

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond
Tarkvarateaduse instituut

Andreas Jürimäe 135060IAPB

RAHVATANTSUANSAMBLI LAOHALDUSTARKVARA

Bakalaureusetöö

Juhendaja: Gert Kanter
Tehnikateaduse
magister

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: Andreas Jürimäe

22.05.2017

Annotatsioon

Antud töö eesmärgiks on veebirakenduse loomine Eesti rahvatantsuansamblitele. Rakenduse kasutusalaaliseks tähtsuseks on rahvatantsuansamblite tegevuse lihtsustamine süsteemse ja kasutajasõbraliku veebipõhise laohaldusvõimaluse teel. Antud rakendus võimaldab anda reaalaajalist ülevaadet laoseisust ning tantsijatele väljastatud rõivastest. Töös on välja toodud rakenduse peamised kasutusjuhud, millest igäüht on koos näidetega põhjalikult kirjeldatud. Lisaks on kirjeldatud ka rakenduse andmestruktuuri ja selle eripärasid. Demonstreerimaks loodud süsteemi vastavust nõuetele ning kasutatavust ja kasutajasõbralikkust keskendub töö lõpuosa rakenduse tulevase kasutaja peal läbi viidud kasutatavuse testile ja selle tulemustele.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 32 leheküljel, 4 peatükki, 22 joonist.

Abstract

Warehouse Management Software For Folk Dance Ensemble

The aim of this thesis is to create a web application for Estonian folk dance ensembles to manage their national clothing warehouses. The application gives a better overview of the warehouse and clothes that have been lent to dancers. The thesis gives overview of the functionality of the application. It lists and describes the use cases for the application and gives detailed descriptions for each of them. The main functionalities of the application are adding new clothes to the system, lending them out to dancers of the ensemble and having dancers return the clothes to the warehouse.

The user interface is made using Angular framework to allow dynamic interface. The server uses Spring framework and REST architectural style. Communicating with the database is done using Spring JPA which allows to use the database by writing Java classes and abstract methods to insert, delete and query data.

In order to validate the results of the thesis a usability test was organized. In the test all the use cases were tested to see how well the user would be able to complete the given tasks. The detailed description and results of the test are described later in the thesis.

The thesis is in Estonian and contains 32 pages of text, 4 chapters, 22 figures.

Lühendite ja mõistete sõnastik

Angular	Raamistik kasutajaliidese loomiseks
kasutaja	Rakenduse kasutaja, laohaldur
rahvariidekomplekt	Komplekt rõivaesemetest, mida kantakse koos.
Spring	Raamistik serveripoolse suhtluse loomiseks
tantsija	Rahvatantsuansambli liige

Sisukord

1 Sissejuhatus	9
2 Taust	10
2.1 Probleemi kirjeldus ja töö eesmärgid	10
2.2 Alternatiivsed lahendused.....	11
3 Veebirakendus	13
3.1 Kasutusjuhud	13
3.1.1 Rühma nime lisamine	13
3.1.2 Uue tantsija sisestamine	14
3.1.3 Rõiva tüübi lisamine.....	15
3.1.4 Uue rõivaeseme sisestamine.....	16
3.1.5 Rõivaeseme aktiivseks või mitteaktiivseks muutmine	16
3.1.6 Rahvariidekomplekti defineerimine	17
3.1.7 Tantsijale rõivaste väljastamine	18
3.1.8 Rõivaste lattu tagastamine	21
3.1.9 Lao seisu vaatamine	21
3.1.10 Rõivaeseme asukoha vaatamine	21
3.1.11 Tantsija käes olevate rõivaste vaatamine	22
3.2 Andmestruktuur	22
3.2.1 Rõivaste struktuur.....	23
3.2.2 Komplektide struktuur.....	24
3.2.3 Rõivaste väljastamise struktuur	24
3.3 Turvalisus	25
3.4 Tulemuste valideerimine	28
3.4.1 Kasutaja saab lisada süsteemi uut tantsijat	28
3.4.2 Kasutaja saab lisada süsteemi uut rõivaeset	28
3.4.3 Kasutaja saab defineerida rõivaesemetest rahvariidekomplekti	28
3.4.4 Kasutaja saab väljastada tantsijale rõivaid	28
3.4.5 Kasutaja saab rõivaesemeid tantsijalt lattu vastu võtta	28
3.4.6 Kasutaja saab vaadata lao seisu	29

3.4.7 Kasutaja saab vaadata rõivaeseme asukohta	29
3.4.8 Kasutaja saab vaadata, mis rõivad tantsija käes on	29
3.5 Kasutajatest.....	29
3.5.1 Rühma nime lisamine	29
3.5.2 Uue tantsija sisestamine	29
3.5.3 Rõiva tüübi lisamine.....	30
3.5.4 Uue rõivaeseme sisestamine.....	30
3.5.5 Rahvariidekomplekti defineerimine	30
3.5.6 Tantsijale rõivaste väljastamine	30
3.5.7 Rõivaste lattu tagastamine.....	30
3.5.8 Lao seisu vaatamine	30
3.5.9 Rõivaeseme asukoha vaatamine	31
3.5.10 Tantsija käes olevate rõivaste vaatamine	31
3.5.11 Kokkuvõte	31
4 Kokkuvõte	32
Kasutatud kirjandus	33

Jooniste loetelu

Joonis 1. Kasutusjuhtude mudel	13
Joonis 2. Rühma lisamine	14
Joonis 3. Tantsija sisestamine.....	15
Joonis 4. Rõiva tüübi lisamine.....	15
Joonis 5. Rõivaeseme sisestamine	16
Joonis 6. Rõivaeseme muutmine	17
Joonis 7. Komplekti lisamine	17
Joonis 8. Komplekti rõiva nimede lisamine	18
Joonis 9. Tantsija otsing	18
Joonis 10. Rõiva otsing numbri järgi.....	18
Joonis 11. Rõiva otsing nime järgi	19
Joonis 12. Rõiva otsing komplekti järgi	19
Joonis 13. Komplekti rõivad.....	20
Joonis 14. Rõivaesemed väljastamisel.....	20
Joonis 15. Rõiva tagastamine	21
Joonis 16. Lao seis.....	21
Joonis 17. Rõivaeseme asukoht.....	21
Joonis 18. Tantsija rõivad.....	22
Joonis 19. Klassiskeem.....	23
Joonis 20. Turvalisuse konfiguratsioon	26
Joonis 21. Autentimine Angulariga	27
Joonis 22. Puudub õigus tantsijat näha.....	27

1 Sissejuhatus

Käesoleva bakalaureusetöö eesmärgiks on luua veebirakendus rahvatantsuansamblitele. Rakenduse eesmärk on lihtsustada ansambli tööd muutes ansambli lao halduse lihtsamaks ja selgemaks. Rakendus ei ole piiratud ühe ansambliga, vaid rakenduse kaudu saavad oma ladu hallata mitmed erinevad ansamblid.

Rahvatantsuansambel koosneb mitmest rühmast, millest üks on reeglina järelkasvu jaoks ning üks kuni kaks põhirühma. Tavaliselt leidub ansamblil ka vilistlasrühm, mis koosneb tantsijatest, kes enam põhirühmas tantsida ei soovi või teatud põhjustel ei saa. See tähendab, et ansambel koosneb vähemalt kahest rühmast, aga enamikul juhtudel on rühmi vähemalt kolm. Näiteks TTÜ Tantsuansambel Kuljus koosneb nooremast rühmast, kahest põhirühmast ja kahest vilistlasrühmast, mis teeb kokku viis rühma [1]. Rahvatantsurühmas on reeglina 8 paari ehk 16 inimest, mis tähendab et kontserdi jaoks tuleb laost välja anda vähemalt 80 komplekti rõivaid, millest iga komplekt koosneb omakorda mitmetest esemetest. Seega on ülevaade laost, tantsijatest ning sellest kus teatud rõivaese asub väga oluline, eriti arvestades seda, et rahvariided on väga kallid ning valmivad käsitööna.

Eesti Rahvatantsuansamblite Liitu kuulub 11 ansamblit [2], kuid lisaks nendele leidub Eesti rahvatantsumaastikul veel ansambleid. Näiteks Tantsuselts Lee [3] ja Hugo Treffneri Gümnaasiumi rahvatantsuansambel [4], millest viimane jaguneb kuueks tantsurühmaks. Seega potentsiaalseid kasutajaid oleks loodaval süsteemil lisaks TTÜ Tantsuansambel Kuljusele kindlasti.

Töö tulemusena valmib toimiv veebirakendus, mis toetab rahvatantsuansamblite tegevust ja aitab omada paremat ülevaadet ansambli rahvarõivastest, nende seisukorrast ning asukohast.

2 Taust

Käesolevas peatükis on kirjeldatud probleemi ning täpsemad eesmärgid, mida tööga saavutatakse.

2.1 Probleemi kirjeldus ja töö eesmärgid

Rahvatantsuansamblid koosnevad mitmetest rühmadest ja selleks, et korraldada kontserte ning esineda, on ansamblitel oma rahvariidelaod. Rahvariidelaos on palju erinevaid esemeid ning nende haldamine vajab eraldi tegelemist. Laos olevaid riideid antakse välja tantsijatele erinevateks esinemisteks ning rahvariided on väga kallid, mistõttu on oluline, et tantsija kellele riided väljastatakse neid ka vastutustundlikult hoiab. Rahvariide väljaandmisel tuleb arvesse võtta hetkelist tantsurühma koosseisu, sest kõigile liikmetele peab leidma sobivad riided ning rühma üldpilt peab olema ühtlane. Kuna koosseisud muutuvad, siis ei ole võimalik ühel tantsijal alati ühte ja sama komplekti kanda ning riideid tuleb sobitada, mis omakorda võtab aega.

Töö eesmärk on luua praktiline ja kasutust leidev süsteem eesti rahvatantsuansamblitele, et kiirendada ja lihtsustada riide jagamist tantsijatele. Ühtlasi peaks loodav süsteem andma parema ja mitmekülgsema ülevaate lao seisust ning tantsijate käes olevatest riietest. Lisaks paremale ülevaatele aitab loodav süsteem ka vältida esemete kaduma minemist. Süsteem ei ole piiratud vaid ühe ansambliga, vaid on kasutatav paljude tantsuansamblite poolt üle Eesti.

Eesmärgiks on realiseerida rakenduses järgnevad funktsionaalsused:

- Kasutaja saab lisada süsteemi uut tantsijat.
- Kasutaja saab lisada süsteemi uut rõivaaset.
- Kasutaja saab defineerida rõivaesemetest rahvariidekomplekti.
- Kasutaja saab väljastada tantsijale rõivaid.

- Kasutaja saab rõivaesemeid tantsijalt lattu vastu võtta.
- Kasutaja saab vaadata lao seisu.
- Kasutaja saab vaadata rõivaeseme asukohta.
- Kasutaja saab vaadata, mis rõivad tantsija käes on.

2.2 Alternatiivsed lahendused

Loodavat rakendust võiks üldistavalt võrrelda esemete rentimise tarkvaraga. Küll aga on olemasolevad saadaval lahendused kas liiga üldised või mõeldud äriliste eesmärkide jaoks, mis antud probleemi lahenduseks ei sobi. Probleem, mida kirjeldatav rakendus lahendab on pigem organisatsiooni sisene ning konkreetse valdkonna vajaduste jaoks mõeldud.

Näiteks olemasolev rakendus Booqable [5] pakub lahendust firmadele, kes tegelevad asjade rendile andmisega. Vaatamata selle rakenduse suurele funktsionaalsusele on tegemist siiski teise eesmärgiga loodud tarkvaraga. Booqable on mõeldud üldiselt esemete rentimiseks ning eseme välja valimine ning reserveerimine on mõeldud tavakasutaja tegevusena. Tavakasutaja vaste töös loodavas süsteemis oleks tantsija, aga rahvatantsuansambelis ei ole rõivad tantsija valida. Lisaks eseme välja valimisele pakub Booqable ka võimalust valida, millal soovitud esemele järgi tulla. Ka sellist vajadust antud süsteemis ei ole, sest riiete väljastamise ja tagastamise ajad määratakse lao poolt.

Turul eksisteerib ka tarkvara nimega 4SIGHT [6], mis võimaldab jälgida ja hallata suurt hulka inventari. Samas on ka selle tarkvara pigem suunatud suurtele ettevõtetele, kes peavad haldama suurtes kogustes esemeid ning kelle tegutsemismuster on teine võrreldes rahvatantsuansambliga. 4SIGHT pakub esemete jälgimist reaalajas juhuks, kui klient eset ei tagasta. Rahvatantsuansambelis aga teab rõivaste väljastaja isiklikult tantsijat, kellele ta rõivaid väljastab. Lisaks sellele ei väljastata rõivaid suvalisele tänavalt tulnud inimesele.

Üldiselt on antud rakenduse funktsionaalsus oluliselt väiksem, aga konkreetse valdkonna vajaduste jaoks mõeldud, mistõttu on väga keeruline leida olemasolevate

lahenduste hulgast mõni rakendus, mis täidaks kõik nõuded ning ei oleks samal ajal liialt keerukas ja liigse funktsionaalsusega.

3 Veebirakendus

Käesolevas peatükis kirjeldatakse rakendust ja selle kasutamise võimalusi.

3.1 Kasutusjuhud

Järgnevalt kirjeldatakse 11 kasutusjuhtu, mis on seotud rakenduse põhifunktsionaalsusega (Joonis 1).



Joonis 1. Kasutusjuhtude mudel

3.1.1 Rühma nime lisamine

Rühma nime lisamine toimub halduse alt, kuhu on koondatud süsteemi kasutamiseks vajalik teematika. Piiranguid rühma nime lisamisel pole. Lähtutud on sellest, et rühma nimel on tähendus ja mõtte kasutaja jaoks. Seega näiteks on võimalik lisada kahte

ühesugust rühma nime. Nime lisamiseks on vaja sisestada soovitud rühma nimi ja vajutada „Lisa rühm“ (Joonis 2).



Rühma nimi	Tantsijate arv	Kustuta
A-rühm	16	
C-rühm	0	

Joonis 2. Rühma lisamine

3.1.2 Uue tantsija sisestamine

Rakenduse jaoks on tantsija üks süsteemi olulisemaid objekte, sest rõivaese saab olla kas laos või väljastatud konkreetsele süsteemi sisestatud tantsijale. Seega kui mõnda rõivaeset pole laos, peab see olema mõne tantsija käes ja nii on võimalik eset õige inimese käest otsida. Tantsija sisestamine toimub „Tantsijad“ alt ning oluline on enne tantsija sisestamist lisada rühma nimi, kuhu tantsija kuulub. Tantsija sisestamisel on eesnimi, perekonnanimi, sugu ja rühm kohustuslikud väljad. Lisaks on võimalik sisestada tantsija pikkus, jalanumber ja pea ümbermõõt, mis ei ole kohustuslikud ning mõeldud selleks, et teatud olukorras võimaldada ainult süsteemis olevate andmete põhjal eseme määramist. Näiteks mõne peakatte puhul saab lähtuda pea ümbermõõdust (Joonis 3).

Eesnimi

Perekonnanimi

Sugu

Rühm

Pikkus

Jala number

Pea ümbermõõt

[Lisa tantsija +](#)

Joonis 3. Tantsija sisestamine

3.1.3 Rõiva tüübi lisamine

Rõiva tüübi lisamine toimub halduse alt, kuhu on koondatud süsteemi kasutamiseks vajalik temaatika. Sarnaselt rõiva nimele puuduvad ka rõiva tüübi lisamisel piirangud, sest tegemist on rõivaste klassifitseerimiseks mõeldud väärtusega. Süsteemis on algselt olemas rõiva tüübid, mida kasutaja võib hakata kasutama või kustutada ning sisestada enda soovitud rõiva tüübid. Rõiva tüübi lisamiseks tuleb sisestada soovitud rõiva tüüp ning vajutada „Lisa rõiva tüüp“ (Joonis 4).

[Lisa rõiva tüüp +](#)

Riide tüüp	Rõivaste arv	Kustuta
PEAKATTED	0	
SÄRGID	0	

Joonis 4. Rõiva tüübi lisamine

3.1.4 Uue rõivaeseme sisestamine

Rõivaese on sarnaselt tantsijaga üks olulisemaid süsteemi objekte, sest rõivaesemele süsteemis peaks vastama reaalne rõivaese laos. Selleks, et lisada rõivaeset, peab süsteemi olema sisestatud rõiva tüübid, mida soovitakse kasutada. Rõivaeseme lisamiseks tuleb kõigepealt valida sobiv rõivatüüp. Seejärel tuleb valida sobiv rõiva nimi (näiteks „Audru seelik“) ning siis on võimalik sisestada konkreetseid rõivaesemeid. Juhul kui soovitud rõiva nime ei ole veel süsteemi sisestatud, on seda võimalik teha vajutades nupule „Sisesta uus rõiva nimi“, mis võimaldab valimise asemel sisestada rõiva nime ning seejärel saab jätkata rõivaesemete sisestamisega. Rõivaeseme sisestamisel on võimalik sisestada eseme number, suurus ja kommentaar, millest ükski pole kohustuslik, sest näiteks kaelarättidel ei mängi suurus ja esemete konkreetne eristamine nii suurt rolli (Joonis 5).

Nr	Eseme number	Suurus	Kommentaar
1	850		
2	851		

Joonis 5. Rõivaeseme sisestamine

3.1.5 Rõivaeseme aktiivseks või mitteaktiivseks muutmine

Rahvariiete kandmise tulemusena rõivad kuluvad ning võib juhtuda, et mõni rõivaese läheb katki ja vajab parandamist. Juhul kui rõivaese on katki või ei soovita mõnda rõivaeset väljastada, on võimalik muuta rõivaese mitteaktiivseks. Vaikimisi on kõik rõivaesemed sisestades aktiivsed. Selleks, et muuta rõivaese mitteaktiivseks, tuleb liikuda rõivaste lehele ning valida soovitud rõiva nimi. Seejärel tuleb soovitud rõivaeseme juures vajutada „Muuda“ ning veerus „Aktiivne“ teha kas linnuke või linnuke ära võtta (Joonis 6).

Aktiivne	Eseme number	Suurus	Kommentaar	Asukoht
<input checked="" type="checkbox"/>	851			Ladu Salvesta Tühista
<input checked="" type="checkbox"/>	850			Aveli Vellerind Muuda

Joonis 6. Rõivaeseme muutmine

3.1.6 Rahvariidekomplekti defineerimine

Komplekt koosneb süsteemis rõiva nimedest ja on mõeldud rõivaste väljastamise kiirendamiseks ja lihtsustamiseks. Kasutaja saab komplekte sisestada ja muuta vastavalt vajadusele ning sarnaselt rühmadele ning rõiva tüüpidele piirangud komplekti nimele puuduvad. Üks rõiva nimi võib kuuluda mitmesse komplekti, mis võimaldab luua ka komplekti variatsioone vastavalt kasutaja vajadusele. Uue komplekti lisamiseks tuleb sisestada soovitud komplekti nimi ning vajutada „Lisa komplekt“ (Joonis 7).

Nr	Komplekti nimi	Rõivaste arv	Kustuta
1	Karja komplekt	6	
2	Audru naise komplekt	4	
3	Audru mehe komplekt	3	

Joonis 7. Komplekti lisamine

Komplekti defineerimiseks ehk komplekti rõiva nimede lisamiseks tuleb vajutada soovitud komplekti nimele, mille peale avaneb komplektivaade. Seal on võimalik otsida rõiva nimesid ning neid komplekti lisada (Joonis 8).

Komplekti kuuluvad rõivad ▲

Nr	Rõiva nimi	Eemalda
1	Audru meeste kuub	<input type="button" value="✕"/>
2	Audru meeste vest	<input type="button" value="✕"/>
3	Audru meeste püksid	<input type="button" value="✕"/>

Otsi rõivaid

Lisatud	Rõiva nimi
---------	------------

Joonis 8. Komplekti rõiva nimede lisamine

3.1.7 Tantsijale rõivaste väljastamine

Rõivaste väljastamine on üks süsteemi põhilisi kasutusjuhtusid ja see on tantsija põhine. Et muuta rõivaste väljastamine võimalikult kiireks on rakenduse avalehel peale sisse logimist tantsija otsingu võimalus (Joonis 9).

Rõivaste väljastamiseks otsi tantsijat.

Otsi tantsijat

Joonis 9. Tantsija otsing

Rõivaste väljastamiseks tuleb otsida rõivaid ja seda on võimalik teha kolmel viisil. Kui laos on rõivad nummerdatud, on võimalik ja mõistlik otsida rõivast numbriga järgi, sest sisestades õige numbriga, leiab süsteem kiiresti õige rõiva üles (Joonis 10).

Rõivaste väljastamine

Rõivaeseme number Rõiva nimi Komplekti nimi

Audru meeste kuub 858

Joonis 10. Rõiva otsing numbriga järgi

Juhul kui soovitud rõivaesemetel numbrit määratud pole, on võimalik otsida esemeid ka rõiva nime järgi. Sellisel juhul kuvatakse otsingu tulemusena kõiki selle nimega rõivaesemeid (Joonis 11).

Rõivaste väljastamine

Rõivaeseme number Rõiva nimi Komplekti nimi

kirju|

Kirju kaelarätt [Lisa](#)

Kirju kaelarätt [Lisa](#)

Kirju kaelarätt [Lisa](#)

Joonis 11. Rõiva otsing nime järgi

Kui soovitakse väljastada korraga suuremat hulka rõivaid, siis selle kiirendamiseks on võimalik otsida rõivaid ka kasutaja poolt defineeritud komplektide järgi (Joonis 12).

Rõivaste väljastamine

Rõivaeseme number Rõiva nimi Komplekti nimi

audru|

Audru mehe komplekt

Audru naise komplekt

Joonis 12. Rõiva otsing komplekti järgi

Kui vajutada komplekti nimele, kuvatakse komplekti juures defineeritud rõiva nimed ning vaikimisi esimene laos oleva eseme number, mida on võimalik omakorda muuta vastavalt sellele, millist eset realselt tantsijale väljastatakse (Joonis 13).

Rõivaste väljastamine

Rõivaeseme number Rõiva nimi Komplekti nimi

audru

Audru seelik 850

Audru kampsun (must) 852

Audru pluus 855

Audru tanu 856

Lisa rõivad

Joonis 13. Komplekti rõivad

Vajutades rõivaeseme juures nupule „Lisa“, lisatakse vastav rõivaese väljastamisele kuuluvate rõivaesemete hulka. Komplekti puhul lisatakse „Lisa rõivad“ nupule vajutades kõik komplekti kuuluvad rõivaesemed, mis on näha komplekti rõivaste vaates. Väljastamisele kuuluvate rõivaste vaates on võimalik veel valida, milliseid rõivaid kasutaja siiski väljastada soovib. Kui kasutaja on veendunud, et soovib valitud rõivaid väljastada, saab ta seda teha vajutades nupule „Väljasta esemed“ (Joonis 14).

Rõivad väljastamisel

	Rõiva nimi	Eseme nr
<input checked="" type="checkbox"/>	Audru seelik	850
<input checked="" type="checkbox"/>	Audru kampsun (must)	852
<input checked="" type="checkbox"/>	Audru pluus	855
<input checked="" type="checkbox"/>	Audru tanu	856

Väljasta esemed

Joonis 14. Rõivaesemed väljastamisel

3.1.8 Rõivaste lattu tagastamine

Rõivaste lattu tagastamine on rõivaste väljastamise kõrval teine oluline kasutusjuht ning ka see on tantsija põhine. See tähendab seda, et rõiva lattu tagastamiseks tuleb süsteemist otsida üles tantsija, kellele rõivas esialgu väljastatud on. Tegevuse kiirendamiseks saab kasutada avalehel tantsija otsingut. Tantsija lehel kuvatakse kõiki tantsija käes olevaid rõivaid koos kuupäevaga, mil need väljastati. Iga eseme järel on nupp „Tagasta“, millele vajutades tagastatakse rõivaese lattu (Joonis 15).

Karja seelik	209	5. 4. 2017	Tagasta
--------------	-----	------------	---------

Joonis 15. Rõiva tagastamine

3.1.9 Lao seisu vaatamine

Lao halduri jaoks on oluline omada ülevaadet üldisest lao seisust ja seda on võimalik näha rõivaste alt. Seal on näha loetelu rõiva nimedest ning iga rõiva nime kohta kuvatakse arvud laos olevate, väljastatud ja mitteaktiivsete rõivaesemete kohta. Samuti kuvatakse iga rõiva nimega seotud esemete koguarvu (Joonis 16).

Nr	Rõiva nimi	Laos	Väljastatud	Mitteaktiivsed	Kokku
1	Karja seelik	6	2	0	8

Joonis 16. Lao seis

3.1.10 Rõivaeseme asukoha vaatamine

Olukorras, kus on vaja teatud esinemiseks väljastada rõivaid, kuid neid ei jätku või mõni ese on puudu, on võimalik süsteemist vaadata, kelle käes rõivaese on. Seda saab teha rõivaste alt, vajutades soovitud rõiva nime peale. Avaneb nimekiri antud rõiva nimega seotud rõivaesemetest ning asukoha all kuvatakse konkreetse tantsija nimi või lihtsalt „Ladu“ (Joonis 17).

Aktiivne	Eseme number	Suurus	Kommentaari	Asukoht
<input checked="" type="checkbox"/>	851			Ladu <input type="button" value="Muuda"/> <input type="button" value="Kustuta"/>
<input checked="" type="checkbox"/>	850			Aveli Vellerind <input type="button" value="Muuda"/> <input type="button" value="Kustuta"/>

Joonis 17. Rõivaeseme asukoht

3.1.11 Tantsija käes olevate rõivaste vaatamine

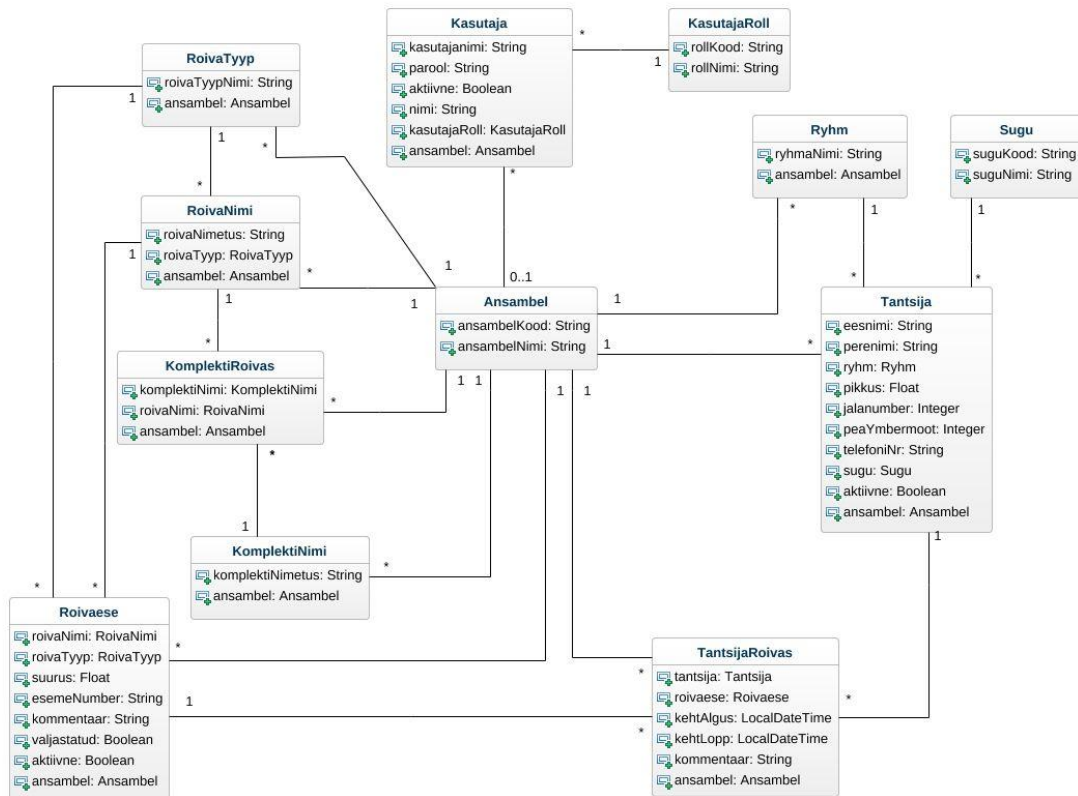
Vastavalt vajadusele võib olla vajalik näha ka tantsija käes olevaid rõivaid. Peamiselt kasutatakse seda vaadet rõivaste tagastamisel. Tantsija käes olevaid rõivaid saab näha konkreetse tantsija lehel. Kasutajale kuvatakse rõiva nime, rõivaeseme numbrit ja rõiva väljastamise kuupäeva (Joonis 18).

Rõiva nimi	Eseme nr	Välja antud	
Audru seelik	850	19. 4. 2017	Tagasta
Audru kampsun (must)	852	19. 4. 2017	Tagasta
Audru pluus	855	19. 4. 2017	Tagasta
Audru tanu	856	19. 4. 2017	Tagasta

Joonis 18. Tantsija rõivad

3.2 Andmestruktuur

Loodav rakendus on mõeldud korraga kasutamiseks mitmele tantsuansamblile, mistõttu on oluline, et eri ansamblite tantsijad, rõivad ja muud süsteemiobjektid omavahel segamini ei läheks. Lisaks segaduse vältimisele on tegemist ka turvalisuse probleemiga. Kui keegi saaks kasvõi kogemata muuta võõra ansambliga seotud andmeid, oleks tegemist küberturvalisuse seisukohast tervikluse printsiibi rikkumisega. Selle tagamiseks on kõik süsteemiobjektid peale objektide KasutajaRoll ja Sugu seotud objektiga Ansambel, mida saab vastavalt vajadusele juurde luua.



Joonis 19. Klassiskeem

3.2.1 Rõivaste struktuur

Rõivad koosnevad struktuuri poolest kolmest osast, mis aitavad süsteemi paremini hallata ja mis peaks muutma süsteemi ka kasutajale selgemaks. Kõige üldisem rõivastruktuuri osa on rõiva tüüp, mis klassiskeemil on nimega RoivaTyp (Joonis 19). Rõiva tüüp on mõeldud rõivaste eristamiseks, et muuta kasutajakogemus dünaamilisemaks ning pakkuda kasutajale paremat ülevaadet. Rõiva tüübid on kasutaja jaoks juba eelnevalt sisestatud ning kohe kasutatavad. Juhul kui kasutaja ei soovi olemasolevaid rõiva tüüpe kasutada, on tal võimalik vastavalt vajadusele tüüpe lisada või kõik olemasolevad tüübid kustutada ja lisada täiesti uued rõiva tüübid vastavalt oma soovile.

Eialgu on süsteemis olemas järgmised rõiva tüübid, mis on valitud koostöös hetkel ansambli lao juures tegutseva inimesega.

- PEAKATTED
- SÄRGID

- PÜKSID
- VESTID JA PEALISRÕIVAD
- SEELIKUD
- PÕLLED
- JALANÕUD
- VÖÖD
- ALUSRÕIVAD
- SUKAD
- SUKAPAEELAD
- EHTED JA AKSESSUAARID
- MUU

Järgmiseks tasemeks on rõiva nimi, mis klassiskeemil on RõivaNimi (Joonis 19). Rõiva nimed on täielikult kasutaja sisestada, sest need sõltuvad otseselt reaalsest laost. Näiteks võib tuua seeliku, mille nimeks on „Audru seelik“. Antud juhul on selline rõiva nimi seotud rõiva tüübiga „SEELIK“ (Joonis 5).

Kolmandaks tasemeks on rõivaese, mis klassiskeemil on Rõivaese (Joonis 19). Igale rõivaesemele vastab laos üks reaalne rõivaese. Rõivaese viitab andmebaasis korraga ühele rõiva nimele.

3.2.2 Komplektide struktuur

Komplektide struktuur koosneb kahest osast. Objekt KomplektiNimi (Joonis 19) kirjeldab komplekti nime, millega võib olla seotud piiramatu arv rõiva nimesid. Piirangut seoste arvule ei ole.

3.2.3 Rõivaste väljastamise struktuur

Rõivaste väljastamise jaoks on süsteemis TantsijaRoivas, mis väljendab konkreetse tantsija ja konkreetse rõivaeseme vahelist seost. Kui kasutaja väljastab rõivaeseme,

luuakse kirje tabelisse TantsijaRoivas kehtivuse algusajaga, mis kajastab kasutaja jaoks rõivaeseme väljastamise aega. Kehtivuse lõppaeg jäetakse tühjaks. Selleks, et ühte rõivaeset ei saaks samal ajal mitmele tantsijale väljastada, on Rõivaeseme küljes väli „väljastatud“, mis kajastab seda, kas rõivaese on laost antud hetkel väljastatud või ei. Rõivaeseme lattu tagastamisel täidetakse objekti TantsijaRoivas kehtivuse lõppaeg, mis kasutaja jaoks on rõivaeseme lattu tagastamise aeg. Kui ülejäänud süsteemiobjekte on võimalik andmebaasist reaalsetl kustutada, siis objektid TantsijaRoivas jäävad andmebaasi alles. See omakorda võimaldab näha ajalugu tantsija või rõivaeseme lõikes. On võimalik näha, mis konkreetsed rõivaesemed on antud tantsija käes olnud või milliste tantsijate käes antud rõivaese mingitel ajahetkedel olnud on.

3.3 Turvalisus

Rakenduse serveri osa on tehtud kasutades Spring raamistikku [7]. Turvalisuse jaoks on kasutatud Spring security [8] võimalusi. Spring security võimaldab määrata, mis leheküljed on kõigile ligipääsetavad ning millised lehed vajavad autentimist. Antud rakenduses on kõigile ligipääsetavad registreerimise ning sisselogimise leheküljed. Kõigi teiste lehekülgede jaoks on vaja kasutaja autentimist. Lisaks lehekülgede kuvamise lubamisele on määratud ka aadressid ehk päringud, mida server lubama peaks. Aadressid „home.html“ ja „/“ on lubatud sellepärast, et tegemist on vaikimisi avalehega, millelt kasutaja suunatakse automaatselt edasi lehele „login.html“ juhul kui autentimist pole toimunud. Ridadel 35-40 on ära toodud aadressid ning rida 41 tähendab seda, et need aadressid on lubatud (Joonis 20). Rida 42 tähendab seda, et kõik ülejäänud aadressid vajavad autentimist.

```

28      @Override
29      protected void configure(HttpSecurity http) throws Exception {
30          http
31              .httpBasic()
32              .and()
33              .authorizeRequests()
34              .antMatchers(
35                  ...antPatterns: "/index.html",
36                  "/",
37                  "/views/home.html",
38                  "/views/login.html",
39                  "/views/registreeri.html",
40                  "/registreeri")
41              .permitAll()
42              .anyRequest().authenticated()
43              .and()
44              .formLogin().loginPage("/login")
45              .usernameParameter("username").passwordParameter("password")
46              .and()
47              .csrf()
48              .csrfTokenRepository(CookieCsrfTokenRepository.withHttpOnlyFalse());
49      }

```

Joonis 20. Turvalisuse konfiguratsioon

Rakenduse kasutajaliidese osa on tehtud kasutades Angulari raamistikku [9], mis võimaldab luua võimalikult lihtsalt dünaamilist kasutajaliidest. Spring hoolitseb küll selle eest, kas päringutele on lubatud serveri poolt vastata, aga ka Angulari poolt on vajalik teatud määral kasutaja autentimine. Täpsemalt teevad Spring ja Angular avalehel koostööd, et juhul kui kasutaja pole ennast autentinud, ei kuvaks rakendus kasutajale mittelubatud sisu. Selleks, et Angulari ja Springi omavahel tööle panna, kasutasin ma juhendit Springi kodulehelt [10]. Ridadel 124126 luuakse objekt, kus on kirjas kasutaja autentimiseks vajalikud andmed (Joonis 21). Real 128 tehakse päring serverile ja antakse kaasa loodud objekt. Kui kasutaja autentimine õnnestub, siis real 130 antakse Angulari globaalsele muutujale vastav väärtus, et kuvada kasutajale lubatud sisu. Kui autentimine ebaõnnestub, siis saab real 133 sama globaalne muutuja vastupidise väärtuse ja kasutajale kuvatakse veateadet.

```

121 .controller('navigationController', function ($rootScope, $http, $location) {
122     var vm = this;
123     var authenticate = function (credentials, callback) {
124         var headers = credentials ? {
125             authorization: "Basic " + btoa(credentials.username + ":" + credentials.password)
126         } : {};
127
128         $http.get('/user', {headers: headers}).then(function (response) {
129             if (response.data.authenticated) {
130                 $rootScope.authenticated = true;
131                 getKasutajaAndmed();
132             } else {
133                 $rootScope.authenticated = false;
134             }
135             callback && callback();
136         }, function () {
137             $rootScope.authenticated = false;
138             callback && callback();
139         });
140     };
141
142     authenticate();
143     vm.credentials = {};
144     vm.login = function () {
145         authenticate(vm.credentials, function () {
146             if ($rootScope.authenticated) {
147                 $location.path("/");
148                 vm.error = false;
149             } else {
150                 $location.path("/login");
151                 vm.error = true;
152             }
153         });
154     };

```

Joonis 21. Autentimine Angulariga

Enamus rakendusest kasutab serveriga suhtlemiseks sisemisi päringuid ehk kui päritakse andmeid näiteks id järgi, siis see id ei kajastu veebilehitseja aadressireal. Selline lahendus hoolitseb lihtsal tasandil selle eest, et ei oleks võimalik kogemata või sihilikult lehekülje aadressi muutes ligi pääseda mõne muu ansambli andmetele. Erandiks antud rakenduses on tantsija lehekülg, mille aadressis sisaldub tantsija id. Et vältida sellisel juhul võõra ansambli tantsija info nägemist, toimub serveris kontroll, kas tantsija kelle andmeid päritakse on kasutajaga samast ansamblist. Juhul kui tantsija on erinevast ansamblit, kuvatakse vastavat veateadet (Joonis 22).



Joonis 22. Puudub õigus tantsijat näha

3.4 Tulemuste valideerimine

Antud töö alguses sai ära toodud 8 eesmärki, mille alusel valideerida tulemuste sobivust.

3.4.1 Kasutaja saab lisada süsteemi uut tantsijat

Tegemist on olulise nõudega, sest rõivaste väljastamine toimub tantsija põhiselt. Antud nõue sai täidetud edukalt, sest tantsija lisamise protsess ei ole liialt keerukas ning süsteem annab tagasisidet selle kohta, kas tantsija lisamine õnnestus või mitte.

3.4.2 Kasutaja saab lisada süsteemi uut rõivaeset

Tegemist on olulise nõudega, sest rõivaste väljastamiseks peab leiduma süsteemis rõivaid, mida väljastada. Rõivaeseme lisamine võib süsteemi esmasel kasutamisel tunduda keerukas kui kasutaja ei ole veel süsteemiga tuttav. Seda sellepärast, et rõivaeseme lisamine hõlmab endas ka rõiva nime lisamist, juhul kui soovitud rõiva nime veel süsteemis ei eksisteeri. Kasutajatega testides oli näha, et rõiva nime lisamine esimesel korral tekitas kasutajates segadust, kuid veidi proovides ja katsetades, said nad rõiva nime lisamisega hakkama. Seega, vaatamata keerukale funktsionaalsusele, sai nõue täidetud.

3.4.3 Kasutaja saab defineerida rõivaesemetest rahvariidekomplekti

Tegemist on nõudega, mis on mõeldud rõivaste väljastamise lihtsustamiseks. Seega on tegemist nõudega, mis ei takista süsteemi põhifunktsionaalsuse toimimist. Nõue sai täidetud edukalt.

3.4.4 Kasutaja saab väljastada tantsijale rõivaid

Tegemist on süsteemi põhifunktsionaalsust kirjeldava nõudega. Rõivaste väljastamiseks vajalikku otsingut on võimalik teostada eseme numbri, rõiva nime või defineeritud komplekti nime järgi. See muudab rõivaste väljastamise dünaamilisemaks ning vastavalt vajadusele efektiivsemaks. Seetõttu võib öelda, et nõue sai edukalt täidetud.

3.4.5 Kasutaja saab rõivaesemeid tantsijalt lattu vastu võtta

Tegemist on süsteemi põhifunktsionaalsust kirjeldava nõudega. Rõivaste lattu tagastamine on teine põhiline funktsionaalsus rõivaste väljastamise kõrval. Rõivaste tagastamine toimub lihtsalt ning nõue sai täidetud edukalt.

3.4.6 Kasutaja saab vaadata lao seisu

Tegemist on laost ülevaadet pakkuva nõudega. Kasutaja saab ülevaate laos olevate ja väljastatud rõivaesemete arvude kohta. Seega on olemas ülevaade lao seisu kohta ning nõue on täidetud.

3.4.7 Kasutaja saab vaadata rõivaeseme asukohta

Tegemist on nõudega, mis on mõeldud kadunud rõivaesemete asukoha väljaselgitamiseks. Valides süsteemis rõiva, on iga rõivaeseme juures kirjas kas ese on mõne tantsija käes või laos. Juhul kui rõivaese on tantsija käes, on ära toodud konkreetse tantsija nimi. Seetõttu on nõue täidetud.

3.4.8 Kasutaja saab vaadata, mis rõivad tantsija käes on

Tegemist on nõudega, mis võimaldab saada infot tantsija käes olevate rõivaesemete kohta. Selle info alusel on võimalik veenduda, et konkreetsele tantsijale on juba mõni rõivaese väljastatud ning selle funktsionaalsuse kaudu toimub ka rõivaesemete lattu tagastamine, sest tagastamine on tantsija põhine tegevus ning tagastada saab ainult esemeid, mis on laost väljastatud. Seega on nõue edukalt täidetud.

3.5 Kasutajatest

Viisin Tallinna Tehnikaülikooli tantsuansambel Kuljuse laohalduriga lõpptulemuse valideerimiseks läbi kasutajatesti, mille käigus testisin töös eelpool kirjeldatud kasutusjuhtusid.

3.5.1 Rühma nime lisamine

Ülesandeks oli lisada süsteemi kaks uut rühma nime, mis olid ette antud. Kasutajal kulus veidi aega, et end menüüga kurssi viia ja pärast seda suutis kasutaja kiiresti vajaliku rühma nime lisada.

3.5.2 Uue tantsija sisestamine

Ülesandeks oli sisestada süsteemi kaks uut tantsijat, kelle nimed olid kasutajale ette antud. Esialgu hakkas kasutaja tantsijate lehel olevatesse otsingulahtritesse uue tantsija infot sisetama, kuid märkas kiiresti oma viga ning leidis üles tantsija lisamise lehe.

3.5.3 Rõiva tüübi lisamine

Ülesandeks oli lisada süsteemi üks uus rõiva tüüp. Kasutajal oli alguses segadus terminitega ning ta üritas rõiva tüüpi lisada rõivaeseme lisamise lehel. Kasutaja sai siiski aru, mida vaja teha on ning leidis rakendusest õige koha üles.

3.5.4 Uue rõivaeseme sisestamine

Ülesandeks oli sisestada süsteemi 12 erinevat rõivaeset 6 erineva rõiva nimega, mis olid kasutajale ette antud. Uue rõiva nime lisamisega, mis on kasutajaliidese koha pealt pigem keeruline, sai kasutaja kiiresti hakkama ning sellega probleeme ei tekkinud. Ka ei tekkinud probleeme ega raskusi rõivaesemete sisestamisega.

3.5.5 Rahvariidekomplekti defineerimine

Ülesandeks oli sisestatud rõivaesemetest defineerida rahvariidekomplekt. Kasutaja leidis kiiresti üles õige koha süsteemis ning komplekti loomine ning defineerimine ei tekitanud raskusi. Kasutaja kommenteeris, et tegemist oli väga lihtsa ja loogilise protsessiga.

3.5.6 Tantsijale rõivaste väljastamine

Ülesandeks oli väljastada tantsijale teatud numbritega rõivaesemed. Tantsija ning rõivaeseme numbrid olid kasutajale ette antud. Süsteemist tantsija leidmisega kasutajal probleeme ei olnud. Vajalike rõivaesemete leidmiseks kasutas kasutaja otsingut eseme numbri järgi. Põhjenduseks tõi kasutaja välja selle, et see variant tundus kõige kiirem, sest talle olid teada esemete numbrid.

3.5.7 Rõivaste lattu tagastamine

Eelnevas ülesandes tuli kasutajal tantsijale väljastada kaks pluusi. Selles ülesandes pidi kasutaja lattu tagastama üleliigse pluusi. Seega pidi kasutaja leidma üles tantsija rõivad ja nende hulgast kaks pluusi. Kasutaja sai ülesandega kiirelt hakkama.

3.5.8 Lao seisu vaatamine

Ülesandeks oli vaadata lao seisu ehk kui palju mis rõivaesemeid on laos ja kui palju väljastatud. Esialgu kulus kasutajal veidi aega, et leida rakendusest üles õige koht. Kui see oli leitud, suutis kasutaja vajaliku info välja lugeda.

3.5.9 Rõivaeseme asukoha vaatamine

Ülesandeks oli vaadata, kelle käes on rõivaesemed, mille nimi oli kasutajale ette antud. Kasutaja leidis õige koha üles ning rõiva nimede hulgast ka vajaliku rõiva nime. Kasutaja sai aru, et vajalik on rõiva nime peale vajutada ning sealt suutis kasutaja edukalt välja lugeda, kelle käes antud rõivaesemed on.

3.5.10 Tantsija käes olevate rõivaste vaatamine

Ülesandeks oli vaadata, mis riided on ette antud tantsija käes. Kasutaja leidis tantsija kiiresti ning tantsija vaates sai kasutaja ilma raskusteta aru, mis riided tantsija käes on.

3.5.11 Kokkuvõte

Kasutaja sõnul kasutab ta praegu lao haldamiseks tabelarvutusprogrammi. Antud rakendus aga pakub kasutaja sõnul paremat ülevaadet ning on mugavam kasutada kui tabelarvutusprogramm. Kokkuvõttes jäi kasutaja rakendusega rahule ja hinnang rakendusele oli positiivne.

4 Kokkuvõte

Bakalaureusetöö eesmärgiks oli luua veebirakendus eesti rahvatantsuansamblitele nende rahvariideladude haldamiseks. Töö käigus valmis rakendus, mida saavad kasutada korraga mitmed eesti rahvatantsuansamblid ning mis lihtsustab oluliselt ansambli rahvarõivaste haldamist.

Rakendus võimaldab sisestada süsteemi rõivaid struktuurselt ning organiseeritult ja annab kasutajale võimaluse ise struktuuri kohandada vastavalt ansambli vajadustele. Lisaks rõivaste sisestamisele võimaldab rakendus lisada süsteemi ka tantsijaid ning neid rühmadesse jaotada. Rakenduse peamiseks funktsionaalsuseks on rõivaste väljastamine tantsijatele ning nende lattu tagastamine. Tänu nendele toimingutele on süsteemis üheselt arusaadavalt näha, millised rõivad on millise tantsija käes ning seeläbi väheneb võimalus, et rõivad jäävad kogemata kellegi kätte seisma. Ning isegi kui rõivad jäävad pikemaks ajaks kellegi kätte, siis on võimalik kerge vaevaga välja selgitada, kelle käest konkreetseid rõivaid otsima peaks.

Realiseeritud sai kogu funktsionaalsus, mis töö alguses kirja pandud sai. Lisaks sellele viisin süsteemi ühe tulevase kasutajaga Tallinna Tehnikaülikooli tantsuansambel Kuljusest läbi kasutajatesti, et valideerida loodud süsteemi vastavust reaalsele vajadustele. Kasutaja oli rakendusega väga rahul ning eelistab seda hetkel kasutatava süsteemi asemel. Loodud funktsionaalsuse ja kasutaja rahulolu põhjal võib öelda, et püstitatud eesmärgid said täidetud.

Kasutatud kirjandus

- [1] Tallinna Tehnikaülikooli Tantsuansambel Kuljus, [Online]. Available: http://www.kuljus.ee/index.php?option=com_content&view=article&id=12&Itemid=28 [Accessed 18 April 2017]
- [2] Eesti Rahvatantsuansamblite Liit, [Online]. Available: <https://www.ertal.ee/ansamblid/> [Accessed 18 April 2017]
- [3] Tantsuselts Lee, [Online]. Available: <http://lee.ee/> [Accessed 18 April 2017]
- [4] Hugo Treffneri Gümnaasiumi tantsuansambel Tantsutallad, [Online]. Available: <http://tantsutallad.ee/index.php> [Accessed 18 April 2017]
- [5] Booqable, [Online]. Available <https://booqable.com/?source=Google%20Ads&campaign=Main&gclid=CN6nsaL9wtMCFUdfGQodQo4CvQ> [Accessed 26 April]
- [6] 4SIGHT, [Online]. Available <https://4sightsolution.com/warehouse-asset-management/> [Accessed 17 May 2017]
- [7] Spring, [Online]. Available <https://spring.io/docs> [Accessed 25 April 2017]
- [8] Spring security, [Online]. Available: <https://docs.spring.io/spring-security/site/docs/current/reference/htmlsingle/> [Accessed 25 April 2017]
- [9] Angular, [Online]. Available <https://angularjs.org/> [Accessed 25 April 2017]
- [10] Spring Security and Angular JS, [Online]. Available <https://spring.io/guides/tutorials/spring-security-and-angular-js/> [Accessed 11 May]