

TALLINNA TEHNIKAÜLIKOOL  
Infotehnoloogia teaduskond  
Arvutitehnika instituut

IAF40LT

Kaarel Rebase 093964IASB

**TESTIMISE ERINEVAD VÕIMALUSED  
TARKVARA MANUAALTESTIMISE  
NÄITEL**

Bakalaureuse töö

Juhendaja: Margit Aarna

MSc

TTÜ Arvutitehnika  
instituudi assistent

Tallinn 2016

## **Autorideklaratsioon**

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

Autor: Kaarel Rebase

12.05.2016

## **Annotatsioon**

Selles bakalaureuse töös on välja toodud testimise erinevad võimalused tarkvara manuaaltestimise näitel. Keskendatakse kolme erineva kasti meetoditele: musta, valge ja halli kasti. Musta kasti meetodi tehnikad on täpsemalt lahti seletatud. Erinevate tehnikate kohta on toodud nii positiivsed kui ka negatiivsed küljed, mis peaks lugejat informeerima iga tehnika vajadusest ja kasulikkusest. Eraldi on toodud võrdlustabelid erinevate kastide meetodite kohta ja ka automat-, manuaaltestimise kohta. On välja toodud ka erinevad õnnetused või mööda panekud, mis on tingitud tarkvara arenduses tehtud vigadest. Praktilises osas uuritakse Elroni piletimüügisüsteemi. Kirjeldatakse süsteemi funktsionaalsust ja osale sellest funktsionaalsusest on kirjutatud ka testilood. Üks testilugu on automatiseeritud.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 30 leheküljel, 5 peatükki, 4 joonist, 9 tabelit.

## **Abstract**

### **Different possibilities of testing based on an example of manual testing**

This thesis is about different possibilities of testing based on an example of manual testing. The main focus is on three types of testing: black-box, white-box and gray-box testing. Black-box testing techniques are written more thoroughly. For many methods and techniques there are advantages and disadvantages pointed out for the reader to know when to choose any of those methods or techniques. There are also two comparison tables. The first one is comparing black-box, white-box and gray-box methods. The second one is comparing manual and automated testing. There is also a paragraph where is pointed out failures in software development which have led to massive failures in different fields. In the practical part the system under test is Elron ticket sales system. Functionality and features of Elron ticket sales system are mentioned also in the practical part. Test-cases are made for some part of the system and one of them is also automated.

The thesis is in Estonian and contains 30 pages of text, 5 chapters, 4 figures, 9 tables.

## Lühendite ja mõistete sõnastik

IBM	International Business Machines
SEB	Skandinaviska Enskilda Banken
LHV	Lõhmus, Haavel & Viisemann
KK	Kvaliteedi kontroll
intrusivne	Pealetükkiv

## Sisukord

1	Sissejuhatus .....	10
2	Ülevaade .....	12
2.1	Musta kasti tähendus testimisel .....	12
2.1.1	Eelised .....	12
2.1.2	Miinused .....	13
2.2	Valge kasti tähendus testimisel .....	14
2.2.1	Eelised .....	14
2.2.2	Miinused .....	14
2.3	Halli kasti tähendus testimisel .....	15
2.3.1	Eelised .....	15
2.3.2	Miinused .....	15
2.4	Musta, valge ja halli kasti võrdlus .....	16
2.5	Manuaaltestide tähtsus .....	16
2.5.1	Eelised .....	17
2.5.2	Miinused .....	17
2.6	Automaattestide tähtsus .....	17
2.6.1	Eelised .....	18
2.6.2	Miinused .....	19
2.7	Võrdlus manuaaltestid versus automaattestid .....	19
2.8	Näited tarkvara vigadest .....	20
3	Musta kasti meetodi tehnikad .....	21
3.1	Paari kaupa testimine (Ing. K. <i>Pairwise testing</i> ) .....	21
3.1.1	Kasutusala .....	22
3.2	Otsustustabeli tehnika ( <i>Decision table test</i> ) .....	22
3.2.1	Eelised .....	23
3.2.2	Miinused .....	23
3.3	Ekvivalentsiklasside tehnika ( <i>Equivalence testing</i> ) .....	24
3.3.1	Eelised .....	24
3.3.2	Miinused .....	25
3.4	Piirväärtuste analüüs ( <i>Boundary value testing</i> ) .....	25

3.4.1	Eelised .....	25
3.4.2	Miinused .....	26
3.5	Põhjus-tagajärg graafiku tehnika ( <i>Cause-effect graph</i> ) .....	26
3.5.1	Eelised .....	27
3.6	Vigade oletamise tehnika ( <i>Error guessing</i> ) .....	27
3.6.1	Eelised .....	28
3.6.2	Miinused .....	28
4	Praktiline näide .....	29
4.1	Ülevaade uuritavast tarkvarast .....	29
4.1.1	Uuritava tarkvara funktsionaalsus .....	29
4.1.2	Porditavus .....	30
4.2	Testilood .....	30
4.3	Automaattest kirjeldatud testiloole .....	36
4.4	Tulemused .....	37
4.5	Kasutatud tehnikad ja tööriistad .....	38
5	Kokkuvõte .....	39
5.1	Edasised sammud .....	39
	Kasutatud kirjandus .....	40
	Lisa 1 – Visualiseeritud testilood .....	43

## Jooniste loetelu

Joonis 1. Automaattesti eluring. ....	18
Joonis 2. Paari kaupa testimine. ....	22
Joonis 3. Põhjus-tagajärg graafiku tehnika kasutamine.....	26
Joonis 4. Dünaamilise perioodipileti automaatne test .....	36



## Tabelite loetelu

Tabel 1. Musta, valge ja halli kasti meetodite võrdlus .....	16
Tabel 2. Manuaal- ja automaattestimise võrdlus .....	19
Tabel 3. Otsustustabeli näide.....	23
Tabel 4. Ekvivalentsiklasside tehnika näide.....	24
Tabel 5. Testilugu Otsi_reisi. ....	31
Tabel 6. Testilugu Vali_otsitud_reis. ....	32
Tabel 7. Testilugu Ühe_korra_pilet.....	33
Tabel 8. Testilugu I_klassi_pilet. ....	34
Tabel 9. Testilugu Dünaamiline_perioodipilet.....	35

# 1 Sissejuhatus

Testimine on kogum protsesse, mille eesmärk on hinnata, uurida ja kinnitada toote valmidust ja kvaliteeti. Testimine tagab toote vastavuse, sellele seatud äriliste, tehniliste, funktsionaalsete ja kasutajate vajadustega ning sellele kirjutatud spetsifikatsiooniga. Põhilised ülesanded tarkvara testimisel on mõõta toote valmidust võrreldes olulisemate nõuetega, leida üles vead süsteemist ja need raporteerida ning tagada toote mitte funktsionaalsete nõuete täitmine, nagu näiteks turvalisus, kasutatavus, jõudlus. [1]

Tarkvara testimisel on kolm erinevat liiki, mis on kõik seotud ligipääsuga süsteemi lähtekoodile. Nendeks on musta, valge ja halli kasti meetod. Esimene neist on musta kasti meetod, kus testija ei tea, kuidas tarkvara funktsionaalsus sisemiselt töötab ja neil pole ligipääsu lähtekoodile. Selle meetodi üks alamliike on visuaalne testimine, kus kõik leitud vead salvestatakse kas piltidena või siis videodena. See meetod on rohkem levinud agiilses arenduses, kuna seal on oluline testijate, arendajate ja projektijuhtide tihe suhtlus. Teine liik on valge kasti meetod, kus testijal on ligipääs lähtekoodile, teadmise kuidas funktsionaalsus sisemiselt töötab. Selle meetodi abil on lihtsam leida süsteemist kriitilisi vigu ja luua uusi testlugusid, mis on suunatud süsteemi koodist leitud piiride pihta. Viimaseks liigiks on halli kasti meetod, kus testija teab sisemisest arhitektuurist ja tarkvara koodist, kuid tal pole täielikku ligipääsu lähtekoodile. Selle meetodi kõige positiivsem efekt on see, et tal on nii valge kui ka musta kasti meetodi positiivsed küljed olemas. [2] [3]

Tarkvara testimisel on kaks peamist taset: funktsionaalne ja mitte funktsionaalne. Need jagunevad oma korda veel alamtasemeteks. Funktsionaalsel testimisel on mõned põhitõed, mis erinevad vastavalt organisatsioonidest, aga mida võiksid teada kõik. Üks põhilisi punkte, mida peaks teadma, on tarkvara oodatud ja tegelik käitumine. Kui need kaks ühtivad, siis töötab kõik õigesti, vastasel juhul on tarkvaras viga, millega tuleks tegeleda. Üldjuhul otsustab tarkvara tellija, kas leitud probleemidega peaks tegelema või mitte. Funktsionaalse testimise alla kuuluvad üksustestid, integratsioonitestid, süsteemitestid, regressioonitestid ja kliendipoolsed testid. Mitte funktsionaalsed testid oma olemuselt pole seotud tarkvara funktsionaalsusega, aga on siiski olulised. Nende abil

testitakse tarkvara jõudlust, turvalisust, kasutajaliidest ja muud sellist. Mitte funktsionaalsete testide alla kuuluvad jõudlus, koormus-, stressi-, kasutatavuse, turvalisuse ja portimise testid. [4]

Tänapäeval on kolme tüüpi testimise protsessi. Kose mudel, agiilne ja ekstreemne meetod. Kose mudeli puhul kaasatakse testija projekti alles hiljem, siis kui on funktsionaalsus juba valmis ja enne kliendini jõudmist tehakse kogu süsteemile keskenduv testimine. Agiilse meetodi puhul kaasatakse testija võimalikult varakult projekti, see tähendab, et testija poolt leitud puudujääkidega tegeletakse koheselt ja seetõttu on agiilne arendusmeetodika väga paindlik muutustele, vastupidiselt Kose mudelile, kus paindlikkus muutustele on pea olematu. Ekstreemse arendus meetodika puhul, täpsemalt siis testidest juhitud arenduse puhul, kirjutab arendaja kõige pealt uue funktsionaalsuse jaoks läbikukkuva testi ja seejärel hakkab ta looma funktsionaalsust, mille kohta tal juba test on olemas. Kusjuures tehes seda kõike võimalikult vähest koodi kirjutades. [5]

Testimisel on väga oluline roll tarkvara arenduses. Siin on mõned põhilised põhjused, miks on testimisel väga oluline roll mõne teenuse või toote arenduses. Testijad viitavad toote testimise käigus leitud defektidele ja vigadele. Heade testijate olemasolu tagab toote kõrge kvaliteedi. Testimisel ja tarkvara hooldusel on väga otsene seos tarkvara arendusega kaasnevate kuludega. Testija olemasolu viib need kulud pikas perspektiivis alla, kuna varakult avastatud vigade parandamine on kordades odavam kui hiljem leitud vigade korrigeerimine. [6]

## 2 Ülevaade

Tarkvara testimine on protsess, kus käivitatakse tarkvara või süsteem eesmärgiga leida sellest vigu. Tarkvara tootes on peaaegu alati vead, põhjus pole selles, et tarkvara arendajad oleksid laisad või hoolimatud, vaid selles, et süsteemile keerukust lisades tõuseb ka keerulisemate vigade arv. Boris Beizer on kirjutanud oma raamatus järgmist: „Iga meetod, mida sa kasutad, et hoiduda vigadest või leida neid, jätab alles jäägi vigadest, mis on sinu kasutatud meetodi suhtes varjatud.“ [7] „Tarkvara keerukus (ja sellepärast ka vigade) tõuseb meie võimekuse piirini hallata seda keerukust.“ [7] Kasutajad suruvad tarkvara loojaid üha keerukamate süsteemide poole ja see omakorda loob olukorra, kus kõiki vigu on pea võimatu leida ning parandada. [8]

Järgnevalt teen ülevaate testimise erinevatest meetoditest lähemalt.

### 2.1 Musta kasti tähendus testimisel

Musta kasti testimine, vahel kutsutakse ka funktsionaalseks või käitumuslikuks testimiseks, teeb kindlaks, kas tarkvara käitub nii nagu on temalt oodatud. Oluline on märkida, et on parem kui testimist ei vii läbi tarkvara arendaja, vaid keegi teine, kuna arendaja testib ainult seda, mida ta on loonud, aga mitte seda, kas tarkvara töötab nii nagu klient seda soovib. Selle meetodi puhul pole testijal vaja teada tarkvara struktuuri, koodi või valitud programmeerimiskeelt. [9] Suurim osa uurivast testimisest tehakse kasutades musta kasti meetodit. Uuriv testimine on üheaegselt testimise õppimine, planeerimine ja sooritamine. Ühele süsteemiosale keskendudes genereeritakse testimise käigus uusi võimalusi, kuidas sellele kohale tarkvaras läheneda. [10]

#### 2.1.1 Eelised

Musta kasti meetodil on mitmeid positiivseid külgi võrreldes kahe teise meetodiga. Siin on loetelu neist:

- Tõhus testimine – sobib hästi suurte projektide jaoks, kus on suur funktsionaalsus.
- Erapooletu testimine – kindlalt on eraldatud kasutaja ja arendaja perspektiiv, kuna KK'l ja arendusmeeskonnal on erinevad kohustused.

- Mitte intrusiivne – koodile ligipääs pole vajalik.
- Lihtne teostada – testilood saab kirjutada valmis kohe peale spetsifikatsiooni valmimist. Testimise läbiviijateks võivad olla testijad, kes ei tea midagi süsteemi sisemisest poolest.
- Programmeerimise oskus pole vajalik – testijad ei pea teadma programmeerimiskeeli ega seda kuidas on tarkvara implementeeritud.
- Aitab leida järjepidamatust spetsifikatsioonis. [12] [13]

### 2.1.2 Miinused

Musta kasti meetodil on ka mitmeid puudujääke võrreldes kahe teise meetodiga. Siin on loetelu neist:

- Lokaliseeritud testimine – Koodi kaetuvus on limiteeritud, kuna kõiki sisendväärtusi ei testita.
- Ebaefektiivne testide loomine – implementatsiooni informatsioonita annab põhjalik sisendväärtuste kaetuvus teadmatuid lisandväärtusi koodi kaetuvusele. Võib nõuda väga suurti ressursse.
- Pime kaetuvus – kuna koodile puudub ligipääs, siis kõiki piire ja koodi teid ei saa testida. Võib jääda vääär arusaam, et kõik on töökorras.
- Koondatud – testid võivad olla koondatud, kui disainer või arendaja on juba testilood läbi käinud.
- Testilugusid on raske formuleerida. [12] [3]

## 2.2 Valge kasti tähendus testimisel

Valge kasti testimisel, vahel ka klaas kasti testimine, on testijal ligipääs kõigele, mis on seotud loodava tarkvaraga: lähtekood, süsteemi struktuur, funktsionaalsuse loogika. Testiplaan on loodud vastavalt tarkvara implementatsioonile, nagu valitud programmeerimiskeel, loogika ja stiil. Testilood on juhitud süsteemi struktuurist. [8] Seda meetodit kasutatakse süsteemi sisemiseks testimiseks ehk süsteemi koodi testimiseks. Testija kontrollib, kas koodis näidatud teed viivad sinna nagu arhitektuur ette näeb ja kas kõik teed on valiidsed. Selleks kirjutab testija automaattestid, millel on kaks olekut, kas õige või vale. Kui olek on vale, siis on kaks võimalust, kas viit ei vii õigesse kohta või test on kirjutatud ebakorrektselt. Nähes süsteemi koodi, saab testija ka aimu tarkvara piiridest ja seega saab siis paremini testida olukordi, kus minnakse välja piiridest. [8]

### 2.2.1 Eelised

Valge kasti meetod on väga erinev musta kasti meetodist, kuid sarnaneb halli kasti meetodile. Loetelu positiivsetest külgedest:

- Tõstetud efektiivsus – kontrollides disaini otsuseid ja eeldusi lähtekoodis, võib välja tuua robustse disaini, aga nende otsuste implementatsioon ei pruugi olla kooskõlas plaanituga.
- Täielik koodi teede kaetuvus – kõiki võimalikke koodi teid saab testida.
- Varajane defektide leidmine – analüüsides lähtekoodi ja luues teste arvestatakse implementatsiooni detailidega, mis loob soodumuse, et avastatakse programmeerimise vead kiiresti.
- Avastatakse peidetud koodi vead – ligipääs lähtekoodile parandab arusaama ja paljastab tahtmatud varjatud tarkvara käitumised.
- Ei pea ootama – testimist saab alustada veel varasemalt. Ei pea ootama, et kasutajaliides oleks valmis.
- Aitab optimeerida koodi. [12] [3]

### 2.2.2 Miinused

Siin on valge kasti meetodi puudujäägid võrreldes kahe teise kasti meetodiga:

- Raske hoomata – testijal peavad olema väga head baasteadmised tarkvarast, testimise tööriistadest ja programmeerimiskeeltest.
- Raske hoida töökorras – on vaja spetsiifilisi tööriistu nagu koodi analüüsija ja siluja.

- Kultuurne stress – piir arendajate ja testijate vahel võib hägustuda, mis võib põhjustada stressi ühel või teisel pool.
- Tööriistade kättesaamatus – kuna selle meetodi puhul on testimine väga tihedalt seotud loodava tarkvaraga, siis tööriistad, mis käsitlevad igat erinevat implementatsiooni või platformi, ei pruugi olla kättesaadavad.
- Väga intrusiivne – tugineb puhtalt lähtekoodil. Seetõttu iga muudatus lähtekoodis võib põhjustada varjatud vigu.
- Kuna testijad peavad olema tasemel, siis on ka testimise maksumus suurem. [12] [3]

## 2.3 Halli kasti tähendus testimisel

Halli kasti testimine on meetod, mis on kombinatsioon musta ja valge kasti meetodist. Testijale on teada *midagi* tarkvara struktuurist, lähtekoodist ja sisemisest loogikast, aga testimine toimub siiski kasutaja tasemel. [11] See meetod sobib hästi veebirakenduste testimiseks. Kuna seda meetodit kasutades muudetakse ka rakenduse koodi kasutajaliidese tasemel, siis kuulub see valge kasti meetodisse ja testides muudatuste tulemusi kasutaja vaates, siis see kuulub musta kasti meetodisse. [12]

### 2.3.1 Eelised

Halli kasti meetodil on omad positiivsed küljed, kuna see on osaliselt nii valge kui ka musta kasti meetod. Siin on loetelu neist:

- Pakub kombineeritud kasu – on nii valge kui ka musta kasti meetodi head küljed.
- Mitte intrusiivne – ei tugine lähtekoodil. Tugineb kasutajaliidese definitsioonidel, spetsifikatsioonil ja tarkvara arhitektuuril.
- Intelligentne testide loomine – kuna ligipääs on piiratud, siis testija saab luua intelligentseid testilugusid, eriti mis on seotud andmetüüpide käsitlemisega, kommunikatsiooni protokollidega ja erandite käsitlemisega.
- Erapooletu testimine – piir testijate ja arendajate vahel on olemas.
- Testimine on tehtud kasutaja vaatevinklist mitte arendaja. [12] [3]

### 2.3.2 Miinused

Halli kasti meetodil on ka negatiivseid külgi ja need on toodud järgnevas loetelus:

- Osaline koodi kaetavus – kuna ligipääs lähtekoodile pole täielik, siis testidega koodi kaetavus sõltub otseselt testija oskustest.

- Defektide identifitseerimine – kõik sõltub kui hästi viskab tarkvara erandeid, et nende abil jõuda vea põhjusteni.
- Koondatavus – kui tarkvaraarendaja on juba testilood läbi käinud. [12] [3]

## 2.4 Musta, valge ja halli kasti võrdlus

Kuna igal meetodil on omad positiivsed ja negatiivsed küljed, siis siin on välja toodud erinevuste tabel:

Tabel 1. Musta, valge ja halli kasti meetodite võrdlus

	Musta kasti meetod	Halli kasti meetod	Valge kasti meetod
Sisemine loogika ja arhitektuur ning lähtekood	Pole teada	Osaliselt teada	Täielikult teada
Teostavad	Lõppkasutaja, testija ja arendaja	Lõppkasutaja, testija ja arendaja	Üldjuhul testija ja arendaja
Millele tuginedes testitakse?	Välistele ootustele	Kõrgtasemel andmebaasi diagrammid ja andmete diagrammid	Kõik tarkvaraga seotu on testijale teada.
Ajakulu ja põhjalikkus	Põhjalik ja kõige vähem ajakulukas.	Osaliselt ajakulukas ja põhjalik.	Kõige põhjalikum ja ajakulukam.
Algoritmide testimine	Pole sobilik	Pole sobilik	Sobilik
Andme domeenide ja sisemiste piiride testimine?	Ainult katse-eksitus meetodiga.	Teadaolevaid saab testida.	Kõige põhjalikumalt saab testida.

[12] [3]

## 2.5 Manuaaltestide tähtsus

Testijal on soovituslik kasutada manuaalseks testimiseks musta ja valge kasti tehnikaid. Manuaaltestid aitavad avastada ja raporteerida vigu või mitte oodatud käitumist tarkvara funktsionaalsuses. [14] On võimalik kiiresti hinnata toote seisukorda ja ja tutvuda tarkvara funktsioonidega juba arendus protsessi alguses. Testilood luuakse vastavalt testija võimele hinnata, kas nõuded on täidetud või mitte. Meeles hoitakse ka teadmist, et tulevikus peab võib olla mõne testiloo automatiseerima, seega kõik testilood on samm-sammult kirja pandud. [15]



Manuaalse testimise alla kuulub ka uuriv testimine, kus testija õpib testimise käigus tarkvara tundma ja märgib üles mõned ebaselged kohad, mis vajavad rohkem tähelepanu ja testimist, et mõista potentsiaalseid riske ja nõrkusi.[15] Käsitsi testimist kasutatakse enamasti kasutatavuse ja kasutajaliidese testimiseks, kuna seda pole mõistlik teha automaatselt.[14] Üks suuremaid eeliseid manuaaltestimisel on see, et testija on kui lõppkasutaja ja tema poolt leitud vead on otseselt seotud potentsiaalsete kasutajate muredega. Käsitsi testimine on ka väga pädlik, muutuvad väärtused ja uitmõtetest tingitud testide loomine on lihtne.[16]

### **2.5.1 Eelised**

Loetelus on toodud manuaaltestimise eelised:

- Lühiajaline maksumus on madalam.
- Uuriv testimine on produktiivsem manuaalselt testides.
- Vead, mis leitakse, on otseselt seotud lõppkasutajaga.
- On pädlik muutustele.
- Kasutajaliidest saab adekvaatselt hinnata ainult inimene, mitte programm. [21] [22]

### **2.5.2 Miinused**

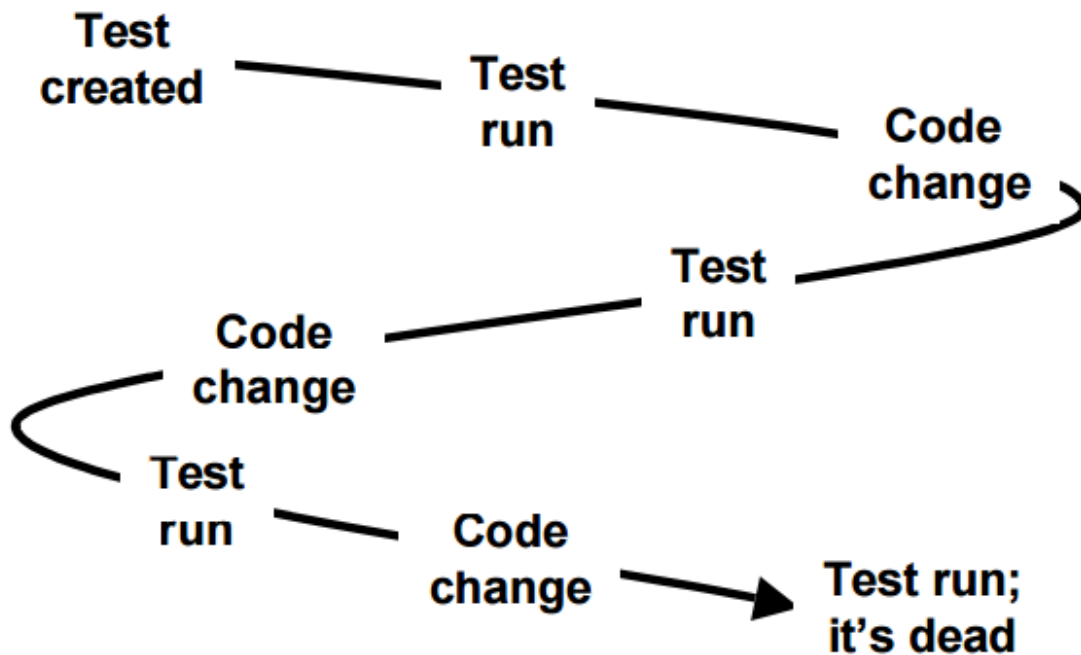
Loetelus on toodud manuaaltestimise puudujäägid:

- Teatud osi tarkvarast on raske testida – koormustestid, regressioonitestid.
- Testijal võib kaduda stiimul – sama tegevuse tegemine korduvalt.
- Teste ei saa korrata 100% kindlusega, et kõik sammud on täpselt samad igal korral.
- Koodi kaetuvust ei saa hinnata.
- Testija võib eksida – testi tulemused võivad olla eksitavad. [21] [22]

## **2.6 Automaattestide tähtsus**

Automaattestide olemasolu muudab tarkvara testimise efektiivsemaks ja kiiremaks.[16] Kui on automaattest loodud, siis seda saab käima panna millal iganes. Valideerides mõni parandus või uuendus tarkvaras ja luues sellele automaatne test, muudab järgnevate

muudatuste tegemise valutumaks. Automaattestid on sobilikud regressiooni testimiseks, kus testitakse, et juba arendatud funktsioonides ei esine vigu arenduse jätkudes. [15]



Joonis 1. Automaattesti eluring.

Joonisel on toodud automaattesti eluring. Üldjuhul muutub arenduse käigus mõni test ebavajalikuks, kuna funktsionaalsus, millele test kirjutati, muutub niivõrd palju. [17]

Koormustestide loomine automaattestide näol on kindlasti odavam ja efektiivsem, kui näiteks korraldada 300 testija üheaegne tarkvara kasutamine. Testijate tasustamine üksi on juba väga kulukas ja saadud koormus ei pruugi olla samaväärne automaatselt saavutatuga.[17]

### 2.6.1 Eelised

Loetelu automaattestide eelistest:

- Loodud teste saab käima panna kiiresti ja efektiivselt.
- Pikas perspektiivis on odavam.
- Testijale on töö huvitavam, kuna ta saab korduvad testid automatiseerida.
- Kõik arendus meeskonna liikmed saavad näha testide tulemusi.
- Koodi kaetuvust saab hinnata.

- Testi võib käima panna ka mitte tehnilise taustaga inimene. [21] [22]

## 2.6.2 Miinused

Loetelu automaatsete puudujääkidest:

- Testide loomiseks vajaminevad tööriistad ja testide hooldamine võivad olla kallid.
- Testide loomiseks vajaminevad oskused on vaja eelnevalt omandada.
- Testide loomine võtab kaua aega.
- Kasutajaliidest ja visuaalseid vigasid on raske testida ja leida.
- Automaatselt ei ole mõistlik teha uurivat testimist.[21] [22]

## 2.7 Võrdlus manuaaltestid versus automaatsetid

Järgnevalt on toodud erinevused ning võrdlused manuaal- ja automaatsete vahel.

Tabel 2. Manuaal- ja automaatsetimise võrdlus.

	Manuaalsetimine	Automaatsetimine
Korratavus	Pole piisavalt usaldusväärne.	Sama testi korratakse täpselt sama moodi igal korral.
Regressioonitestid	Pole piisavalt usaldusväärne	Koodi muudatustest tingitud vigade leidmine on tunduvalt tõenäolisem
Kasulikkus?	Kui sama testi ei pea tegema mitu korda	Kui sama testi peab tegema mitmeid kordi
Kasutajaliidese testimine	Väga sobilik	Sobilik, aga mitte piisavalt usaldusväärne
Ajakulukus	Ühe teema testimiseks läheb iga kord sama kaua aega	Peale testi valmimist, läheb testimiseks alla 10 sekundi
Testijate hulk	Mida suurem projekt seda rohkem on vaja	Automaatsetide olemasolu vajab väksem arv testijaid
Rahakulukus	Algselt on odavam	Algne investeering on kallis, aga pikas perspektiivis on odavam
Koodi kaetuvus	Ebamäärane	Saab määratleda palju on kaetud

[18] [19] [20]

## 2.8 Näited tarkvara vigadest

Alljärgnevalt loetelu juhtumitest, kus tarkvara arenduses märkamata jäänud vead on põhjustanud erinevaid tagajärgesid:

- Neljandal juunil aastal 1996 ebaõnnestus Euroopa Kosmose Agentuuri poolt korraldatud raketi start. Rakett plahvatas 40 sekundit peale starti. Uurimuse tulemusena avastati, et põhjuseks oli 64 bitise arvu konverteerimine 16 bitiseks, aga 16 bitise arvu suurim number on 32,767. Konverteeritud arv oli suurem sellest ja tulemuseks oli plahvatus.[23]
- Mais 1996 said tarkvara vea tõttu 823 ühe Ameerika suurima panga klienti oma panga kontole 924 844 208.32 dollarit. Ameerika Pankurite Ühendus väitis, et see oli suurim selline viga panganduse ajaloos. Panga esimees ütles avalikult, et programmeerimis vead said parandatud ja raha tagastatud. [24]
- Mais 2005 raporteeris üks ajalehe väljaanne, et üks suurim hübriidautode tootja pidi ligi 20 000 autole installeerima tarkvara paranduse, kuna neil ei töötanud mootoririkke tuli ja vahel jäid autod ootamatult seisma. Selle teemaga seotud artiklis väitis üks autode arendusega seotud spetsialist, et autode tootjad kulutavad aastas kaks kuni kolm miljardit dollarit tarkvara vigade parandamise peale. [24]
- Mängude loomisele keskenduv ettevõtte andis 2010 aastal välja nii vigase mängu, et ettevõtte tegevjuht saatis isiklikult klientidele välja kirja, kus ta vabandas mängu kehva kvaliteedi eest. [24]
- Aprillis 2014 oli Ameerika Ühendriikide seitsmes osariigis probleemid hädaabi numbrile 911 helistamisega. See oli tingitud tarkvara veast ja liin ei töötanud 6 tundi, mille tõttu jäi üle 6000 hädaabi kõne vastamata. [24]

### 3 Musta kasti meetodi tehnikad

Ideaalses maailmas testiksime kõike võimalikku, mida saab teha loodava tarkvaraga, aga testide loomine ja läbi viimine on kallis. Seega eesmärgiks on luua testilood selle kohta, mida kasutaja teeb tõenäoliselt kõige rohkem ja luues selleks võimalikult vähe testilugusid. Testilood peavad olema võimalikult täpsed ja lihtsad, kuna ka testilugudes endas võib olla vigu sees, mis võib anda eksitavat informatsiooni tarkvara olukorra kohta. [9]

#### 3.1 Paari kaupa testimine (Ing. K. *Pairwise testing*)

Paari kaupa testimise tehnika vähendab testilugude arvu, kuna testitakse kõiki paare, mitte aga ei püüta testida kõiki kombinatsioone kõikide muutujatega. [26] See testimise tehnika on kohaldatav tarkvara või riistvara testimiseks. Sellel on oma põhjalikkuse tasemed: 1 – 6. Iga tase lisab juurde ühe muutuva väärtuse. [25]

Jooniselt 2 on näha, et kui on näiteks 30 paari kaupa testimise testilugu, siis neist 12 testimine hõlmab endas juba rohkem kui 95% paaridest. Ülejäänud 18 testiloo läbimine hõlmab endas kõigest 5%. Seega osaline testilugude läbimine, meie näite puhul esimese 12, võib leida 95% tarkvara vigadest, mida täielik testilugude läbikäimine leiaks. [25]

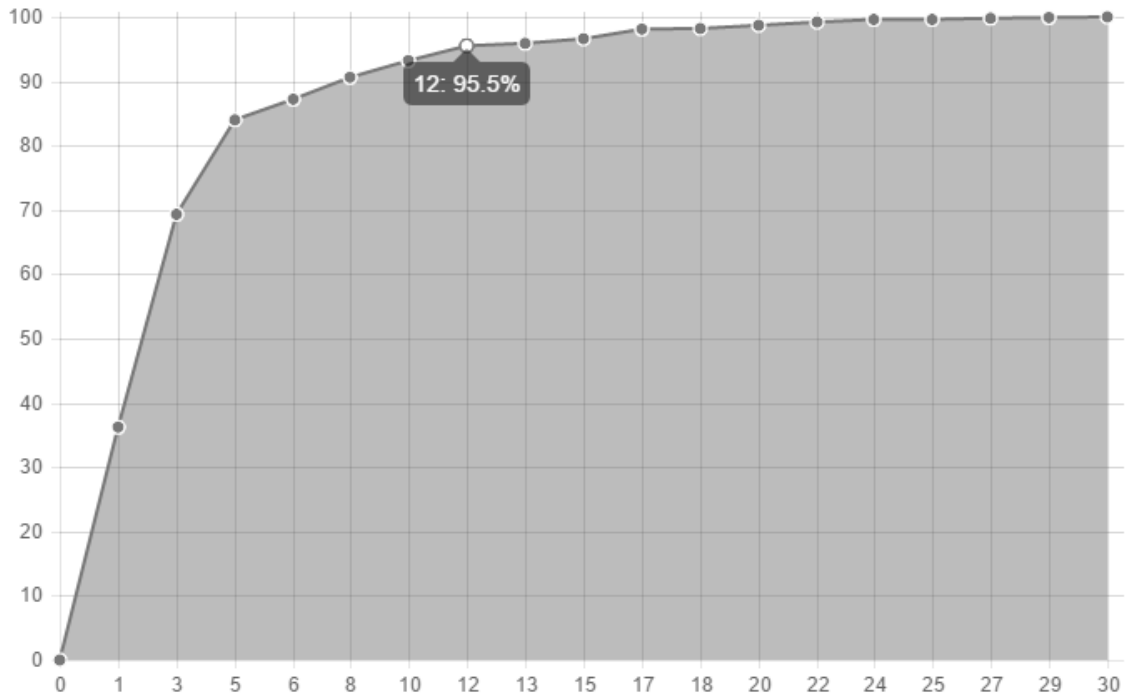
## 2-wise Coverage:

Tests: 30

Coverage: 100%

T-sets covered / total (uncovered): 7564 / 7564 (0)

## Accumulated Coverage:



Joonis 2. Paari kaupa testimine.

### 3.1.1 Kasutusosalad

Siin on loetelu aladest, kus on mõistlik kasutada paari kaupa testimist:

- Tarkvara või riistvara liinitööd.
- Tugevalt muudetavad süsteemid.
- Keerulised ja varieeruvad keskkonnad.
- Kasutajaliidese arendus protsessis.
- Sisemiste arendusloogikate testimiseks – funktsioonid, meetodid, klassid ja moodulid.[25]

### 3.2 Otsustustabeli tehnika (*Decision table test*)

Otsustustabeli tehnikaga keskendutakse süsteemi tingimuste testimisele.[10] See tehnika annab võimaluse jälgida keerulist otsuste loogikat. All pool on toodud näide otsustustabeli kasutamisest.

Tabel 3. Otsustustabeli näide.

	Reegel 1	Reegel 2	Reegel 3
Raha on kaardil	J	E	E
Sularaha on taskus	-	J	E
Raha pole	-	-	J
Saab minna kinno	X	X	
Peab raha minema teenima			X

„Raha on kaardil“, „Sularaha on taskus“ ja „Raha pole“ on tingimused, mida uuritakse ja vastused on „J“(jah), „E“(Ei) või pole oluline. „Saab minna kinno“ ja „Peab raha minema teenima“ on tegevused, mida saab teha vastavalt otsustele. „X“ tähistab tegevust, mida saab teha.

### 3.2.1 Eelised

Loetelu otsustustabeli kasutamise eelistest.

- Otsuste reeglid on selgelt kirjeldatud.
- Otsuste langetamine on järjepidev.
- Projekti juhtidel ja analüütikutel on lihtsam suhelda oma vahel.
- Dokumentatsiooni on kerge muuta.
- Lihtne kasutada ja mugav selgitada mõndasid otsuseid.
- Lihtsam on luua suuremahulist dokumentatsiooni. [27]

### 3.2.2 Miinused

Loetelu otsustustabeli kasutamise puudujääkidest:

- Lisaülesanded lisanduvad testimisele.
- Ei ole näha töövoogu.
- Pole lihtne tõlkida projektist kaugele inimesele.
- Kõiki alternatiive ei uurita. [27]

### 3.3 Ekvivalentsiklasside tehnika (*Equivalence testing*)

Ekvivalentsiklasside testimise tehnika põhiidee on selles, et testija jagab tarkvara samaväärseteks osadeks ning kasutab testimisel iga osa ühte väärtust. See tehnika rajaneb eeldusel, et kui üks väärtus testivast osast on õige, siis on õiged ka kõik ülejäänud selle osa väärtused.[28]

All pool on toodud näide ekvivalentsiklasside tehnika kasutamisest:

#### Näide

Riik loob uue hoiustamise poliitika ning määrab uued kehtivad intressid hoiustele. Summale 0 eurot kuni 500 eurot on intressiks 1%, summale 500.01 eurot kuni 2000 eurot on 2% ja summale alates 2000.01 eurot on intressiks 3%.

Ekvivalentsiklasside tehnikat arvestades moodustub kolm kehtivat ekvivalentsiklassi ja üks kehtetu. Oluline tähelepanek seisneb selles, et kehtetu ekvivalentsi klass võib olla kehtetu ainult intressi arvutamisel.

Tabel 4. Ekvivalentsiklasside tehnika näide.

Kehtetu	Kehtiv	Kehtiv	Kehtiv
-0,01	0,00 - 500	500,01 - 2000	2000,01 -

Intressi arvutamise testilugude loomisel võetakse igast klassist üks suurus ja kontrollitakse selle õigsust. Selle tulemust arvestades otsustatakse, kas ülejäänud klassi väärtused on õiged.

#### 3.3.1 Eelised

Loetelus on toodud ekvivalentsiklasside tehnika positiivsed küljed:

- Testilugude hulk väheneb.
- Klassidesse jagamine aitab testijal fookuseerida väiksematele väärtuste hulgale, mistõttu vigade leidmise tõenäosus on suurem.
- Kõrvaldab põhjaliku testimise kogu sisendite domeenis - mis ei olegi teostatav.
- Võimaldab katta suuri domeene sisenditega. [29]



### 3.3.2 Miinused

Loetelu ekvivalentsiklasside tehnika puudujääkidest:

- Tavaliselt ei arvestata äärejuhtudega.
- Pole selgeid juhiseid, milliseid konkreetseid väärtusi testida.
- Vahel ignoreeritakse probleeme äärejuhtudega.
- Testijale võib jääda väär arusaam, et kõik sisendväärtuste tulemused on korrektsed. [29] [30]

### 3.4 Piirväärtuste analüüs (*Boundary value testing*)

Tuntud, mitmete testimisalaste raamatute autor, Boris Beizer on kirjutanud oma raamatus: „Vead varitsevad nurkades ja kogunevad piirialadel.“[7] Kuna äärealad on kõige kriitilisemad ja programmeerijad teevad tihti just seal vigu, siis peaksidki testid olema suunatud just piiride pihta. Piirväärtuste analüüs programmeerimise tähenduses on järgmine: testandmete valimise tehnika, kus väärtused valitakse antud piiride ääres. Piirväärtused koosnevad maksimaalsetest ja minimaalsetest väärtustes, mis on täpselt loodud piirides või neist väljas. Eeldatakse, et kui nende väärtuste sisestamisel või testimisel töötab tarkvara korrektselt, siis on ka muud väärtused, mis on piiride vahel, korrektsed. [9]

#### Näide

Ühte klassi mahub 50 eurot. Piirväärtuste analüüsi põhjal peaksime looma kolm testilugu, kust testitavateks väärtusteks on 49 eurot, 50 eurot ja 51 eurot. Kui need testid on läbitud ja vigu ei esine, siis võib eeldada, et ülejäänud väärtused vahemikus 0 kuni 50 ei põhjusta vigu tarkvaras.

#### 3.4.1 Eelised

Loetelus on toodud piirväärtuste analüüsi positiivsed küljed:

- Hea tööriist leidmaks vigu kasutajaliideses või sisestamisväljadel.
- Väga selgesti arusaadav missuguseid testilugusid koostada.
- Väike hulk testilugusid. [30]

### 3.4.2 Miinused

Loetelu piirväärtuste analüüsi puudujääkidest:

- Ei testita kõiki sisendväärtusi.
- Ei testita sisendväärtuste kombinatsioonide omavahelisi sõltuvusi. [30]

### 3.5 Põhjuse-tagajärg graafiku tehnika (*Cause-effect graph*)

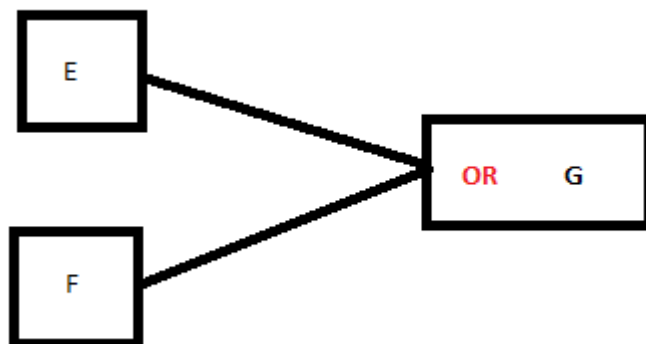
Põhjuse-tagajärg graafiku tehnika leiutas 1973. aastal Bill Elmendorf IBM'ist. Põhiidee on selles, et testija proovib moduleerida tarkvara funktsionaalsuse kasutades põhjuse-tagajärg graafikut ja ta kasutab selleks tarkvara, mis oskab seda tehnikat kasutada. Selle abil arvutatakse välja õige arv testilugusid, mis katavad kogu funktsionaalsuse. [31]

#### Näide

Lihtne näide. Spetsifikatsioonist tuleb välja: „Kui E või F, siis G.“

Sellega kaasnevad järgnevad reeglid:

- Kui E on õige ja F on õige, siis G on õige
- Kui E on õige ja F on vale, siis G on õige
- Kui E on vale ja F on õige, siis G on õige
- Kui E on vale ja F on vale, siis G on vale



Joonis 3. Põhjuse-tagajärg graafiku tehnika kasutamine.

Kõik sõltuvused on kujutatud oma vahel joontega ja selleks on olemas mõned reeglid, mis on järgmised:

- Kui E viib alati G, siis on nende vahel must joon.
- Kui E või F viib G, siis on nende vahel „OR“ suhtlus.
- Kui E ja F viib G, siis on nende vahel „AND“ suhtlus.
- Kui E ei vii G, siis on nende vahel punane joon.

### 3.5.1 Eelised

Siin on loetelu põhjus-tagajärg graafikute kasutamise positiivsetest külgedest:

- Aitab välja selgitada probleemide algsed põhjused, kasutades selleks struktureeritud lähenemist
- Aitab leida peidetud vigu, mis ei ole kohe märgatavad, kuna kõik nagu toimib, aga taustal on siiski miskit valesti
- Selgesti arusaadav, mis on graafikutega püütud selgitada
- Viitab võimalikele varieeruvustele arendus protsessis
- Identifitseerib kohad, mille kohta oleks tarvis koguda rohkem andmeid hilisemaks kasutamiseks
- Aitab kaasa tiimi arengule ja teadmiste protsessist
- Kasvatab tiimi teadmisi tööprotsessist ja aitab kõigil aru saada, missugused elemendid on omavahel seotud ja kuidas [32] [31]

## 3.6 Vigade oletamise tehnika (*Error guessing*)

Vigade oletamise tehnikat kasutatakse alles siis, kui tarkvara on valmimise järgus ja ülejäänud tehnikatega on juba enamus vead üles leitud. See tehnika põhineb otseselt testija kogemustel ja teadmistel, kuna testitakse olukordi, millele üldjuhul tähelepanu pööratakse vähe. See tehnika on rohkem testijale kindlustunde tekitamiseks, kus võib näitena tuua, et kui süsteem käitub nendes tingimustes korrektselt, siis käitub ta õigesti ka ülejäänud tingimustes, teisisõnu tarkvara funktsionaalsus on töökorras. [10]

## Näide

Esimene reegel: Kasutaja parool peab olema kaheksa kuni 16 märki pikk ja sisaldama endas vähemalt üht suurt tähte ja üht numbrit.

Tarkvaras on väli nimega Kasutaja parool.

Vigade oletamise tehnikat kasutades tekiksid järgmised küsimused:

- Mis juhtub kui sinna ei kirjutata midagi?
- Mis juhtub kui kõik märgid pole numbrid ega tähed?
- Mis juhtub kui kirjutada alla kaheksa või üle 16 tähemärgi?
- Mis juhtub kui sisestada sinna mõni link?
- .....

### **3.6.1 Eelised**

Siin on loetelu vigade oletamise tehnika kasutamise positiivsetest külgedest:

- Väga edukad testijad leiavad kiirelt vigu tarkvarast ja suudavad hinnata selle valmidust
- Sobib hästi piirväärtuste analüüsi ja ekvivalentsiklasside tehnika puudujääkide kompenseerimiseks
- Selle tehnika omandamine tuleb testijale kasuks, kuna selle tõuseb tema efektiivsus ja kasulikkus [33]

### **3.6.2 Miinused**

Siin on loetelu vigade oletamise tehnika puudujääkidest:

- On sobilik ainult edasijõudnud testijale, algaja pole piisavalt efektiivne seda tehnikat kasutades
- Baseerub testija intuitsioonil [33]

## 4 Praktiline näide

Selles peatükis toon välja testimise erinevad võimalused tarkvara manuaaltestimise näitel ja loon ühele manuaalsele testile automaatse testi, mis on iseenesest näide, kuidas saab käsitsi tehtavad testid automatiseerida. Esiteks teen ülevaate tarkvarast, mida kasutan testimiseks, siis kirjutan täpsustavad testilood ning lõpuks toon näite ühest funktsionaalsest testist, mille olen automatiseerinud. Hiljem on ülevaade tulemustest ning ka ära kirjeldatud kasutatud tehnikad ning tööriistad.

### 4.1 Ülevaade uuritavast tarkvarast

Uuritavaks süsteemiks on Elroni piletimüügisüsteem, mis on nii mobiiltelefonist kui ka veebibrauserist kättesaadav aadressilt [pilet.elron.ee](http://pilet.elron.ee). Järgnev on see, mida Elron enda kohta kirjutab: „*Eesti Liinirongid AS (ELRON) on ühistranspordiettevõtte, mis korraldab Eesti reisirongiliiklust.*

*Elron pakub Eesti tingimustele vastavat innovatiivset, kiiret, mugavat, turvalist ja keskkonnasõbralikku ühistransporditeenust, mis on hea valik maanteetranspordi (auto või bussi) asemel. Soovime olla Eesti mainekaim ühistranspordiettevõtte, mis oleks oma hea teenusega eeskujuks teistele ühistranspordiettevõtetele kogu Läänemere regioonis.“[34]*

#### 4.1.1 Uuritava tarkvara funktsionaalsus

Siin on loetelu uuritava tarkvara funktsionaalsusest:

- Kasutajal on võimalik osta endale rongipileteid, kusjuures need võivad olla nii liinipõhised- kui ka ümberistumisega piletid.
- Kasutajal on võimalik vaadata valitud reiside sõiduplaane ja talle on ka märgitud ära millised reisid on soovituslikud jalgrattaga reisimiseks.
- Kasutajal on võimalik logida sisse Elroni sõidukaardiga, mida tehes näeb ta enda kehtivaid pileteid ja tal on võimalik näha ka oma ostude ajalugu: viimase 30 päeva tehingud või kõik tehingud. Konto väljavõtte saab ka välja printida.
- Kui kasutaja on sisse loginud sõidukaardiga, siis on tal võimalik kanda sõidukaardile raha, mida ta saab kasutada ainult Elroni rongides.
- Kui kasutaja on sisse loginud Elroni sõidukaardiga, siis saab ta pileti ostmisel maksesoodustust 10% igalt ostetud piletilt.

- Kasutajal on võimalik osta endale ka Esimese klassi pileteid, kusjuures kui osta ette teatud arv päevasid pilet, siis on kuvatud ka piiratud liikumisvõimega isikute istekohad.
- Kasutajal on võimalik osta endale dünaamiline perioodipilet, mille kehtivust saab ta ise muuta ja Ostukorvis saab ta muuta ka kehtivuse algus ja lõpp kuupäevaid.
- Kasutajal on võimalik pileti ostmisel valida, kas pilet salvestatakse Elroni sõidukaardile või e-mailile, mille saab ta ise sisestada.
- Kasutajal on võimalik maksta piletite eest kuut eri moodi: Swedbanki, SEB, Danske Banki, Nordea, LHV panki või Krediidipanki pangakaardiga; Mastercard või Visa krediitkaardiga.

#### **4.1.2 Porditavus**

Siin on loetelu veebibrauseritest ja seadmetest, millega saab Elroni piletimüügisüsteemi kasutada:

- Google Chrome
- Internet Explorer
- Mozilla Firefox
- Microsoft Edge
- Opera
- Androidi süsteemiga seadmed (mobiiltelefonid, tahvelarvutid)
- iOS süsteemiga seadmed (mobiiltelefonid, tahvelarvutid, lauarvutid)
- Windows Phone 10
- Windows Phone 8
- Windowsi operatsioonisüsteemid
- Linuxi operatsioonisüsteemid

## **4.2 Testilood**

All järgnevad testilood katavad ainult osa funktsionaalsusest, aga on üles ehitatud nõnda, et viimane testilugu automatiseeritakse. Kaetud funktsionaalsus on:

- Kasutajal on võimalik osta endale rongipileteid, kusjuures need võivad olla nii liinipõhised- kui ka ümberistumisega piletid.

- Kasutajal on võimalik vaadata valitud reiside sõiduplaane ja talle on ka märgitud ära millised reisid on soovituslikud jalgrattaga reisimiseks.
- Kasutajal on võimalik osta endale ka Esimese klassi pileteid, kusjuures kui osta ette teatud arv päevasid pilet, siis on kuvatud ka piiratud liikumisvõimega isikute istekohad.
- Kasutajal on võimalik osta endale dünaamiline perioodipilet, mille kehtivust saab ta ise muuta ja Ostukorvis saab ta muuta ka kehtivuse algus ja lõpp kuupäevasad.

Lisas 1, on iga testiloo kohta olemas ka visualiseeritud sammud.

Tabel 5. Testilugu Otsi\_reisi.

ID	Otsi_reisi	Nõuded testi kohta	
Eesmärk	Kinnitada, et kasutaja saab reisi otsida		
Eeltingimus			
Sammud:	Oodatud tulemus:		Tegelik tulemus:
Ava veebileht <a href="http://pilet.elron.ee">pilet.elron.ee</a>	Laetakse veebileht <a href="http://pilet.elron.ee">pilet.elron.ee</a>		OK
Sisesta Lähtekoht Tallinn; Sihtkoht Tartu; Kuupäev päev pärast tänast	Välja täidetakse valitud väärtustega		OK
Vajuta nuppu „Otsi“	Kuvatakse leht „Sõiduplaan“, kust on näha reisi Tallinn -> Tartu ja kuupäeva (Päev pärast tänast)		OK

Leheküljel 43 on toodud testiloo visualiseeritud sammud.

Tabel 6. Testilugu Vali\_otsitud\_reis.

ID	Vali_otsitud_reis	Nõuded testi kohta	
Eesmärk	Kinnitada, et kasutajal on võimalik avada ja sulgeda sõiduplaani vaates reisi detailvaadet ja minna pileti ostmise lehele		
Eeltingimus	Kasutaja saab reisi otsida (Otsi_reisi)		
Sammud:	Oodatud tulemus:	Tegelik tulemus:	
Vali esimene reis ja vajuta nuppu „Detailid“	Kasutajale avaneb detailid vaade, kust on näha, kus peatustest peatub rong, mis kellaajal ja kui palju maksavad erinevad piletid. Kasutaja jääb samale lehele	OK	
Vajuta nuppu „Peida“	Sulgeb detailid vaate	OK	
Vajuta nuppu „Osta“	Kasutaja läheb lehele „Pileti ostmise“	OK	

Leheküljel 44 on toodud testiloo visualiseeritud sammud.



Tabel 7. Testilugu Ühe\_korra\_pilet.

ID	Ühe_korra_pilet	Nõuded testi kohta	
Eesmärk	Kinnitada, et kasutaja saab lisada ostukorvi Ühe korra pileti ja Ühe korra sooduspileti		
Eeltingimus	Kasutaja on lehel „Pileti ostmine“ (Vali_otsitud_reis)		
Sammud:	Oodatud tulemus:		Tegelik tulemus:
Vajuta kaks korda nuppu „+“ Ühe korra pilet real	Ühe korra pilet real muutub oranž hind ja valges kastis kuvatakse number 2		OK
Vajuta nuppu „-“, Ühe korra pilet real	Ühe korra pilet real muutub oranž hind kaks korda väiksemaks ja valges kastis kuvatakse number 1		OK
Vajuta kaks korda nuppu „+“ Ühe korra sooduspilet real	Ühe korra sooduspilet real muutub oranž hind ja valges kastis kuvatakse number 2		OK
Vajuta nuppu „-“, Ühe korra sooduspilet real	Ühe korra sooduspilet real muutub oranž hind kaks korda väiksemaks ja valges kastis kuvatakse number 1		OK
Vajuta nuppu „Lisa ostukorvi“	Kuvatakse leht „Ostukorv“. Ülesse paremasse nurka tuleb nupp „Ostukorv“. Lehel ostukorv kuvatakse pileteid: Ühe korra sooduspilet (1 tk) ja Ühe korra pilet (1 tk)		OK

Leheküljel 45 on toodud testiloo visualiseeritud sammud.

Tabel 8. Testilugu I\_klassi\_pilet.

ID	I_klassi_pilet	Nõuded testi kohta	
Eesmärk	Kinnitada, et kasutaja saab lisada ostukorvi esimese klassi pileti ja		
Eeltingimus	Kasutaja on lehel „Pileti ostmine“ (Vali_otsitud_reis)		
Sammud:	Oodatud tulemus:		Tegelik tulemus:
Vajuta nuppu „Vali kohad“	Avaneb modaalaken		OK
Vajuta ühele vabale istekohale	Valitud istekoht muutub kollaseks		OK
Vajuta nuppu „Eemalda kõik“	Valitud istekoht kuvatakse nagu ülejäänud vabad istekohad		OK
Vajuta ühele vabale istekohale	Valitud istekoht muutub kollaseks		OK
Vajuta nuppu „Kinnita“	Modaalaken sulgub ja real I klassi pilet tekib number 1 nupu „Muuda“ ette		OK
Vajuta piletile „Nullpilet“	Kuvatakse info nullpileti kohta		OK
Vajuta real nullpilet nupule „Vali kohad“	Avaneb modaalaken		OK
Vajuta ühele vabale istekohale	Valitud istekoht muutub kollaseks		OK
Vajuta nuppu „Eemalda kõik“	Valitud istekoht kuvatakse nagu ülejäänud vabad istekohad		OK
Vajuta ühele vabale istekohale	Valitud istekoht muutub kollaseks		OK
Vajuta nuppu „Kinnita“	Modaalaken sulgub ja real nullpilet tekib number 1 ennem nuppu „Muuda“		OK
Vajuta nuppu „Lisa ostukorvi“	Kuvatakse „Ostukorv“. Ülesse paremasse nurka tekib nupp „Ostukorv“. Lehel kuvatakse kaks piletit: Nullpilet (teeustasu 1 EUR) (1 tk) ja I klassi pilet (1 tk)		OK

Leheküljel 48 on toodud testiloo visualiseeritud sammud.

Tabel 9. Testilugu Dünaamiline\_perioodipilet.

ID	Dünaamiline_perioodipilet	Nõuded testi kohta	
Eesmärk	Kinnitada, et kasutaja saab lisada ostukorvi dünaamilise perioodipileti ja saab muuta kehtivuse perioodi lehel „Ostukorv“		
Eeltingimus	Kasutaja on lehel „Pileti ostmine“ (Vali_otsitud_reis)		
Sammud:	Oodatud tulemus:	Tegelik tulemus:	
Vajuta nuppu, mis on kasti „Vali päevade arv“ real	Avaneb menüü, kus saab valida päevade arvu	OK	
Vali sealt päevade arv	„Vali päevade arv“ asemele tuleb valitud päevade arv koos pileti hinnaga ja ka paremale pool muutub perioodipileti hind valituga võrdseks	OK	
Vajuta nuppu „Lisa ostukorvi“	Avaneb leht „Ostukorv“. Lehel on üks pileti: perioodipilet (1 tk,(valitud päevade arv) päeva)	OK	
Vajuta nuppu, mis on „Kehtivuse lõpp (k.a):“ all ja muuda kehtivuse lõppkuupäeva ühe päeva võrra varasemaks	Muutub pileti hind ja ka kuvatud pileti: perioodipilet (1 tk,(valitud päevade arv lahutada üks päev ) päeva )	<b>!OK</b>	

Leheküljel 53 on toodud testiloo visualiseeritud sammud.

### 4.3 Automaattest kirjeldatud testiloole

Automaatse funktsionaalse testi kirjutam järgnevale: Testilugu  
Dünaamiline\_ perioodipilet.

```
public class When_User_Buys_Period_Ticket_And_Changes_Its_Period : SpecsBase
{
    private CartIndexPage _cartIndexPage;
    private string _initialPeriodTicketPeriod;
    private string _periodTicketSalesPagePrice;
    protected override void Given()
    {
        var scheduleIndexPage = ScheduleIndexPage
            .NavigateTo(SUT)
            .SearchSchedule();

        var salesIndexPage = scheduleIndexPage
            .GoToSalesForFirstTrain()
            .SelectOnePeriodTicket();

        _periodTicketSalesPagePrice =
        salesIndexPage.GetPeriodTicketPriceInTotal();
        _cartIndexPage = salesIndexPage.AddItemToCart();
        _initialPeriodTicketPeriod =
        _cartIndexPage.GetPeriodTicketPeriod();
    }

    protected override void When()
    {
        _cartIndexPage.SelectPeriodTicketPeriodEndDate();
    }

    [Test]
    public void Then_Period_Ticket_Period_Should_Change()
    {
        // Act
        var changedPeriodTicketPeriod =
        _cartIndexPage.GetPeriodTicketPeriod();

        // Assert
        Assert.That(changedPeriodTicketPeriod,
        Is.Not.EqualTo(_initialPeriodTicketPeriod));
    }

    [Test]
    public void Then_Period_Ticket_Price_Should_Change()
    {
        // Act
        var periodTicketCartPagePrice =
        _cartIndexPage.GetPeriodTicketPrice();

        // Assert
        Assert.That(periodTicketCartPagePrice,
        Is.Not.EqualTo(_periodTicketSalesPagePrice));
    }
}
```

Joonis 4. Dünaamilise perioodipileti automaatne test

## 4.4 Tulemused

Selles peatükis käsitletud testilood katavad ainult osa funktsionaalsusest, mis tähendab, et ainult nendele ei saa toetuda testitava tarkvara kvaliteedi hindamisel. Testilood on kirjutatud kasutades musta kasti tehnikaid, seega tarkvara sisemisele loogikale ja arhitektuurile pole tähelepanu pööratud. Adekvaatse hinnangu andmiseks on vaja kontrollida ka tarkvara sisemisi seoseid ehk lisaks musta kasti tehnikatele oleks vaja kasutada ka valge kasti tehnikaid. Loodud automaatne test Dünaamilise perioodipileti automaatne test hõlmab endas halli kasti tehnikaid, kuna siin on teada vaid funktsionaalsete testide sisemine loogika ja arhitektuur, kuna tarkvara enda oma pole automaatse testi kirjutamisel oluline.

Põhjus, miks valisin just Testilugu Dünaamiline\_perioodipilet. automatiseerimiseks on selles, et Elroni piletimüügisüsteemi arendamisel esines sellega probleeme ja sellest tulenevalt tekkis vajadus antud testiloo automatiseerimisele. Toodud praktilist näidet saaks mitmeti täiendada, näiteks luues testilood kogu piletimüügisüsteemi funktsionaalsusele ja mitte ainult kasutaja vaatest sõltuvalt, vaid pöörates tähelepanu ka Elroni administraatori vaatele. Kogu funktsionaalsust testivate testilugude kirjutamine ja ka läbi viimine on väga mahukas, mis tõttu valisin ainult osa sellest.

## 4.5 Kasutatud tehnikad ja tööriistad

Toodud praktilise näite sooritamisel kasutasin järgmisi tehnikaid:

- Visuaalne testimine
- Vigade oletamise tehnika
- Uuriv testimine
- Manuaalne testimine
- Automaatne testimine

Toodud praktilise näite sooritamisel kasutasin järgmisi tööriistasid:

- Visual Studio 2015
- SpecsFor
- Programmeerimiskeeleks on C#
- Google Chrome
- Mozilla Firefox
- Internet Explorer
- Opera
- Microsoft Edge
- Selenium

## 5 Kokkuvõte

Selles bakalaureusetöös on välja toodud testimise erinevad võimaluse tarkvara manuaaltestimise näitel. Lühidalt on lahti seletatud kolme erineva kasti tähendused testimisel – musta, valge ja halli -, neid oma vahel võrreldud ning välja valitud üks, mille kohta on kirjutatud täpsemalt. Välja valiti musta kasti meetodi tehnikad ning keskenduti erinevate tehnikate võrdlemisele, mille käigus toodi välja erinevate tehnikate negatiivsed ja positiivsed küljed.

Praktilises näites kasutati nii manuaalset kui ka automaatset testimist ning pigem musta kasti meetodit. Testide automatiseerimine on tänapäeval pea möödapääsmatu, kuna suure hulga manuaalsete testijate rakendamine on väga kulukas ja ei pruugi olla nii efektiivne, kuid tuleb siiski meeles pidada, et testide automatiseerimine ei tohiks kunagi täielikult eemaldada manuaalset testimist. Automaatsel testil, teisisõnu arvutil, kellele on öeldud mida peab vaatama ja kuhu peab vajutama, puudub inimestele omane intuitsioon. Sellepärast märkavadki testijad, kes kasutavad nii manuaalset kui ka automaatset testimist rohkem vigu ja on oma töös kui ka tööturul rohkem hinnatud ja produktiivsemad, kui need kes kasutavad ainult üht kahest variandist.

### 5.1 Edasised sammud

Seda bakalaureusetööd saaks edasi arendada mitmeti. Loetelus on toodud erinevad võimalused:

- Kirjeldada kogu Elroni piletimüügisüsteemi funktsionaalsust ja kirjutada selle kontrollimiseks testilood ning need läbi viia.
- Automatiseerida kõik Praktilises näites toodud testilood.
- Uurida ja kirjutada lahti valge ja ka halli kasti meetodi tehnikad.
- Kasutada selles töös toodud materjale ning leida oma arendusmeeskonnale sobivaim viis välja anda võimalikult puhas ja korrektne tarkvara.

## Kasutatud kirjandus

- [1] *Software testing*. (2016). [WWW] <https://www.techopedia.com/definition/17681/software-testing> (13.04.2016)
- [2] *Black box testing*. (2010). [WWW] <http://softwaretestingfundamentals.com/black-box-testing> (01.04.2016)
- [3] *Software testing – Methods*. (2016) [WWW] [http://www.tutorialspoint.com/software\\_testing/software\\_testing\\_methods.htm](http://www.tutorialspoint.com/software_testing/software_testing_methods.htm). (01.04.2016)
- [4] *Software testing – Levels*. (2016) [WWW] [http://www.tutorialspoint.com/software\\_testing/software\\_testing\\_levels.htm](http://www.tutorialspoint.com/software_testing/software_testing_levels.htm) (01.04.2016)
- [5] Markvardt M., (2006) *Tarkvara testimise käsitlev juhendmaterjal. – ASA Quality Services OÜ* [online] [https://webcache.googleusercontent.com/search?q=cache:4-Jc6r31IJ0J:https://www.mkm.ee/sites/default/files/tarkvara\\_testimise\\_juhis\\_-\\_koopia.doc+&cd=2&hl=et&ct=clnk&gl=ee](https://webcache.googleusercontent.com/search?q=cache:4-Jc6r31IJ0J:https://www.mkm.ee/sites/default/files/tarkvara_testimise_juhis_-_koopia.doc+&cd=2&hl=et&ct=clnk&gl=ee) (01.04.2016)
- [6] *Why is software testing necessary?* (2016). [WWW] <http://istqbexamcertification.com/why-is-testing-necessary> (01.04.2016)
- [7] *Boris Beizer*. (2010). [WWW] <http://www.softwarequotes.com/showquotes.aspx?id=558&name=> (21.04.2016)
- [8] Pan, J. (1999). *Dependable Embedded Systems*. Carnegie Mellon University. [WWW] [https://users.ece.cmu.edu/~koopman/des\\_s99/sw\\_testing/](https://users.ece.cmu.edu/~koopman/des_s99/sw_testing/) (21.04.2016)
- [9] Williams, L., (2006). *Testing overview and Black-Box Testing Techniques*. [WWW] <http://agile.csc.ncsu.edu/SEMaterials/BlackBox.pdf> (21.04.2016)
- [10] Kaljuma, K., (2008). *Ülevaade testimisetechnikatest* : Seminaritöö. Tallinn. Tallinna Ülikool.
- [11] *Gray box testing fundamentals*. (2010). [WWW] <http://softwaretestingfundamentals.com/gray-box-testing/> (22.04.2016)
- [12] Acharya, S., Pandya, V., (2016) *Bridge between Black Box and White Box – Gray Box Testing Technique*. [WWW] <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.303.4479&rep=rep1&type=pdf> (22.04.2016)
- [13] *Pros and Cons of Black Box Testing Methods*. (2014). [WWW] <http://kagrana.com/blog/pros-and-cons-of-black-box-testing-methods/> (22.04.2016)
- [14] *Manual Testing*. (2016). [WWW] <http://www.onestoptesting.com/manual-testing/overview.asp> (26.04.2016)
- [15] Silberman, J., (2015). *The Advantages of Manual Vs. Automated Testing*. [WWW] <https://www.blazemeter.com/blog/advantages-manual-vs-automated-testing> (26.04.2016)
- [16] Arsenault, R., (2014). *When You Should Choose Manual vs. Automated Testing*. [WWW] <https://www.utest.com/articles/when-you-should-choose-manual-vs-automated-testing> (26.04.2016)
- [17] Marick, B., (2014). *When Should a Test Be Automated?* [WWW] <https://communities->

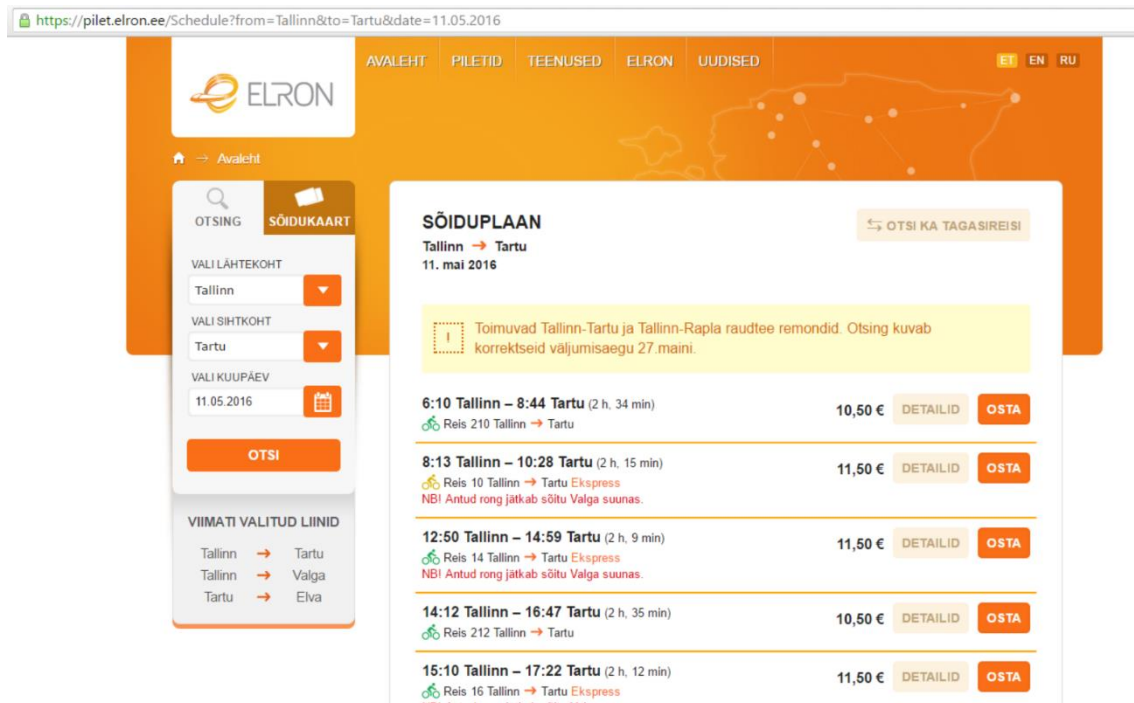
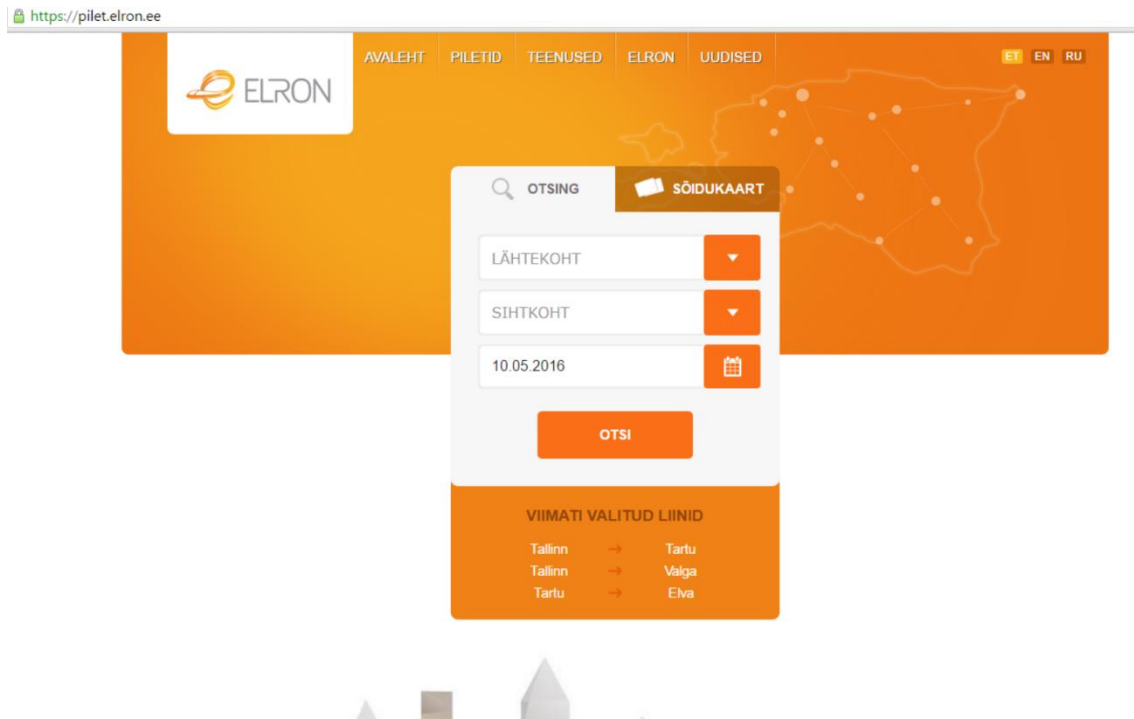


- 3bd5.kxcdn.com/sites/default/files/article/file/2014/When%20Should%20a%20Test%20Be%20Automated.pdf (26.04.2016)
- [18] Soko, B., (2015). *Test Automation vs Manual Testing in Software Development* [WWW] <http://www.embedded.com/electronics-blogs/say-what-/4439669/Test-Automation-vs-Manual-Testing-in-Software-Development-> (26.04.2016)
- [19] *Automation Testing vs Manual Testing* (2014). [WWW] <http://www.softwaretestingclass.com/automation-testing-vs-manual-testing/> (26.04.2016)
- [20] *Manual Testing vs Automated Testing: Pros and Cons* (2015). [WWW] <http://www.optimusinfo.com/blog/manual-testing-vs-automated-testing-pros-and-cons/> (26.04.2016)
- [21] *Automated vs. Manual Testing: The Pros and Cons of Each* (2013). [WWW] <http://www.base36.com/2013/03/automated-vs-manual-testing-the-pros-and-cons-of-each/> (27.04.2016)
- [22] *Manual Testing vs Automated Testing: Pros and Cons* (2015). [WWW] <http://www.optimusinfo.com/blog/manual-testing-vs-automated-testing-pros-and-cons/> (27.04.2016)
- [23] *The Explosion of the Ariane 5* (2000). [WWW] <https://www.ima.umn.edu/~arnold/disasters/ariane.html> (27.04.2016)
- [24] *Software QA and Testing Frequently-Asked-Questions, Part 1* (2016). [WWW] <http://www.softwareqatest.com/qatfaq1.html> (27.04.2016)
- [25] *Introduction to Pairwise Testing* (2015) [WWW] <https://inductive.no/pairwiser/knowledge-base/introduction-to-pairwise-testing/> (28.04.2016)
- [26] Allott., S., K., (2007) Test Design Workshop Module 4 Pairwise Testing [WWW] <http://www.bcs.org/upload/pdf/testdesign-nov09.pdf> (29.04.2016)
- [27] *Software Engineering and Testing* (2010). Agarwal, B. B. ,Tayal, S. P. ,Gupta, M.
- [28] Aasna, M. (2006) *Staatilised ja dünaamilised tarkvara testimise tehnikad* : seminaritöö. Tallinn, Tallinna ülikool.
- [29] *What is Equivalence Class Partitioning (ECP)? What are the advantages and Limitations?* (2013). [WWW] <http://qeworks.com/equivalence-class-partitioning/> (02.05.2016)
- [30] *Black Box Testing Techniques* (2004). [WWW] [http://www.kabinfol.net/kabinfol\\_Bva.asp](http://www.kabinfol.net/kabinfol_Bva.asp) (02.05.2016)
- [31] Mogyorodi, G., E., Math, B., M., B., A., (2010) *Requirements-based Testing – Cause-Effect Graphing* [WWW] [http://barbie.uta.edu/~mehra/59\\_RBT\\_Cause-Effect\\_Graphing2.pdf](http://barbie.uta.edu/~mehra/59_RBT_Cause-Effect_Graphing2.pdf) (03.05.2016)
- [32] *Cause-Effect Graph* (2016). [WWW] [http://www.tutorialspoint.com/software\\_testing\\_dictionary/cause\\_effect\\_graph.htm](http://www.tutorialspoint.com/software_testing_dictionary/cause_effect_graph.htm) (03.05.2016)
- [33] Zavistanavičiuité, R., (2016). Testing based on error guessing [WWW] [http://webcache.googleusercontent.com/search?q=cache:-oKgL3ixPZ4J:www.elen.ktu.lt/studentai/lib/exe/fetch.php%3Fmedia%3D10\\_testing\\_basjed\\_on\\_error\\_guessing.ppt+&cd=1&hl=et&ct=clnk&gl=ee](http://webcache.googleusercontent.com/search?q=cache:-oKgL3ixPZ4J:www.elen.ktu.lt/studentai/lib/exe/fetch.php%3Fmedia%3D10_testing_basjed_on_error_guessing.ppt+&cd=1&hl=et&ct=clnk&gl=ee) (03.05.2016)

[34] Elron (2016). [WWW] <http://elron.ee/elron/> (05.05.2016)

# Lisa 1 – Visualiseeritud testilood

## 1. Testilugu Otsi\_reisi.



## 2. Testilugu Vali\_otsitud\_reis.

Esimesed kaks pilti on samad, mis Otsi\_reisi testiloos.

https://pilet.elron.ee/Schedule?from=Tallinn&to=Tartu&date=11.05.2016

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

**OTSI**

VIIMATI VALITUD LIINID

- Tallinn → Tartu
- Tallinn → Valga
- Tartu → Elva

Toimuvad Tallinn-Tartu ja Tallinn-Rapla raudtee remondid. Otsing kuvab korrektseid väljumisaegu 27. maini.

**6:10 Tallinn – 8:44 Tartu** (2 h, 34 min) 10,50 € PEIDA OSTA

Reis 210 Tallinn → Tartu

**210 Tallinn → Tartu**

SÕIDUPLAAN:

**6:10 Tallinn**

6:16 Kitseküla

6:21 Ülemiste

6:39 Kehra

6:51 Aegviidu

6:54 Neljärke

6:58 Jäneda

7:03 Lehtse

7:11 Tapa

7:22 Tamsalu

7:30 Kiltsi

7:37 Rakke

7:44 Vägeva

7:51 Pejja

8:00 Jõgeva

8:15 Kaarepere

8:24 Tabivere

8:34 Kärkna

**8:44 Tartu**

PILETIHINNAD:

Kuvatakse hindasid, mis kehtivad rongis (ühe korra piletid) ja Balti Jaama kassas (ühe korra ja 30 päeva piletid). Rongist müüdavate 30 päeva piletite hinnad on leitavad kodulehe pilethindade lehelt. Internetist ostes on hinnad alolevatest veel 10% soodsamad.

PILETID	HIND	HIND SÕIDUKAARDIGA
ÜHE KORRA PILET	10,50 €	9,45 €
ÜHE KORRA SOODUSPILET	7,40 €	6,66 €
I KLASSI PILET	12,60 €	11,34 €
NULLPILET (TEENUSTASU 1 EUR)	1 €	1 €
PERIOODIPILET (30 PÄEVA)	-	208,25 €
SOODUS PERIOODIPILET (30 PÄEVA)	-	146,13 €

https://pilet.elron.ee/Schedule?from=Tallinn&to=Tartu&date=11.05.2016

AVALEHT PILETID TEENUSED ELRON UUDISED

ET EN RU

ELRON

→ Avaleht

OTSING SÕIDUKAART

VALI LÄHTEKOHT  
Tallinn

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

**OTSI**

VIIMATI VALITUD LIINID

- Tallinn → Tartu
- Tallinn → Valga
- Tartu → Elva

**SÕIDUPLAAN** OTSI KA TAGASIREISI

Tallinn → Tartu  
11. mai 2016

Toimuvad Tallinn-Tartu ja Tallinn-Rapla raudtee remondid. Otsing kuvab korrektseid väljumisaegu 27. maini.

**6:10 Tallinn – 8:44 Tartu** (2 h, 34 min) 10,50 € DETAILID OSTA

Reis 210 Tallinn → Tartu

---

**8:13 Tallinn – 10:28 Tartu** (2 h, 15 min) 11,50 € DETAILID OSTA

Reis 10 Tallinn → Tartu Ekspress  
NBI Antud rong jätkab sõitu Valga suunas.

---

**12:50 Tallinn – 14:59 Tartu** (2 h, 9 min) 11,50 € DETAILID OSTA

Reis 14 Tallinn → Tartu Ekspress  
NBI Antud rong jätkab sõitu Valga suunas.

---

**14:12 Tallinn – 16:47 Tartu** (2 h, 35 min) 10,50 € DETAILID OSTA

Reis 212 Tallinn → Tartu

---

**15:10 Tallinn – 17:22 Tartu** (2 h, 12 min) 11,50 € DETAILID OSTA

Reis 16 Tallinn → Tartu Ekspress  
NBI Antud rong jätkab sõitu Valga suunas.

OTSING SÕIDUKAART

VALI LÄHTEKOHT  
Tallinn

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

OTSI

VIIMATI VALITUD LIINID  
Tallinn → Tartu  
Tallinn → Valga  
Tartu → Elva

### PILETI OSTMINE

NBI Sõidukaardile ostes on kõik piletihinnad 10% soodsamad. Sisesta soodustuse saamiseks Sõidukaardi number.

Sõidukaardi number  RAKENDA

11. mai 2016  
6:10 Tallinn – 8:44 Tartu (2 h, 34 min)  
Reis 210 Tallinn → Tartu

#### ÜHESIKIPILETID

ÜHE KORRA PILET	9,45 €	-	0	+	0,00 €
ÜHE KORRA SOODUSPILET	6,66 €	-	0	+	0,00 €
I KLASSI PILET	11,34 €	0	VALI KOHAD	+	0,00 €

Vabu kohti: 15

#### PERIOODIPILETID

NBI Alguskuupäeva saab muuta ostukorvis! Kõikide soodustustega hinda kuvatakse, kui olete sisestanud ja rakendanud Sõidukaardi numbr.

PERIOODIPILET  
Vali päevade arv  0,00 €

SOODUS PERIOODIPILET  
Vali päevade arv  0,00 €

### 3. Testilugu Ühe\_korra\_pilet.

Esimesed kaks pilti on samad mis Otsi\_reisi testiloos ja kolmas pilt on sama, mis Vali\_otstitud\_reis testiloos.

OTSING SÕIDUKAART

VALI LÄHTEKOHT  
Tallinn

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

OTSI

VIIMATI VALITUD LIINID  
Tallinn → Tartu  
Tallinn → Valga  
Tartu → Elva

### PILETI OSTMINE

NBI Sõidukaardile ostes on kõik piletihinnad 10% soodsamad. Sisesta soodustuse saamiseks Sõidukaardi number.

Sõidukaardi number  RAKENDA

11. mai 2016  
6:10 Tallinn – 8:44 Tartu (2 h, 34 min)  
Reis 210 Tallinn → Tartu

#### ÜHESIKIPILETID

ÜHE KORRA PILET	9,45 €	-	2	+	18,90 €
ÜHE KORRA SOODUSPILET	6,66 €	-	0	+	0,00 €
I KLASSI PILET	11,34 €	0	VALI KOHAD	+	0,00 €

Vabu kohti: 15

#### PERIOODIPILETID

NBI Alguskuupäeva saab muuta ostukorvis! Kõikide soodustustega hinda kuvatakse, kui olete sisestanud ja rakendanud Sõidukaardi numbr.

PERIOODIPILET  
Vali päevade arv  0,00 €

SOODUS PERIOODIPILET  
Vali päevade arv  0,00 €

Avaleht

OTSING SÕIDUKAART

VALI LÄHTEKOHT  
Tallinn

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

OTSI

VIIMATI VALITUD LIINID  
Tallinn → Tartu  
Tallinn → Valga  
Tartu → Elva

### PILETI OSTMINE

NBI Sõidukaardile ostes on kõik pilethinnad 10% soodsamad.  
Sisesta soodustuse saamiseks Sõidukaardi number:

Sõidukaardi number  RAKENDA

11. mai 2016  
6:10 Tallinn – 8:44 Tartu (2 h, 34 min)  
Reis 210 Tallinn → Tartu

#### ÜKSIKPILETID

ÜHE KORRA PILET	9,45 €	-	1	+	9,45 €
ÜHE KORRA SOODUSPILET	6,66 €	-	0	+	0,00 €
I KLASSI PILET	11,34 €	0	VALI KOHAD		0,00 €

Vabu kohti: 15

#### PERIOODIPILETID

NBI Alguskuupäeva saab muuta ostukorvis! Kõikide soodustustega hinda kuvatakse, kui olete sisestanud ja rakendanud Sõidukaardi numbril.

PERIOODIPILET  
Vali päevade arv  0,00 €

SOODUS PERIOODIPILET  
Vali päevade arv  0,00 €

Avaleht

OTSING SÕIDUKAART

VALI LÄHTEKOHT  
Tallinn

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

OTSI

VIIMATI VALITUD LIINID  
Tallinn → Tartu  
Tallinn → Valga  
Tartu → Elva

### PILETI OSTMINE

NBI Sõidukaardile ostes on kõik pilethinnad 10% soodsamad.  
Sisesta soodustuse saamiseks Sõidukaardi number:

Sõidukaardi number  RAKENDA

11. mai 2016  
6:10 Tallinn – 8:44 Tartu (2 h, 34 min)  
Reis 210 Tallinn → Tartu

#### ÜKSIKPILETID

ÜHE KORRA PILET	9,45 €	-	1	+	9,45 €
ÜHE KORRA SOODUSPILET	6,66 €	-	2	+	13,32 €
I KLASSI PILET	11,34 €	0	VALI KOHAD		0,00 €

Vabu kohti: 15

#### PERIOODIPILETID

NBI Alguskuupäeva saab muuta ostukorvis! Kõikide soodustustega hinda kuvatakse, kui olete sisestanud ja rakendanud Sõidukaardi numbril.

PERIOODIPILET  
Vali päevade arv  0,00 €

SOODUS PERIOODIPILET  
Vali päevade arv  0,00 €

Avaleht

OTSING SÖIDUKAART

VALI LÄHTEKOHT  
Tallinn

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

OTSI

VIIMATI VALITUD LIINID  
Tallinn → Tartu  
Tallinn → Valga  
Tartu → Elva

### PILETI OSTMINE

NBI Sõidukaardile ostes on kõik pilethinnad 10% soodsamad. Sisesta soodustuse saamiseks Sõidukaardi number:

Sõidukaardi number  RAKENDA

11. mai 2016  
6:10 Tallinn – 8:44 Tartu (2 h, 34 min)  
Reis 210 Tallinn → Tartu

#### ÜKSIKPILETID

ÜHE KORRA PILET	9,45 €	- 1 +	9,45 €
ÜHE KORRA SOODUSPILET	6,66 €	- 1 +	6,66 €
I KLASSI PILET	11,34 €	0	0,00 €

VALI KOHAD  
Vabu kohti: 15

#### PERIOODIPILETID

NBI Alguskuupäeva saab muuta ostukorvis! Kõikide soodustustega hinda kuvatakse, kui olete sisestanud ja rakendanud Sõidukaardi numbrit.

PERIOODIPILET  
Vali päevade arv  0,00 €

SOODUS PERIOODIPILET  
Vali päevade arv  0,00 €

AVALEHT PILETID TEENUSED ELRON UUDISED ET EN RU

OSTUKORV: 16,11 €

Avaleht

OTSING SÖIDUKAART

VALI LÄHTEKOHT  
Tallinn

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

OTSI

VIIMATI VALITUD LIINID  
Tallinn → Tartu  
Tallinn → Valga  
Tartu → Elva

### OSTUKORV

Ostu sooritamiseks on jäänud aega 15 minutit

11. mai 2016  
6:10 Tsoon 1 – 8:44 Tartu (2 h, 34 min)  
Reis 210 Tallinn → Tartu

ÜHE KORRA PILET (1 tk)	9,45 €	×
ÜHE KORRA SOODUSPILET (1 tk)	6,66 €	×

#### KÄTTETOIMETAMINE

Salvesta piletid Sõidukaardile:

Sõidukaardi number

või saada piletid e-mailile:

Sinu e-mail

Kui täidate mõlemad väljad, salvestatakse piletid Sõidukaardile ja saadetakse kinnitus e-mailile. NBI E-maili aadress salvestatakse ja kasutatakse vajadusel reisinfo edastamiseks.

#### 4. Testilugu I\_klassi\_pilet.

Esimesed kaks pilti on samad mis Otsi\_reisi testiloos ja kolmas pilt on sama, mis Vali\_otsitud\_reis testiloos.

The screenshot shows a web browser window with the URL <https://pilet.elron.ee/Sales?Trains%5B0%5D.From=Tallinn&Trains%5B0%5D.To=Tartu&Trains%5B0%5D.Date=2016-05-11&Trains%5B0%5D.Number=210&Trains%5B0%5D.Composi>. A modal window is open with the title "Palun valige kohad ning kinnitamiseks vajutage nupul 'Kinnita'". The modal displays "TARTU" and a 4x4 grid of seat numbers: 16, 15, 14, 13; 12, 11, 10, 9; 8, 7, 6, 5; 4, 3, 2, 1. The number 4 is highlighted in red. Below the grid are two orange bars and buttons "EEMALDA KÕIK" and "KINNITA". The background shows a sidebar with "OTSING" and "VALI LÄHTEKOHT" (Tallinn, Tartu) and "VALI SIHTKOHT" (Tartu, Valga, Elva). The main content area shows "ÜKSIKPILETID" with prices for "ÜHE KORRA PILET" (9,45 €), "ÜHE KORRA SOODUSPILET" (6,66 €), and "I KLASSI PILET" (11,34 €). There are also "PERIOODIPILETID" and a "VALI KOHAD" button.

This screenshot is identical to the one above, but in the modal window, the seat number 10 is highlighted in yellow. The rest of the interface, including the sidebar and main content area, remains the same.



Palun valige kohad ning kinnitamiseks vajutage nupul 'Kinnita'

**TARTU**

16	15	14	13
12	11	10	9
8	7	6	5
4	3	2	1

EEMALDA KÕIK KINNITA

**ÜKSIKPILETID**

ÜHE KORRA PILET	9,45 €	- 0 +	0,00 €
ÜHE KORRA SOODUSPILET	6,66 €	- 0 +	0,00 €
I KLASSI PILET	11,34 €	0	0,00 €

**PERIOODIPILETID**

NBI Alguskuupäeva saab muuta ostukorvis! Kõikide soodustustega hinda kuvatakse, kui olete sisestanud ja rakendanud Sõidukaardi numbrit.

Palun valige kohad ning kinnitamiseks vajutage nupul 'Kinnita'

**TARTU**

16	15	14	13
12	11	10	9
8	7	6	5
4	3	2	1

EEMALDA KÕIK KINNITA

**ÜKSIKPILETID**

ÜHE KORRA PILET	9,45 €	- 0 +	0,00 €
ÜHE KORRA SOODUSPILET	6,66 €	- 0 +	0,00 €
I KLASSI PILET	11,34 €	0	0,00 €

**PERIOODIPILETID**

NBI Alguskuupäeva saab muuta ostukorvis! Kõikide soodustustega hinda kuvatakse, kui olete sisestanud ja rakendanud Sõidukaardi numbrit.

ELRON

AVALEHT PILETID TEENUSED ELRON UUDISED ET EN RU

→ Avalchit

OTSING SÖIDUKAART

VALI LÄHTEKOHT  
Tallinn

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

OTSI

VIIMATI VALITUD LIINID  
Tallinn → Tartu  
Tallinn → Valga  
Tartu → Elva

### PILETI OSTMINE

NBI Sõidukaardile ostes on kõik piletihinna 10% soodsamad. Sisesta soodustuse saamiseks Sõidukaardi number:

Sõidukaardi number  RAKENDA

11. mai 2016  
6:10 Tallinn – 8:44 Tartu (2 h, 34 min)  
Reis 210 Tallinn → Tartu

#### ÜKSIKPILETID

ÜHE KORRA PILET	9,45 €	- 0 +	0,00 €
ÜHE KORRA SOODUSPILET	6,66 €	- 0 +	0,00 €
I KLASSI PILET Istekohad: 11	11,34 €	1 MUUDA	11,34 € Vabu kohti: 15
NULLPILET (teenustasu 1 EUR) ▶	1,00 €	0 VALI KOHAD	0,00 € Vabu kohti: 15

PERIODIPILETID

Palun valige kohad ning kinnitamiseks vajutage nupul 'Kinnita'

↑ TARTU

16	15	14	13
12	11	10	9
8	7	6	5
4	3	2	1

EEMALDA KÕIK KINNITA

#### ÜKSIKPILETID

ÜHE KORRA PILET	9,45 €	- 0 +	0,00 €
ÜHE KORRA SOODUSPILET	6,66 €	- 0 +	0,00 €
I KLASSI PILET Istekohad: 11	11,34 €	1 MUUDA	11,34 € Vabu kohti: 15
NULLPILET (teenustasu 1 EUR) ▶	1,00 €	0 VALI KOHAD	0,00 € Vabu kohti: 15

PERIODIPILETID

https://pilet.elron.ee/Sales?Trains%5B0%5D.From=Tallinn&Trains%5B0%5D.To=Tartu&Trains%5B0%5D.Date=2016-05-11&Trains%5B0%5D.Number=210&Trains%5B0%5D.Composi

Palun valige kohad ning kinnitamiseks vajutage nupul 'Kinnita'

**TARTU**

16	15	14	13
12	11	10	9
8	7	6	5
4	3	2	1

EEMALDA KÕIK KINNITA

**ÜKSIKPILETID**

ÜHE KORRA PILET	9,45 €	- 0 +	0,00 €
ÜHE KORRA SOODUSPILET	6,66 €	- 0 +	0,00 €
I KLASSI PILET Istekohad: 33	11,34 €	1 MUUDA	11,34 € Vabu kohti: 15
NULLPILET (teenustasu 1 EUR) + Istekohad: 33	1,00 €	1 MUUDA	1,00 € Vabu kohti: 15

**PERIOODIPILETID**

https://pilet.elron.ee/Sales?Trains%5B0%5D.From=Tallinn&Trains%5B0%5D.To=Tartu&Trains%5B0%5D.Date=2016-05-11&Trains%5B0%5D.Number=210&Trains%5B0%5D.Composi

Palun valige kohad ning kinnitamiseks vajutage nupul 'Kinnita'

**TARTU**

16	15	14	13
12	11	10	9
8	7	6	5
4	3	2	1

EEMALDA KÕIK KINNITA

**ÜKSIKPILETID**

ÜHE KORRA PILET	9,45 €	- 0 +	0,00 €
ÜHE KORRA SOODUSPILET	6,66 €	- 0 +	0,00 €
I KLASSI PILET Istekohad: 33	11,34 €	1 MUUDA	11,34 € Vabu kohti: 15
NULLPILET (teenustasu 1 EUR) + Istekohad: 33	1,00 €	0 VALI KOHAD	0,00 € Vabu kohti: 15

**PERIOODIPILETID**

https://pilet.elron.ee/Sales?Trains%5B0%5D.From=Tallinn&Trains%5B0%5D.To=Tartu&Trains%5B0%5D.Date=2016-05-11&Trains%5B0%5D.Number=210&Trains%5B0%5D.Composi

Palun valige kohad ning kinnitamiseks vajutage nupul 'Kinnita'

**TARTU**

16	15	14	13
12	11	10	9
8	7	6	5
4	3	2	1

EEMALDA KÕIK **KINNITA**

**OTSING**

VALI LÄHTEKOHT  
Tallinn

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

**OTSI**

VIIMATI VALITUD LIINID

- Tallinn → Tartu
- Tallinn → Valga
- Tartu → Elva

**ÜKSIKPILETID**

ÜHE KORRA PILET	9,45 €	-	0	+	0,00 €
ÜHE KORRA SOODUSPILET	6,66 €	-	0	+	0,00 €
I KLASSI PILET Istekohad: [?]	11,34 €	1	<b>MUUDA</b>		11,34 €
Vabu kohti: 15					
NULLPILET (teenustasu 1 EUR) + Istekohad: [?]	1,00 €	1	<b>MUUDA</b>		1,00 €
Vabu kohti: 15					

**PERIODIPILETID**

https://pilet.elron.ee/Sales?Trains%5B0%5D.From=Tallinn&Trains%5B0%5D.To=Tartu&Trains%5B0%5D.Date=2016-05-11&Trains%5B0%5D.Number=210&Trains%5B0%5D.Composi

[AVALEHT](#) | [PILETID](#) | [TEENUSED](#) | [ELRON](#) | [UUDISED](#)

[ET](#) | [EN](#) | [RU](#)

→ Avalaht

**OTSING** | **SÕIDUKAART**

VALI LÄHTEKOHT  
Tallinn

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

**OTSI**

VIIMATI VALITUD LIINID

- Tallinn → Tartu
- Tallinn → Valga
- Tartu → Elva

**PILETI OSTMINE**

NBI Sõidukaardile ostes on kõik pilethinnad 10% soodsamad. Sisesta soodustuse saamiseks Sõidukaardi number

Sõidukaardi number  **RAKENDA**

**11. mai 2016**  
**6:10 Tallinn – 8:44 Tartu** (2 h, 34 min)  
🚲 Reis 210 Tallinn → Tartu

**ÜKSIKPILETID**

ÜHE KORRA PILET	9,45 €	-	0	+	0,00 €
ÜHE KORRA SOODUSPILET	6,66 €	-	0	+	0,00 €
I KLASSI PILET Istekohad: [?]	11,34 €	1	<b>MUUDA</b>		11,34 €
Vabu kohti: 15					
NULLPILET (teenustasu 1 EUR) + Istekohad: [?]	1,00 €	1	<b>MUUDA</b>		1,00 €
Vabu kohti: 15					

**PERIODIPILETID**

https://pilet.elron.ee/Cart

AVALEHT PILETID TEENUSED ELRON UUDISED ET EN RU

OSTUKORV: 12,34 €

→ Avaleht

OTSING SÖIDUKAART

VALI LÄHTEKOHT  
Tallinn

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

OTSI

VIIMATI VALITUD LIINID  
Tallinn → Tartu  
Tallinn → Valga  
Tartu → Elva

**OSTUKORV**

Ostu sooritamiseks on jäänud aega 15 minutit

11. mai 2016  
6:10 Tallinn – 8:44 Tartu (2 h, 34 min)  
Reis 210 Tallinn → Tartu

NULLPILET (teenustasu 1 EUR) (1 tk) 1,00 €

11. mai 2016  
6:10 Tsoon 1 – 8:44 Tartu (2 h, 34 min)  
Reis 210 Tallinn → Tartu

I KLASSI PILET (1 tk) 11,34 €

KÄTTETOIMETAMINE

Salvesta piletid Sõidukaardile:

## 5. Testilugu Dünaamiline\_perioodipilet.

Esimesed kaks pilti on samad mis Otsi\_reisi testiloos ja kolmas pilt on sama, mis Vali\_otsitud\_reis testiloos.

https://pilet.elron.ee/Sales?Trains%5B0%5D.From=Tallinn&Trains%5B0%5D.To=Tartu&Trains%5B0%5D.Date=2016-05-11&Trains%5B0%5D.Number=210&Trains%5B0%5D.Comp

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

OTSI

VIIMATI VALITUD LIINID  
Tallinn → Tartu  
Tallinn → Valga  
Tartu → Elva

11. mai 2016  
6:10 Tallinn – 8:44 Tartu (2 h, 34 min)  
Reis 210 Tallinn → Tartu

Vali päevade arv

1 päeva – 17,50 €			
2 päeva – 34,13 €			
3 päeva – 50,75 €	9,45 €	- 0 +	0,00 €
4 päeva – 67,38 €			
5 päeva – 84,00 €			
6 päeva – 92,75 €	6,66 €	- 0 +	0,00 €
7 päeva – 101,50 €			
8 päeva – 110,25 €			
9 päeva – 119,00 €			
10 päeva – 127,75 €	11,34 €	0	0,00 €
11 päeva – 136,50 €			
12 päeva – 145,25 €			
13 päeva – 152,25 €			
14 päeva – 159,25 €			
15 päeva – 162,75 €			
16 päeva – 166,25 €			
17 päeva – 169,75 €			
18 päeva – 173,25 €			
19 päeva – 176,75 €			

VALI KOHÄD  
Vabu kohti: 15

Kõikide soodustustega hinda kuvatakse, kui olete sisestanud ja

SOODUS PERIOODIPILET  
Vali päevade arv

KOKKU: 0,00 €

LISA OSTUKORVI

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

OTSI

VIIMATI VALITUD LIINID  
Tallinn → Tartu  
Tallinn → Valga  
Tartu → Elva

11. mai 2016  
6:10 Tallinn – 8:44 Tartu (2 h, 34 min)  
Reis 210 Tallinn → Tartu

ÜKSIKPILETID

ÜHE KORRA PILET	9,45 €	- 0 +	0,00 €
ÜHE KORRA SOODUSPILET	6,66 €	- 0 +	0,00 €
I KLASSI PILET	11,34 €	0	0,00 €

VALI KOHAD  
Vabu kohti: 15

PERIOODIPILETID

NBI Alguskuupäeva saab muuta ostukorvis! Kõikide soodustustega hinda kuvatakse, kui olete sisestanud ja rakendanud Sõidukaardi numbrit.

PERIOODIPILET  
7 päeva – 101,50 €

SOODUS PERIOODIPILET  
Vali päevade arv

KOKKU: 101,50 €

LISA OSTUKORVI

ELRON

AVALEHT PILETID TEENUSED ELRON UUDISED ET EN RU

OSTUKORV: 101,50 €

AVALEHT

OTSING SÕIDUKAART

VALI LÄHTEKOHT  
Tallinn

VALI SIHTKOHT  
Tartu

VALI KUUPÄEV  
11.05.2016

OTSI

VIIMATI VALITUD LIINID  
Tallinn → Tartu  
Tallinn → Valga  
Tartu → Elva

OSTUKORV

Ostu sooritamiseks on jäänud aega 15 minutit

Tsoon 1 ↔ Tartu

perioodipilet (1 tk, 7 päeva) 101,50 €

Vali kehtivuse algus: 11.05.2016 Kehtivuse lõpp (k.a): 17.05.2016

KÄTTETOIMETAMINE

Salvesta pilet Sõidukaardile:  
Sõidukaardi number

Soovi korral saadame Teile kinnituse ka e-malli teel:  
Sinu e-mail

NBI E-malli aadress salvestatakse ja kasutatakse vajadusel reisiinfo edastamiseks.



→ Avaleht

OTSING SÕIDUKAART

VALI LÄHTEKOHT

Tallinn

VALI SIHTKOHT

Tartu

VALI KUUUPÄEV

11.05.2016

OTSI

VIIMATI VALITUD LIINID

Tallinn → Tartu  
Tallinn → Valga  
Tartu → Elva

## OSTUKORV

Ostu sooritamiseks on jäänud aega 15 minutit

Tsoon 1 → Tartu

perioodipilet (1 tk,8 päeva)

110,25 €

Vali kehtivuse algus:

11.05.2016

Kehtivuse lõpp (k.a):

18.05.2016

## KÄTTETOIMETAMINE

Salvesta pilet Sõidukaardile:

Sõidukaardi number

Soovi korral saadame Teile kinnituse ka e-maili teel:

Sinu e-mail