siis lähtutakse ka autori tähelepanekutest ja kogemustest.

### Kasutusjuhendite kättesaadavus Internetis

SmartBear'i poolt on tasuta kättesaadav e-raamat „Beginners's Guide to functional Testing“. Raamat koosneb viiekümnest leheküljest. Antud raamat on mõeldud soapUI Pro'le, aga annab siiski ülevaate soapUI funktsioonidest. Palju kasutusjuhendeid on ka soapUI kodulehel, mis on hästi koostatud.

JMeter'i kohta on väga põhjalik kasutusjuhend JMeter'i kodulehel.

Tuleks ka ära märkida veebisaidi YouTube juhendid. SoapUI kohta on YouTube's ligikaudu 13 600 videot ning JMeter'i kohta 13 700 videot. Ka Google'i otsing annab muljetavaldavaid tulemusi: JMeter'i puhul leiti vasteid 1 690 000 ja soapUI puhul 924 000.

Siiski tuleb tunnistada, et soapUI õppimine võtab vähem aega tänu kergesti kasutatavale kasutajaliidesele.

### **Kasutusmugavus**

Kasutusmugavuse hindamisel on lähtunud töö autori tähelepanekutest.

SoapUI puhul kuvatakse nii päring kui ka vastus silmale loetavamalt. Tekstis on kasutatud erinevaid värve. Vastus kuvatakse nii, et iga märgis (*tag*) on uuel real. JMeter kuvab nii vastuse kui ka päringu ühe värviga ning vastus on kuvatud ühtse tekstina, mis ei ole korrastatud märgiste (*tag*) järgi. SoapUI's on *XPath Assertion* testimise üheks eeliseks nupp *Expected Result*, mis annab oodatud tulemuse automaatselt. JMeter'is tuleb oodatud tulemus ise sisestada. Teiseks eeliseks on uue *Assersation* 'i lisamine, mis on soapUI's mugavam. JMeter'i ühe eelisenä võib välja tuua, et nimeruum määratakse ühe korra failis „JMeter.properties“, kuid SoapUI'ga tuleb iga testjuhtumi puhul vajutada nuppu *Declare*. Teiseks JMeter'i plussiks on navigeerimine erinevate akende vahel, mis annab võimaluse teksti kopeerida. SoapUI'ga aga ei anna teksti kopeerida, kui *Assertion* aken on avatud.

### **Tarkvara installeerimise võimalused**

Mõlemad tarkvarad on tasuta kättesaadavad ning avatud lähtekoodiga.

SoapUI kasutamiseks tuleb arvutisse installeerida Java Development Kit (JDK) v1.6+. SoapUI on ehitatud Java platvormile, tänu sellele saab seda kasutada paljudes operatsioonisüsteemides nagu Windows XP, Windows Vista, Windows 7, Windows Server 2003, Windows Server 2008, erinevad Linuxi süsteemid nagu Ubuntu, Red Hat, Fedora, SuSe ja CentOS ning Mac OS X v10.4 ja kõrgemad. [20]

JMeter'i kasutamiseks piisab JVM 1.3 olemasolust arvutis. Kui aplikatsioon nõuab testimiseks HTTPS protokollituge, siis sellisel juhul tuleb installeerida ka JSSE, sest JVM 1.3 ei toeta seda. Kuna JMeter on Java-põhine, saab seda käivitada paljudes operatsioonisüsteemides nagu ka soapUI'd, näiteks UNIX, Linux, Open VMS Alpha 7.3+ ja Windows (98 või kõrgem). JMeteri'it saab kasutada igal platvormil, mis võimaldab käivitada JVM. [21]

## WSDL loomine

SoapUI suudab SOAP päringu automaatselt ise genereerida WSDL failist, kuid JMeter'ga peab arendaja ise ehitama näidispäringu XML'i.

## Teenuse simulatsioon

SoapUI'ga on võimalus simuleerida veebiteenust enne, kui teenus on liidestatud. See võimaldab testida veebiteenust kasutajavaates, isegi kui konkreetne veebiteenus hetkel puudub.

## Testimisvõimalused

SoapUI'ga saab testida erinevaid veebiteenuseid nagu SOAP-l põhinevaid veebiteenuseid, RESTful veebiteenuseid, JMS, HTTP veebiteenuseid, andmebaase ja palju muud. SoapUI'ga on võimalik läbi viia koormus-, turva-, funktsionaalseid ja palju muid teste. [20]

JMeter toetab funktsionaalseid, koormus-, regressiooni- ja stressiteste erinevatel protokollidel. Samuti toetavad osad protokollid HTTP, HTTPS, SOAP, REST, JMS teste. SoapUI ja JMeter lubavad kasutada isetehtud skripte, mille abil on võimalik funktsioone laiendada. [21]

Tabel 1. SoapUI ja JMeter'i võrdlus

Võrreldavad näitajad	SoapUi	JMeter
Kasutusjuhendite kättesaadavus	Väga hea	Väga hea
Kasutusmugavus	Lihtsam	Keerulisem
Tarkvara installeerimise võimalused	Väga hea	Väga hea
WSDL loomine	Lihtsam	Keerulisem
Teenuse simulatsioon	Olemas	Puudulik
Testimisvõimalused	Väga hea	Väga hea

## 4.5 SoapUI ja JMeter'i tehniline võrdlus

Tabel 2. SoapUI ja JMeter'i tehniline võrdlus

<b>Tool</b>	<b>First release</b>	<b>Latest Version</b>	<b>Programming language</b>	<b>Operating Sytem Support</b>	<b>Require - ment</b>	<b>Developed By</b>	<b>Lisence</b>
JMeter	2001	3.0	Java	Cross – platform	JRE 1.5 +	Apache Software Foundation	Apache License 2.0
soapUI	2005	5.3	Java	Cross – platform	JRE 1.6+	SmartBear Software	GNU LGPL 2.1

## 5 Testimine

### 5.1 Päringu RR76AndmIK spetsifikatsiooni kirjeldus

Rahvastikuregistri X-tee teenuste spetsifikatsioone on kirjeldatud sarnaselt. Alljärgnevalt kirjeldatakse näitena spetsifikatsiooni RR76AndmIK.

Spetsifikatsioonis on välja toodud päringu ja teenuse nimetus, kasutaja, tarkvara moodul, olemasoleva päringu kirjeldus, sisend, tingimused, väljund ja vead. Kui sisend on vigane, annab päring vastuse „Päring ei saanud vastust. Andmed puuduvad.(10010)“. Väljundandmed on toodud välja tabelina, mille veergude pealkirjad on „Nimetus“, „Tabel“, „Tingimus“, „Väärtus“ ja „Taginimetus“.

Veerus „Nimetus“ kirjeldatakse, millised andmed väljastatakse tekstina. Veerus „Tabel“ tuuakse välja andmetabelid, kust andmed on võetud. Veerus „Tingimus“ kirjeldatakse, mis tingimustel andmed väljastatakse. Veerg „Väärtus“ annab ülevaate, mis väärtus andmebaasi poolt väljastatakse. Veerus „Taginimetus“ kirjeldatakse, millise nimetusega märgisega (*tag*) andmed väljastatakse.

Dokumendi lõpus on välja toodud näite andmed (vt Lisa 1).

### 5.2 Päringu RR76AndmIK testjuhtumid

Edukaks testimiseks tuleb alati koostada testjuhtumid, et kontrollida, kas loodud tarkvara vastab ootustele. Järgnevalt tuuakse näitena välja kaks testjuhtumit. Esimene testjuhtum tagastab isiku andmed ning teine testjuhtum lõppeb veateatega.

Tabelis 3 on toodud testjuhtum, mille puhul päring tagastab andmed.

Tabel 3. Testjuhtum 1.

Test-stsenaarium	Test-juhtum	Eeltingimus	Test-sammud	Test-andmed	Oodatud tulemus
Kontrolli, kas isikukood väljastab õiged andmeväljad	Kontrolli tulemust sobiva isikukoodi sisestamisel	Tarkvaras olemas teenus RR76 ning kasutajal olemas õigus antud teenust kasutada	1. käivitada tarkvara 2. valida päring RR76 3. sisestada isikukood 4. vajutada start nuppu	Isikukood: 37904030074	1. isiku perenimi 2. isiku eesnimi 3. isiku olek 4. andmete olek 5. isiku sugu 6. isiku sünnipäev 7. surmakuupäev 8. kodakondsus 9. isiku sünnikoha EHAK kood 10. sünnikoht 11. põhielukoha EHAK kood 12. elukoht tekstina

Tabelis 4 on toodud testjuhtum, mille puhul päring tagastab veateate

Tabel 4. Testjuhtum 2

Test-stsenaarium	Test-juhtum	Eeltingimus	Test-sammud	Test-andmed	Oodatud tulemus
Kontrolli, kas vigane isikukood annab veateate	Kontrolli tulemust vigase isikukoodi sisestamisel	Tarkvaras olemas teenus RR76 ning kasutajal olemas õigus antud teenust kasutada	1. käivitada tarkvara 2. valida päring RR76 3. sisestada isikukood 4. vajutada start nuppu	Isikukood: 97904030074	Päring ei saanud vastust. Andmed puuduvad

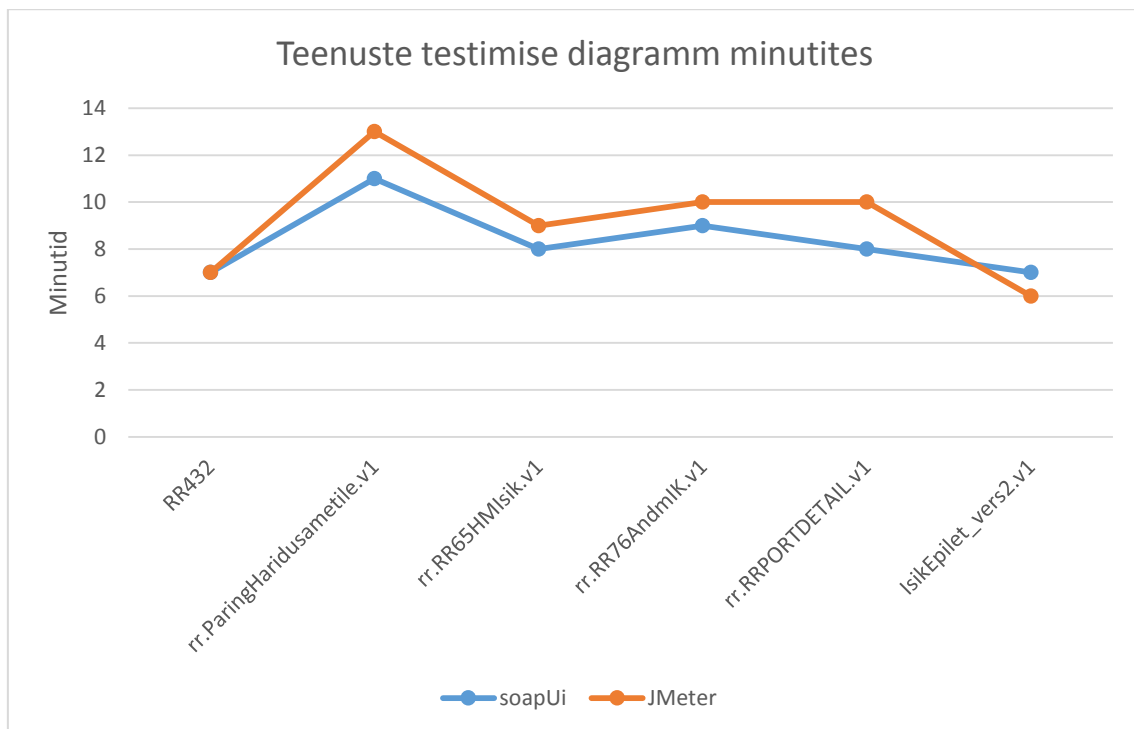
### 5.3 Testide läbiviimine

Testimisprotsess viidi läbi SMIT'is. Antud töös testiti 6. rahvastikuregistri X-tee teenust. Igal teenusel oli rohkem kui kaheksa andmevälja. Esiteks tutvuti teenuste spetsifikatsioonidega ning koostati testjuhtumid. Testimist mõõdeti ajas minutites. Aega hakati lugema esimese teenuse testimisest ja lõpetati viimase päringu testimisega. Kõik teenuse andmeväljad pidid tagastama oodatud tulemuse, mida kontrolliti *XPath*

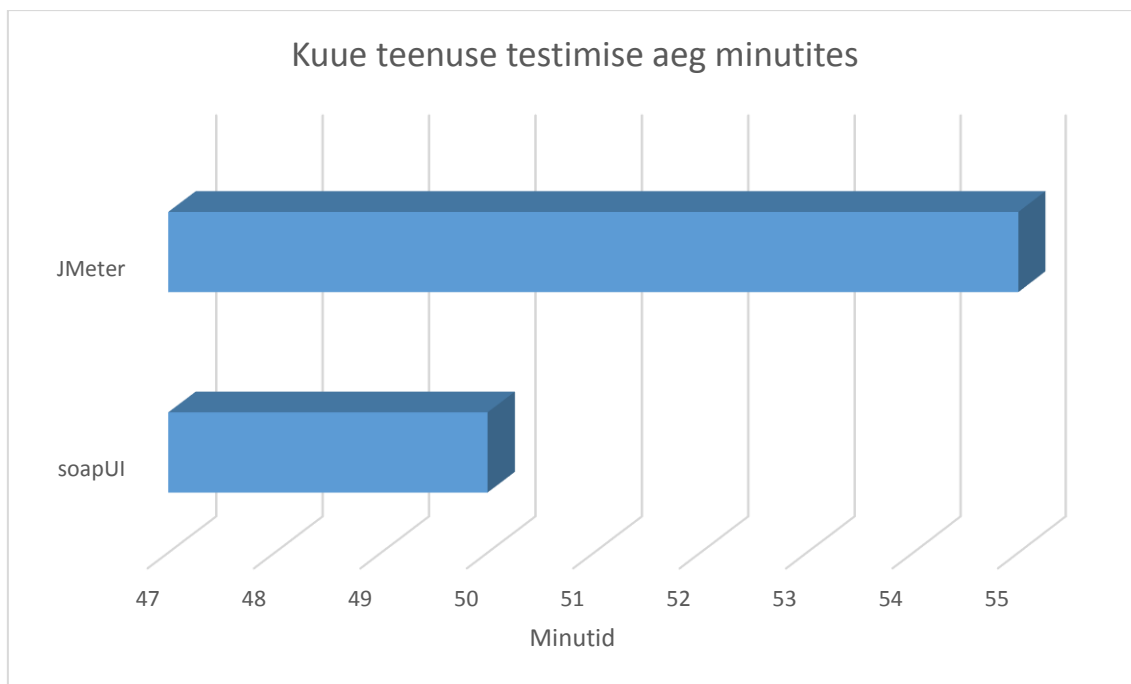
*Assertion*'i abiga. Antud tulemused kanti vastavasse tabelisse, kuhu märgiti testimiseks kulunud aeg ning andmeväljade arv.

Vastavad tulemused on esitatud joonisel 12. Joonisel on välja toodud testimiseks kulunud aeg teenuste lõikes. Teenuse RR432 testimiseks kulus nii soapUI kui JMeter'iga 7 minutit. Päringus, kus andmevälju oli rohkem osutus kiiremaks soapUI tänu *Excepted Results* nupu olemasolule. Päringu IsikEpilet testimiseks kulus JMeter'iga vähem aega. Põhjuseks on märgiste (*tag*) pikad nimed, mis soapUI-d kasutades tuleb käsitsi sisestada, kuid JMeter'iga saab kopeerida, kui aken *Assersation* on avatud.

Joonisel 13 on diagrammina esitatud kuue teenuse testimise aeg minutites nii soapUI kui JMeteri'iga.



Joonis 12. Testimistulemused kuue teenuse lõikes minutites



Joonis 13. Kuu teenuse testimistulemus minutites

Antud testimistulemuste põhjal saame arvutada, kui kaua kulub keskmiselt aega 145 teenuse testimiseks.

$$jm\_koef = (kogu\_aegJM)/(and\_arv + testj\_arv)$$

kus

*jm\_koef* – JMeter'i koefitsient

*kogu\_aegJM* – JMeter'iga testimiseks kulunud aeg

*and\_arv* – teenuse andmeväljade arv

*testj\_arv* - testitud testjuhtumite arv

$$sUI\_koef = (kogu\_aegSUI)/(and\_arv + testj\_arv)$$

kus

*sUI\_koef* – soapUI koefitsient

*kogu\_aegJM* – soapUI'iga testimiseks kulunud aeg

*and\_arv* – teenuse andmeväljade arv

*testj\_arv* - testitud testjuhtumite arv

$$test\_kuluvAegJm = teenusteArv * andme\_kesk * testj\_arv * jm\_koef$$

kus

*test\_kuluvAegJm* – JMeter'iga testimiseks kuluv aega minutites



*teenusteArv* – rahvastikuregistri teenuste arv  
*andme\_kesk* – keskmine andmeväljade arv ühes teenuses  
*testj\_arv* - testjuhtumite arv ühes teenuses

*test\_kuluvAegSUI* = *teenusteArv* \* *andme\_kesk* \* *testj\_arv* \* *sUI\_koef*  
kus

*test\_kuluvAegSUI* – soapUI'iga testimiseks kuluv aega minutites  
*teenusteArv* – rahvastikuregistri teenuste arv  
*andme\_kesk* – keskmine andmeväljade arv ühes teenuses  
*testj\_arv* - testjuhtumite arv ühes teenuses

Esiteks leiame JMeter koefitsiendi kuue teenuse testimise põhjal. Teenuste testimiseks kulus 55 minutit, testiti 2. testjuhtumit ning 72 andmevälja *XPath Assertion*'iga. Seega saab koefitsiendiks

$$jm\_koef = 55 / (72 + 12) = 0.654$$

Teiseks leiame soapUI koefitsiendi 6 teenuse testimise põhjal. Teenuste testimiseks kulus 50 minutit, testiti 2. testjuhtumit ning 72 andmevälja *XPath Assertion*'iga. Seega saab koefitsiendiks

$$sUI\_koef = 50 / (72 + 12) = 0.595$$

Nüüd on võimalik leida kui kaua kulub keskmiselt aega 145 teenuse testimiseks. Oletades, et keskmiselt on teenustel 16 andmevälja ning testiti kõiki teenuseid 2. testjuhtumiga.

JMeter'iga saab tulemuseks

$$test\_kuluvAegJm = 145 * 16 * 2 * 0.654 = 3038 \text{ minutit} = 50 \text{ tundi}$$

SoapUI'ga saab tulemuseks

$$test\_kuluvAegSUI = 145 * 16 * 2 * 0.595 = 2761 \text{ minutit} = 46 \text{ tundi}$$

Antud tulemustest selgub, et soapUI'ga kuluks teenuste testimiseks **4 tundi** vähem aega.

## 6 Kokkuvõte

Antud töö eesmärkideks oli kirjeldada, mis on X-tee ning miks on see turvaline keskkond andmevahetuseks. Samuti selgitada, kuidas veebiteenuseid testida ning mis on funktsionaalne testimine.

Töö põhieesmärgiks oli leida tarkvara, millega oleks kiirem ning mugavam testida rahvastikuregistri X-tee teenuseid. Leidmaks sobivat tarkvara tutvus töö autor erinevate rakendustega ning uuris, mis võimalusi antud tarkvarad pakuvad. Sobivaks osutusid avatud lähtekoodiga, funktsionaalset testimist võimaldavad ning *XPath Assersation* funktsionaalsust pakuvad tarkvarad. Võrdluse tulemusel jäi sobilike valikutena alles soapUI ja JMeter. Testimine viidi läbi 6. juhusliku rahvastikuregistri X-tee teenusega. Testimist mõõdeti ajas minutites ning tulemused kanti vastavasse tabelisse, mis hiljem esitati diagrammidena. Testimistulemuste põhjal arvutati, kui kaua kulub keskmiselt aega 145 teenuse testimiseks, kui ühel teenusel on umbes 16 andmevälja ja 2 testjuhtumit. Tulemustest selgus, et soapUI-ga kulub testimiseks 4 tundi vähem aega.

SoapUI funktsionaalsuse testimise eelisteks saab välja tuua järgmised omadused:

- Teksti kuvamine on silmale loetavam
- *Expected Results* nupu olemasolu aknas *Assersation*
- Mugavam lisada uut *Assertion*'i

Lõputöö eesmärgid saavutati toetudes välistele infoallikatele, testimisele ja autori tähelepanekutele.

Töö võimalikuks edasiarenduseks võiks olla rahvastikuregistri X-tee teenuste jõudluse testimine. Tulemuste põhjal saaks leida, milline on keskmine päringu vastuse aeg erinevatel ajahetkedel. Testida saaks erinevate tarkvaradega ning neid võrrelda.

Antud töö oli eriti oluline töö autorile, kes hakkab rahvastikuregistri X-tee teenuseid testima. Töö autor jõudis ka järeldusele, et soapUI on funktsionaaltestimiseks mugavam, ning kasutab edaspidi X- tee teenuste testimiseks just viimast.

## Kasutatud kirjandus

- [1] Rahvastikuregister, [WWW] <https://www.siseministeerium.ee/et/tegevusvaldkonnad/rahvastikutoimingud/rahvastikuregister>. (14.09.2016).
- [2] Üleminek X-tee versioonile 6, [WWW] <https://www.ria.ee/ee/uleminek-x-tee-versioonile-6.html> (12.09.2016).
- [3] Riigi Infosüsteemi Amet, [WWW] [https://www.ria.ee/public/x\\_tee/X-tee\\_teenuste\\_kirjeldamise\\_parimad\\_praktikad\\_ja\\_tuupilised\\_vead\\_v1.0.pdf](https://www.ria.ee/public/x_tee/X-tee_teenuste_kirjeldamise_parimad_praktikad_ja_tuupilised_vead_v1.0.pdf) (20.09.2016).
- [4] Riigi Infosüsteemi Amet, [WWW] <https://www.ria.ee/ee/x-tee-tutvustus.html> (22.09.2016).
- [5] Riigi Infosüsteemi Teejuht, [WWW] <https://www.ria.ee/teejuht/riigi-infosusteemi-olemus-ja-komponendid/infosusteemide-andmevahetuskiht-x-tee> (22.09.2016).
- [6] RIA X-tee rakendusjuhise, [WWW] <https://moodle.ria.ee/mod/page/view.php?id=268> (22.09.2016).
- [7] Osaliste õigused ja kohustused X-teel, [WWW] <https://moodle.ria.ee/mod/book/tool/print/index.php?id=343> (22.09.2016).
- [8] RIHA, [WWW] <https://riha.eesti.ee/riha/main> (22.09.2016).
- [9] C. Ferris and J. Farrell. What are Web Services? [WWW]. <http://d.web.umkc.edu/di5x7/output/Paper%20Critique%20-%20Web%20Services.pdf> (12.11.2016).
- [10] S. Hanna, M. Munro. Fault-based Web Services Testing [WWW] <http://ieeexplore.ieee.org/document/4492524/> (12.09.2016).
- [11] R. Eeratta, S. S. Shenoy, C R Vijeth, N. John. Service Test Automation Framework [WWW] <http://ieeexplore.ieee.org/document/7019575/> (12.09.2016).
- [12] S. Sharma and Dr. A. K. Sharma. Empirical Analysis of Web Service Testing Tools [WWW] <http://www.ijetmas.com/admin/resources/project/paper/f201504171429253273.pdf> (12.11.2016).
- [13] S. Hussain, Z. Wang, I. K. Toure, A. Diop. Web Service Testing Tools: A Comparative Study [WWW] <https://arxiv.org/abs/1306.4063>. (12.11.2016).
- [14] T.Amruthavalli, S.MahaLakshmi, K.HariKrishnan. A Review On Software Testing In SDIC And Testing [WWW] <http://www.ijecs.in/issue/v3-i9/44%20ijecs.pdf>. (12.11.2016).
- [15] S. Xu, L. Chen, C. Wang, O. Rud. A Comparative Study on Black-Box Testing with Open Source Applications [WWW] <http://ieeexplore.ieee.org/abstract/document/7515953/>. (11.12.2016).

- [16] J. Pan. Software Testing [WWW]  
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.103.7121&rep=rep1&type=pdf> (15.11.2016).
- [17] D. Nevedrov. Using JMeter to Performance Test Web Services [WWW]  
<http://loadstorm.com/files/Using-JMeter-to-Performance-Test-Web-Services.pdf> (12 .11. 2016)
- [18] JMeter [WWW] [http://jmeter.apache.org/usermanual/component\\_reference.html](http://jmeter.apache.org/usermanual/component_reference.html)  
(23.09.2016).
- [19] SoapUI [WWW] <https://www.soapui.org/functional-testing/structuring-and-running-tests.html> (12.12.2016).
- [20] C. Kankanange, Web Services Testing with soapUI, BIRMINGHAM: PACKT, 2012.
- [21] B. Erinle, Performance Testing with JMeter, Packt Publishing, 2015.
- [22] X-tee tutvustus [WWW] <https://www.ria.ee/ee/x-tee-tutvustus.html> (20.09 2016).

## Lisa 1 – spetsifikatsioon RR76

76 Isiku nime, sünnikuupäeva, surmakuupäeva, kodakondsuse, elukohta, sünnikoha ja staatuse väljastamine isikukoodi põhjal

**Päringu nimetus:** RR76 – Isiku nime, sünnikuupäeva, surmakuupäeva, kodakondsuse, elukohta, sünnikoha ja staatuse väljastamine isikukoodi põhjal.

**Kasutaja:** X-TEE

**Teenuse nimetus:** RR.RR76AndmIK

**Tarkvara moodul** rr76.p

**Olemasoleva päringu kirjeldus:** RR76KrimiRaivoleKirjeldus.doc

**Sisend:** Otsitava isiku isikukood

**Tingimused:**

Isik.IsIsikukood=Sisend

**Väljund:**

- perenimi
- eesnimi
- isiku olek
- andmete olek
- sugu
- sünnikuupäev
- surmakuupäev
- kodakondsuse kood
- kodakondsuse nimetus
- sünnikoha valla või asula kood EHAK-i järgi
- sünnikoht tekstina
- isiku põhielukohta valla või asula kood EHAK-i järgi
- elukoht tekstina

### Vead

- Päring ei saanud vastust. Andmed puuduvad. (10010)

Nimetus	Tabel	Tingimus	Väärtus	Tagi - nimetus
Isiku perenimi	Isik		IsPerenimi	Isikupnimi
Isiku eesnimi	Isik		IsEesnimi	Isikuenimi
Isiku olek	Kodifikaator	Kodifikaator.KdID=Isik.KdIDIsikuStaat	KdLyhikeNimi	Isikuolek
Isiku andmete olek	Kodifikaator	Kodifikaator.KdID=Isik.KdIDKirjeStaat	KdLyhikeNimi	Kirjestaatus
Isiku sugu	Kodifikaator	Kodifikaator.KdID=Isik.KdIDSugu	KdElemendiKood	Isikusugu

Isiku sünnikuupäev	Isik		IsSynniaeg	Isikusynnikip
Isiku surmakuupäev	Isik		IsSurmaaeg	Isikusurmakp
Isiku kodakondsus (kood) (võib olla mitu)	Kodifikaator	Kodakondsus.IsID = Isik.IsID Kodakondsus.KodKehtib Kuni >=current date või NULL Kodifikaator.KdID = Kodakondsus.KdIDRiik Väljastada kõik isiku kodakondsuse riigi koodid eraldajaks – „,“	KdLisaKood (3 täheline kood)	Isikukodak
Isiku kodakondsus (nimetus) (võib olla mitu)	Kodifikaator	Kodakondsus.IsID = Isik.IsID Kodakondsus.KodKehtib Kuni >=current date või NULL Kodifikaator.KdID = Kodakondsus.KdIDRiik Väljastada kõik isiku kodakondsused eraldajaks – „,“	KdLyhikeNimi	Isikukodaknm
Isiku sünnikohavalla või asula kood EHAk-i järgi	AadressKomponent	IsikuAadress.IsID = Isik.IsID Kodifikaator.KdID = IsikuAadress.AadressiLiik , Kodifikaator.KdLyhikeNimi = „Sünnikoht” Aadress.AdrID = IsikuAadress.AdrID Kodifikaator.KdID = Aadress.KdIDTase0, KdLyhikeNimi = „Eesti” AadressKomponent.AKpID = Aadress.AkpIDTase3 OR IF Aadress.AkpIDTase3 = 0 THEN Aadress.AkpIDTase2 Kui Aadress.AkpIDTase3 = 0 AND Aadress.AkpIDTase2 = 0, siis väljastada sünnikohatekstina (järgmine väli).	AKpKood	Isikusynnivald

Sünnikoht tekstina	Aadress	<p>IsikuAadress.IsID =  Isik.IsID  Kodifikaator.KdID =  IsikuAadress.AadressiLiik  ,  Kodifikaator.KdLyhikeNimi = „Sünnikoht”  Aadress.AdrID =  IsikuAadress.AdrID  AadressKomponent.AKpID =  Aadress.AkpIDTase3,  Aadress.AkpIDTase3 = 0  AND  AadressKomponent.AKpID =  Aadress.AkpIDTase2,  Aadress.AkpIDTase2 = 0  Väljastada isiku sünnikoha aadress eraldajaks – „tühik“</p>	<p>AdrAadress  Kõik eraldajad „ “ aadressis asendada – „,“. Kui on 2 või rohkem eraldajad korruga (näiteks „   “), siis asendada ainult ühega eraldajaga „,“.</p>	Isikusünnikoht
Isiku põhialukohta valla või asula kood EHAK-i järgi	Aadress Komponent	<p>IsikuAadress.IsID=Isik.IsID  Kodifikaator.KdID=  IsikuAadress.KdAadressiLiik,  Kodifikaator.KdLyhikeNimi=  ”elukoht”  Kodifikaator.KdID=  IsikuAadress.KdAadressiStaatus,  Kodifikaator.KdLyhikeNimi=”kehtiv”  Aadress.AdrID=IsikuAadress.AdrID  Kodifikaator.KdID=  Aadress.KdIDTase0,  KdLyhikeNimi=”Eesti”  AadressiKomponent.AkpID=  Aadress.AkpIDTase3 OR  IF  Aadress.AkpIDTase3 =0  THEN  Aadress.AkpIDTase2  Kui Aadress.AkpIDTase3 = 0 AND  Aadress.AkpIDTase2 = 0,</p>	AKpKood	Isikuvald



		siis väljastada elukoha tekstina (järgmine väli).		
Elukoht tekstina (Eesti puhul: tänav maja-korter)	Aadress	IsikuAadress.IsID=Isik.IsID Kodifikaator.KdID= IsikuAadress.KdAadressiLiik, Kodifikaator.KdLyhikeNimi= "elukoht" Kodifikaator.KdID= IsikuAadress.KdAadressiStaatus, Kodifikaator.KdLyhikeNimi="kehtiv" Aadress.AdrID=IsikuAadress.AdrID AadressKomponent.AKpID = Aadress.AkpIDTase3, Aadress.AkpIDTase3 = 0 AND AadressKomponent.AKpID = Aadress.AkpIDTase2, Aadress.AkpIDTase2 = 0 Väljastada isiku elukoha aadress eraldajaks – „tühik“	AdrAadress Kõik eraldajad „ “ aadressis asendada – „,“ . Kui on 2 või rohkem eraldajad korruga (näiteks „   “), siis asendada ainult ühega eraldajaga „,“ .	Isikuelukoht

Näide:

Isiku perenimi	<b>ÖÖVIHUL</b>
Isiku eesnimi	<b>KATI</b>
Isiku olek	<b>ELUS</b>
Isiku andmete olek	<b>REGISTRIS</b>
Isiku sugu	<b>N</b>
Isiku sünnikuupäev	<b>03.02.1977</b>
Isiku surmakuupäev	
Isiku kodakondsus (kood)	<b>EST</b>
Isiku kodakondsus (nimetus)	<b>EESTI</b>
Isiku sünnikoha valla või asula kood EHAK-i järgi	<b>170</b>
Sünnikoht tekstina	
Isiku põhielukoha valla või asula kood EHAK-i järgi	<b>795</b>
Elukoht tekstina (Eesti puhul: tänav maja-korter)	<b>JÕE 12-12</b>