

TALINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Informaatika instituut

Infosüsteemide õppetool

# **Toitlustusettevõtete broneerimissüsteemi lahendus**

Bakalaureusetöö

Üliõpilane: Mihkel Auväärt

Üliõpilaskood: 104437IABB

Juhendaja: lektor Raul Liivrand

Tallinn  
2014

## **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: Mihkel Auväärt

.....  
(*allkiri*)

28.05.2014

## **Annotatsioon**

Käesoleva bakalaureusetöö eesmärk on pakkuda välja mugav ning kaasaegne broneerimissüsteemi lahendus toitlustusettevõtetele (peamiselt restoranid, sööklad, pubid, baarid jne). Lahendusega püütakse saavutada mugavam laua broneerimise protsess nii toitlustusettevõtte töötajatele kui ka küllastajatele.

Olulisemad töös käsitletavad probleemid:

- milline on hetkel üldine broneerimise protsess toitlustusettevõttes;
- kuidas saaks parendada protsessi, kasutades infotehnoloogilisi lahendusi;
- milline oleks parendatud protsessi rakendav infosüsteem.

Töö olulisemad tulemused:

- mugavama ja kaasaegsema broneerimisprotsessi lahendus;
- kaasaegsemat broneerimisprotsessi realiseeriva keskkonna analüüs.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 54 leheküljel, 5 peatükki, 16 joonist, 13 tabelit ja üks lisa.

## **Abstract**

The purpose of this thesis is to provide a better and modern reservation system solution for catering companies. Solution aims to achieve a handy table reservation process for customers and convenient reservations management tools for catering staff.

This thesis deals with the following problems:

- what is the current reservation process;
- how could be process improved by using information technology;
- how to implement the improved process into a web application.

Main results:

- More convenient reservation solution for catering companies.
- Analysis of the system that implements suggested business process.

The thesis is in Estonian and contains 54 pages of text, 5 chapters, 16 figures, 13 tables and one appendix.

## Lühendite ja mõistete sõnastik

<b>AJAX</b>	<b><i>Asynchronous JavaScript and XML</i></b> „Asünkroonne JavaScript ja XML.“ [1]
<b>Bootstrap</b>	<b><i>Bootstrap</i></b> Populaarne ja skaleeruv raamistik veebirakenduste kiireks arendamiseks. [2]
<b>BPMN</b>	<b><i>Business Process Model and Notation</i></b> Standardne äriprotsesside ja –mudelite modelleerimise notatsioon, mis lihtsustab ettevõtete äriprotsesside graafilist esitamist. [3]
<b>CAPTCHA</b>	<b><i>Completely Automated Public Turing test to tell Computers and Humans Apart</i></b> „Robotilõks. Veebivormidele lisatav Turingi test, mis peab tagama selle, et antud vormi saab täita ja ära saata ainult inimene, mitte robot.“ [1]
<b>CSS</b>	<b><i>Cascading Style Sheets</i></b> „Kaskaadlaadistik. Veebilehtede valmistajatele ja kasutajatele mõeldud laadistik. Laadilehed ( <i>style sheets</i> ) kirjeldavad, kuidas HTML dokumente esitada kuvaril, printeril või kõnesüntesaatorist kostva kõnena.“ [1]
<b>ERP</b>	<b><i>Enterprise Resource Planning</i></b> Ettevõtte ressursside planeerimine. Lai tegevuste ring, kus kasutatakse erinevaid rakendusprogramme ja mis aitab tootmis- või muul ettevõttel hallata olulisi ärikomponente. [1]
<b>Git</b>	<b><i>Git</i></b> Vabavaraline hajutatud versioonihaldustarkvara. [4]
<b>HTML</b>	<b><i>Hypertext Markup Language</i></b> „Hüpertext-märgistuskeel. Enimlevinud kodeerimissüsteem

(tekstivorming) veebidokumentide loomiseks.“ [1]

**Javascript**

***JavaScript***

„Netscape’i poolt välja töötatud skriptikeel, mis võimaldab veebiautoritel luua interaktiivseid veebisaite.“ [1]

**jQuery**

***jQuery***

Kiire, väike ning võimalusterohke JavaScripti teek. [5]

**jQuery UI**

***jQuery User interface***

Kogum kasutajaliideste ehitamiseks mõeldud funktsioone, mis on ehitatud jQuery teegi põhjal. [6]

**JSON**

***JavaScript Object Notation***

Kerge ja väikesemahuline andmeedastusformaad. [7]

**LAMP**

***Linux, Apache, MySQL, PHP***

Vabavaraline tarkvarakomplekt Linux kernelil põhinevale operatsioonisüsteemile, mis koosneb Apache HTTP Serverist, MySQL andmebaasisüsteemist ja PHPst. [8]

**PHP**

***PHP: Hypertext Preprocessor***

PHP on platvormist sõltumatu skriptikeel, kus veebiserver pöördub PHP poole, et transleerida ja täita PHP skriptiga ette nähtud instruktsioone. [1]

**PHPUnit**

***PHPUnit***

Programmeerijale orienteeritud PHP testimise raamistik. [9]

**REST**

***Representational state transfer***

Tarkvaraarhitektuuri stiil, mis on mõeldud jagatud hüpermeedia süsteemide jaoks. [10]

**Scrum**

***Scrum***

Agiilne, iteratiivne ja inkrementaalne tarkvaraarenduse raamistik. [11]

**SQL**

***Structured Query Language***

„Struktuurpäringukeel. Enimkasutatav päringukeel, mida toetavad

kõik klient-server keskkonnale projekteeritud  
relatsioonandmebaasid.“ [1]

## **UML**

### ***Unified Modeling Language***

Universaalne modelleerimiskeel. Üheainsale metamudelile toetuv graafiliste tähistuste pere, mis aitab kirjeldada ja projekteerida tarkvarasüsteeme. [12]

## **WSDL**

### ***Web Services Description Language***

„Protokoll veebiteenuste kirjeldamiseks ja neile teenustele juurdepääsu võimaldamiseks.“ [1]

## **XML**

### ***The Extensible Markup Language***

„Laiendatav märgistuskeel. Suvaliste andmete struktureerimiseks mõeldud märgistuskeel.“ [1]

## **XML-RPC**

### ***The Extensible Markup Language - Remote Procedure Call***

„XML-kaugprotseduurikutse. Komplekt XML-põhiseid teostusi, mis võimaldavad üle interneti toimivaid platvormist ja programmeerimiskeelest sõltumatuid protseduurikutseid.“ [1]

## Jooniste nimekiri

Joonis 1. Praeguse broneerimise äriprotsessi kirjeldus .....	15
Joonis 2. Lahendusena välja pakutav broneerimise äriprotsess .....	18
Joonis 3. Broneeringu tühistamise äriprotsess.....	19
Joonis 4. Üldine kasutusjuhtude diagramm.....	23
Joonis 5. Külastaja kasutusjuhtude diagramm.....	24
Joonis 6. Töötaja kasutusjuhtude diagramm.....	26
Joonis 7. Juhataja kasutusjuhtude diagramm.....	27
Joonis 8. Administraatori kasutusjuhtude diagramm .....	29
Joonis 9. Keskkonna kontseptuaalmudel.....	31
Joonis 10. Keskkonna arhitektuuri diagramm .....	33
Joonis 11. Keskkonna andmemudel koos atribuutide ja nende tüüpidega .....	34
Joonis 12. Ekraanitõmmis prototüübist, kus saab hallata laudade asukohti ja broneeringuid. 43	
Joonis 13. Valitud laua broneeringute vaatamine prototüübis. ....	44
Joonis 14. Keskkonna prototüübi avaleht.....	53
Joonis 15. Keskkonna prototüübi ettevõtte informatsiooni vaade.....	54
Joonis 16. Keskkonna prototüübi ettevõtte toidumenüü vaade .....	54



## **Tabelite nimekiri**

Tabel 1. Olemasoleva protsessi puudused ja mõõdikud.....	16
Tabel 2. Kasutajate tabeli detailne kirjeldus.....	36
Tabel 3. Kasutajatüüpide tabeli detailne kirjeldus .....	37
Tabel 4. Töötaja tabeli detailne kirjeldus .....	37
Tabel 5. Töötajatüüpide tabeli detailne kirjeldus .....	37
Tabel 6. Toitlustusettevõtte tabeli detailne kirjeldus.....	38
Tabel 7. Toitlustusettevõtete tüüpide tabeli detailne kirjeldus .....	39
Tabel 8. Ruumi tabeli detailne kirjeldus.....	39
Tabel 9. Ruumitüübi tabeli detailne kirjeldus .....	40
Tabel 10. Laudade tabeli detailne kirjeldus.....	40
Tabel 11. Lauatüüpide tabeli detailne kirjeldus.....	41
Tabel 12. Broneeringute tabeli detailne kirjeldus.....	41
Tabel 13. Staatuste tabeli detailne kirjeldus .....	42

## Sisukord

1. Sissejuhatus .....	12
1.1 Taust ja probleem .....	12
1.2 Ülesande püstitus .....	12
1.3 Metoodika .....	13
1.4 Ülevaade tööst .....	13
2. Broneerimise äriprotsess .....	14
2.1 Olemasolev protsess .....	14
2.1.1 Kirjeldus .....	14
2.1.2 Tugevused ja puudused .....	15
2.2 Väljapakutav lahendus .....	17
2.2.1 Broneerimise põhiprotsess .....	17
2.2.2 Broneeringu tühistamise protsess .....	18
2.2.3 Ärireeglid .....	19
3. Analüüs .....	20
3.1 Tegutsejad .....	20
3.2 Funktsionaalsed nõuded .....	20
3.3 Mittefunktsionaalsed nõuded .....	21
3.4 Kasutusjuhtude mudel .....	22
3.4.1 Külastaja kasutusjuhud .....	23
3.4.2 Töötaja kasutusjuhud .....	26
3.4.3 Juhataja kasutusjuhud .....	27
3.4.4 Administraatori kasutusjuhud .....	29
3.5 Kontseptuaalmudel .....	30
4. Disain .....	32
4.1 Kasutatavad vahendid ja tehnoloogiad .....	32
4.2 Arhitektuur .....	33
4.3 Andmemudel .....	34
4.3.1 Tabelite detailsed kirjeldused .....	36
4.4 Prototüüp .....	42
5. Võrdlus teiste keskkondadega .....	45

5.1 Naudi.ee.....	45
5.2 Vabalaud.ee .....	45
5.3 Keskkondade puudused ja tugevused .....	46
6. Võimalikud edasiarendused.....	47
Kokkuvõte .....	49
Summary.....	50
Kasutatud kirjandus .....	51
Lisa 1 .....	53

# **1. Sissejuhatus**

Käesolev töö püüab pakkuda välja lahenduse toitlustettevõtetele, millega külastajad saaksid mugavalt ja interaktiivselt reserveerida söögikohas endale laudu. Lisaks pakub lahendus töötajatele välja mugavama viisi kinnitatud broneeringuid näha ning neid hallata.

## **1.1 Taust ja probleem**

Elame ajastul, kus aina rohkem ja rohkem igapäevaseid rutiinseid tegevusi viiakse rohkem elektroonilisele kujule. Käesoleva töö eesmärk on võtta üks üldtuntud protsess – laua broneerimine mõnda toitlustusettevõttesse, analüüsida seda ning seejärel kaasajastada protsessi infotehnoloogiliste vahenditega.

Osapooled, keda käesolev töö võiks huvitada, on kindlasti toitlustusettevõtete liikmed. Lahendus pakub välja veel ühe võimaliku viisi külastajateni jõudmiseks reklaami või suusõnalise informatsiooni kõrval.

## **1.2 Ülesande püstitus**

Töö esimeseks eesmärgiks on analüüsida ning uurida praegust laua broneerimise protsessi ja tuua välja praeguse protsessi tugevused ja nõrkused.

Seejärel analüüsitakse, kuidas praegust protsessi saaks parendada, muutes seda mugavamaks mõlemale osapoolle (ettevõttele ja külastajale). Samuti on eesmärgiks uurida, kas töös välja pakutud protsessile analoogset lahendust Eestis juba mõni ettevõtte pakub, ning põgusalt analüüsida juba leiduvaid lahendusi, tuues välja ka nende tugevused ja nõrkused.

Võttes arvesse praegust protsessi, selle parendamiseks väljapakutud ideid ning õppides olemasolevatest analoogsetest lahendustest, pakutakse töö autori poolt välja nägemus uudest toitlustusettevõtete broneerimissüsteemist.

### **1.3 Metoodika**

Metoodika, millega püütakse töös seatud eesmärkideni jõuda, on kõigepealt üldtuntud protsessi analüüsimine. Analüüsimise käigus koostada praeguse protsessi kohta BPMN notatsioonis diagramm, mis tagab suurema ülevaatlikkuse.

Seejärel, arvestades protsessi analüüsi tulemusi, analüüsitakse uut äriprotsessi teostavat keskkonda. Peamiselt toetub töö autor siinkohal teoreetilistele ja praktilistele alustele, mida ta on omandanud oma bakalaureuseõppe kestel.

### **1.4 Ülevaade tööst**

Töö on jaotatud viie peatükiga loogilisteks osadeks. Esimeses peatükis on vaatluse all praegune ja väljapakutav äriprotsess broneeringu tegemisel. Teises peatükis on väljapakutud protsessi realiseeriva võimaliku keskkonna analüüs ning seejärel, kolmandas peatükis, on kirjeldatud võimaliku keskkonna disaini. Neljas peatükk keskendub põgusale uurimusele, kus töö autor püüdis leida olemasolevaid ning sarnast funktsionaalsust pakkuvaid keskkondi. Viimases peatükis on töös kirjeldatud keskkonna võimalikke edasiarendusi.

## **2. Broneerimise äriprotsess**

Järgnevalt on vaatluse all praegune ja lahendusena välja pakutav ühe toitlustusettevõtte broneerimise põhiprotsess. Eesmärgiks on broneerimise protsessi võimalikult suur automatiseerimine.

### **2.1 Olemasolev protsess**

Olemasolev protsess, mida järgnevalt uuritakse, on näiteks restorani (üks võimalik toitlustusettevõtte) laua broneerimine, mis on kujutatud BPMN notatsioonis joonisel 1. Restorani laua broneerimine on piisavalt üldine protsess, mida võib kasutada ka muude toitlustusettevõtete broneerimisprotsessi iseloomustamiseks (sööklad, pubid, baarid jne).

Selle protsessi üldise kulgemisega peaks kursis olema väga paljud inimesed. Sellest hoolimata tuuakse välja üksikasjalikult protsessi tugevused ja puudused.

#### **2.1.1 Kirjeldus**

Praeguse seisuga toimub enamus broneeringuid restoranidesse telefoni või meili teel. Kõigepealt kogub võimalik restoranikülastaja endale informatsiooni, millises restoranis ning millisel ajal einestada. Valiku tegemiseks kasutab potentsiaalne külastaja varasemat kogemust, lähikondlaste ja / või sõprade kogemust, reklaame ning ettevõtte veebilehekülgi. Kui külastaja on valinud välja enda jaoks meelepärase restorani, siis algab protsessi kõige tähtsam osa – laua broneerimine.

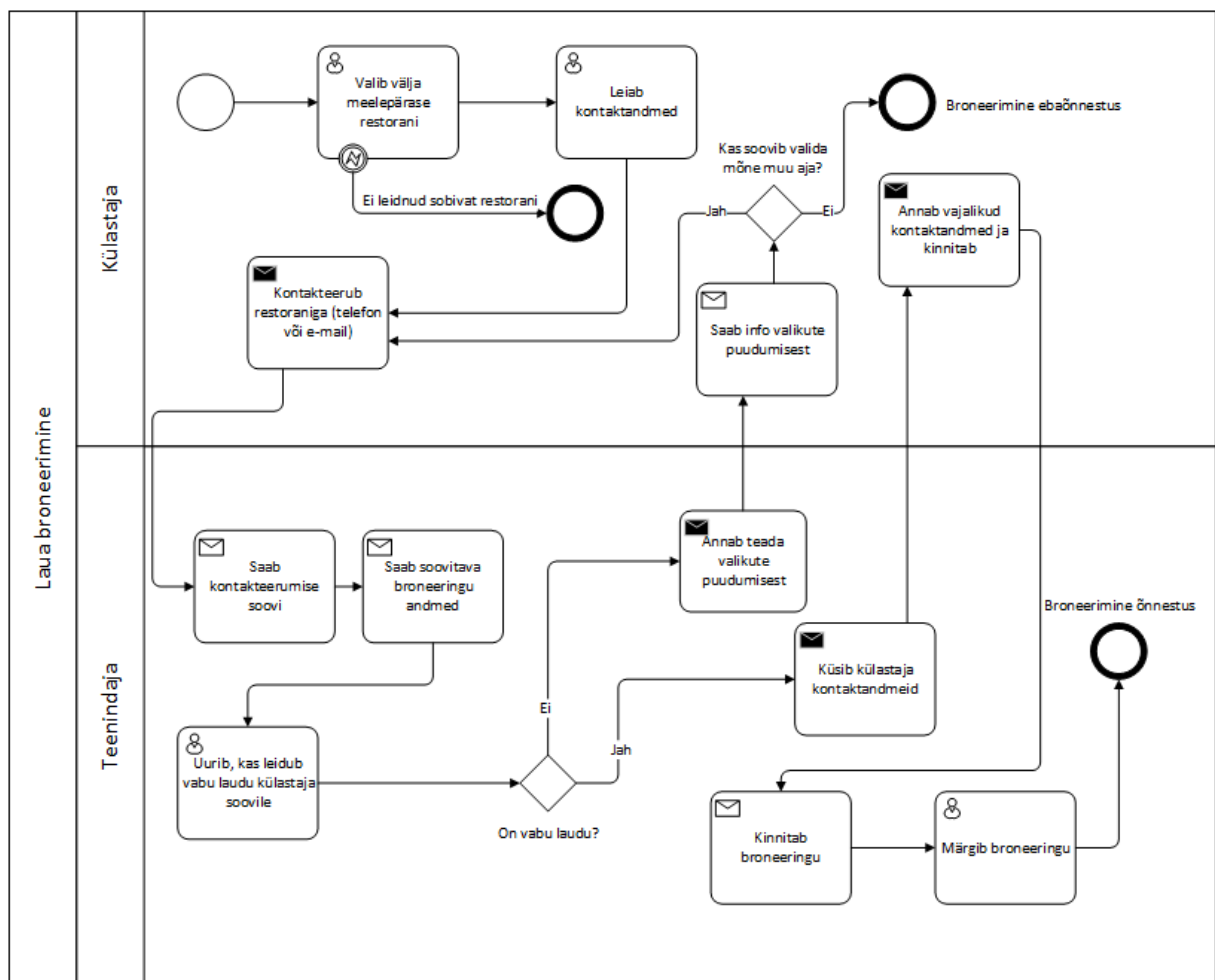
Broneerimine ei pruugi alati olla nõutud tegevus. Nimelt võib klient saada restoranis koha ka ilma broneerimata, kuid peab siiski arvestama võimalusega, et restoranis puuduvad soovitud ajal vabad kohad või restorani poliitika näeb ette laudade eelnevat broneerimist. Lihtsuse huvides on protsessi analüüsil eeldatud, et klient tõesti soovib eelnevalt ja kindluse mõttes koha restoranis broneerida ning ei arvesta võimalusega, et saab endale ilma aega broneerimata vaba koha.

Broneerimiseks otsib külastaja välja restorani kontaktandmed – telefoninumbri või e-posti aadressi. Üldiselt võime eeldada, et klient (edaspidi külastaja) eelistab helistada restoranis, sest

nii saab asuda dialoogi restoranis oleva töötajaga (edaspidi teenindaja), mis tagab kiirema broneerimisprotsessi kulgemise nii külastaja kui ka ettevõtte jaoks.

Seejärel esitab külastaja oma broneeringu ajalise soovi ning koostöös teenindajaga leitakse mõlemaid osapooli rahuldav lahendus ehk kas külastaja poolt sobiva suuruse lauda on soovitud ajal restoranis saadaval. Kui laud on saadaval, teeb teenindaja märkme broneeringust mõnele paberdokumendile või paremal juhul mõnda ERP vahendisse.

Broneeringu kinnitamiseks oleks kindlasti vaja külastaja nime, kelle nimele broneering tehakse, ning kontaktandmeid juhuks, kui broneeringut oleks vaja külastajaga uuesti täpsustada.



**Joonis 1.** Praeguse broneerimise äriprotsessi kirjeldus

### 2.1.2 Tugevused ja puudused

Puuduseks sellise protsessi juures on olukord, kus külastaja ei saa täpselt teada, millise asukohaga talle broneeritud laud restoranis on. Näiteks võib olla rohkem privaatsust sooviva

külastaja korral broneering tehtud suurema seltskonna jaoks mõeldud laua kõrvale ning seetõttu on külastaja kogemus tugevalt ja negatiivselt mõjutatud kõrvalasuvast rohkearvulisest seltskonnast. Samuti võib olla lastega külastajate jaoks broneering tehtud restorani lastetoast kaugemale, mis võib samuti mõjutada negatiivselt külastajate kogemust.

Puuduseks on samuti ka olukord, kus külastajale sobivate näitajatega (kuupäev, kellaaeg, asukoht, laua suurus) vaba lauda ei ole. Sellisel juhul tuleks leida koostöös klienditeenindajaga kompromiss või peab külastaja leidma mõne muu restorani. Konkreetne puudus siinkohal on nii külastaja kui ka teenindaja ajaline kulu, mis sisuliselt mitte mingit tulemust ei andnud – külastaja läheb teise ettevõtte restorani einestama ning teenindaja väärtuslik tööaeg ei andnud soovitud tulemust.

Ühtlasi on meili teel broneerimisel suur ajaline kulu. Iga külastaja broneeringusoovile tuleks vastata kirjalikult ja oodata ka kirjalikku vastust. Ajaline kulu sõltub sellest, kui tihti restorani personal loeb meili ning kui kiiresti jõuab ka külastaja nendele vastata. Mitme meili edasi ja tagasi saatmisel mitmele broneeringusoovile on summaarselt ajaline kulu märkimisväärne. Sellest tulenevalt on raskendatud ka broneeringute ülevaatlikkus restorani personali jaoks.

Praegust protsessi on kulukas teha ka paralleelseks. Sellisel juhul tuleks paigaldada näiteks mitu telefoni ning palgata rohkem teenindajaid, et samaaegselt vastu võtta mitme külastaja broneeringusoovid.

Järgnevalt (Tabel 1) on ülevaatlikult toodud protsessi puuduse mõõdikud. Teenindaja kogukulu sõltub suuresti restorani populaarsusest ja suuruselt, kuid lihtsuse huvides on võetud näiteks olukord, kus teenindaja võtab päevas vastu 10 broneeringut ning keskmine broneering kestab 7 minutit.

**Tabel 1. Olemasoleva protsessi puudused ja mõõdikud**

<b>Kasutaja</b>	<b>Keskmine kulu</b>	<b>Kogus</b>	<b>Ajakulu päevas</b>
<b>Teenindaja</b>	7 minutit	10 broneeringut	70 minutit
<b>Külastaja</b>	7 minutit	1 broneering	7 minutit

Protsessi tugevuseks on kindlasti tema lihtsus. Broneerimiseks ei ole põhimõtteliselt tehnilisi oskusi kasutada arvutit, internetti ja erinevaid veebilehitsejaid. Seetõttu saavad selle protsessi iseseisvalt läbimisega hakkama peaaegu igas vanusegrupis olevad isikud. Samuti võib mõni



külastaja eelistada vahetut suhtlemist klienditeenindajaga ainult selleks, et saada personaalset ja vahetut teenindust.

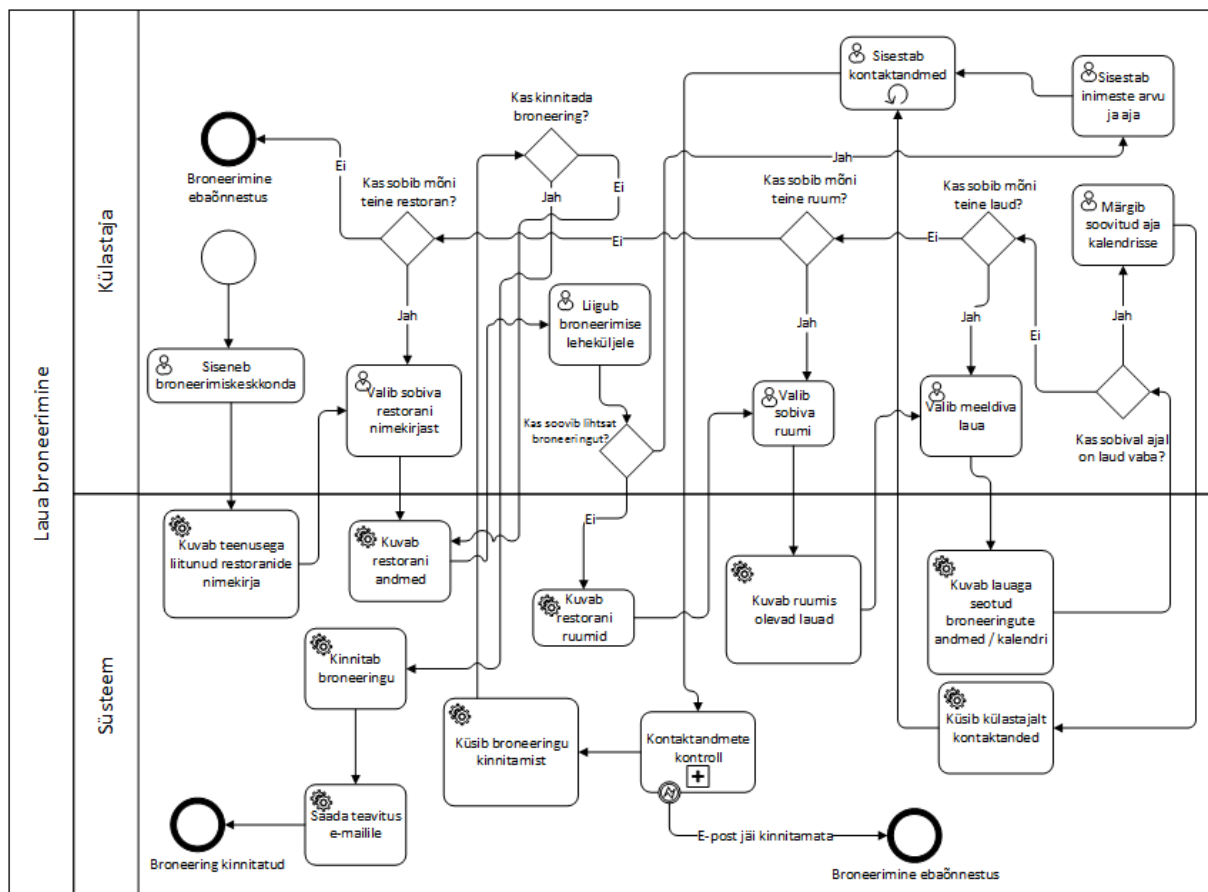
## **2.2 Väljapakutav lahendus**

Väljapakutava lahenduse eesmärgiks on võimalikult palju automatiseerida broneerimisprotsessi. Tulemusena õnnestus teenindaja roll broneerimise protsessist eemaldada ning teda asendab broneerimise infosüsteem.

### **2.2.1 Broneerimise põhiprotsess**

Joonisel 2 kujutatud protsessi tegevuste jada on küll pikem, aga sealt on näha, et väljapakutud protsess on mugavam nii kasutaja kui ka ettevõtte jaoks. Peamised muutused praeguse broneerimisprotsessiga võrreldes:

- külastajal on suurem valikuvõimalus restorani ruumi ja laua asukoha üle;
- teenindaja roll broneerimisprotsessis otseselt puudub;
- süsteem võimaldab teenindada ühel restoranil samaaegselt mitut broneeringuprotsessi samaaegselt;
- süsteem võimaldab külastajal kasutada ka lihtsamat broneeringu protsessi, kus ei pea ruumi ja lauda valima (sellisel juhul on vajalik broneeringu sisendandmetena inimeste arv, broneeringu aeg ning kontaktandmed).

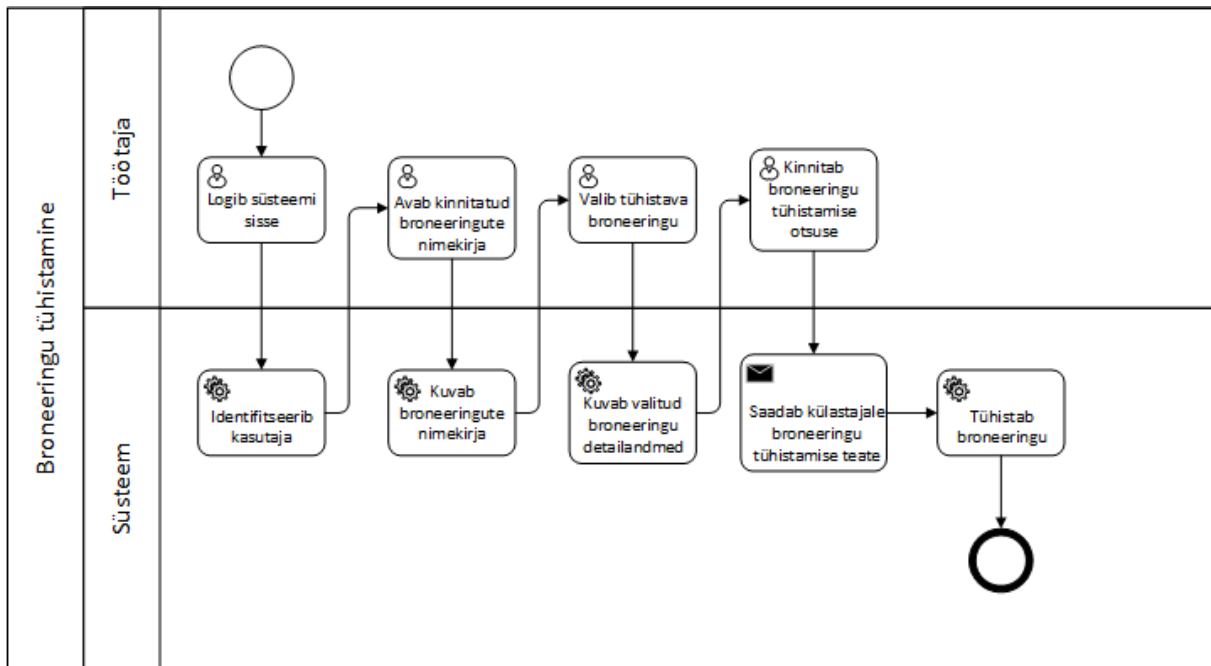


**Joonis 2.** Lahendusena välja pakutav broneerimise äriprotsess

Süsteem muudab turvalisemaks kasutaja tuvastamise. Kontaktandmete küsimise järel saadetakse külastaja poolt sisestatud e-posti aadressile kinnituskirja, milles sisalduva lingi avamise järel saab broneeringu kinnitamisega jätkata. Kinnituskirja eesmärk on tuvastada, kas sisestatud e-posti aadress on tõesti külastaja poolt kasutatav ning samuti aitab see takistada robotite ning petturite tegevusi. Ühtlasi võib alternatiivina kasutada CAPTCHA kontrolli. Selline turvameede rakendub siis, kui külastaja ei oma antud broneeringu portaalis eelnevalt registreeritud kasutajat.

### 2.2.2 Broneeringu tühistamise protsess

Järgnevalt on näitena ära kirjeldatud ja joonisel 3 kujutatud olukord, kus restorani töötaja on sunnitud mõnel ärilisel eesmärgil tühistama külastaja broneeringut.



**Joonis 3.** Broneeringu tühistamise äriprotsess

Ettevõtte töötaja jaoks on tehtud protsess üsnagi lihtsaks, kus tegevuseks on tühistatava broneeringu leidmine ning selle tühistamine. Infosüsteem teavitab ise külastajat tema broneeringu tühistamisest ning seejärel viib muudatused sisse.

### 2.2.3 Ärireeglid

Järgnevalt on ära toodud broneerimise protsessi ärireeglid:

1. külastaja teeb broneeringu;
2. üks külastaja võib omada mitut broneeringut;
3. toitlustusettevõttel on üks või rohkem ruumi;
4. ruumis on üks või rohkem lauda;
5. laual on mitu broneeringut;
6. ühel laual ei või broneeringu ajad kattuda;
7. töötaja näeb broneeringuid;
8. töötaja võib broneeringuid tühistada;
9. süsteem kontrollib broneeringute korrektsust;
10. süsteem identifitseerib sisselogimisel kasutaja.

### 3. Analüüs

Järgnevalt on esitatud väljapakutud äriprotsessi realiseeriva keskkonna analüüs.

#### 3.1 Tegutsejad

Järgnevalt on toodud välja loodava keskkonna tegutsejad:

- **Külastaja** – keskkonna peamine kasutaja. Isik, kes teeb broneeringuid eesmärgiga külastada toitlustusettevõtet. Külastajaks loetakse nii sisse loginud kui ka sisse logimata külastajat.
- **Töötaja** – toitlustusettevõtte töötaja, peamiselt teeninda või kelner. Haldab broneeringuid ja laudadega seotud informatsiooni.
- **Juhataja** – toitlustusettevõtte juhataja või töötaja, kellele on usaldatud keskkonnas temaga seotud ettevõtte haldamise suhtes suuremad õigused. Haldab ettevõtte ruume ja üldist ettevõttega seotud informatsiooni.
- **Administraator** – keskkonna administraator. Täidab peamiselt keskkonna haldamisega seotud ülesandeid ning omab kõige suuremat keskkonna õiguste komplekti.

#### 3.2 Funktsionaalsed nõuded

Järgnevalt on toodud ära keskkonna funktsionaalsed nõuded:

1. keskkond peab olema veebipõhine;
2. avalehel peab olema nähtav keskkonnaga liitunud ettevõtete nimekiri;
3. keskkonnaga liitunud ettevõtete nimekirja saab filtreerida nime järgi;
4. avalehel on link keskkonda haldava üksuse kontaktandmetele;
5. ettevõtte vaates saab näha ettevõtte kontaktandmeid, nimetust, logo ja söögikoha tüüpi;
6. ettevõtte vaates on võimalik näha ettevõtte toidukaarti (menüüd);
7. ruumi vaates on näha ruumi plaan ja laudade asukohad selles;
8. broneeringu vaates on võimalik vaadata lauaga seotud broneeringuid;

9. broneeringu kalendrisse saab sisestada soovitud broneeringu;
10. on võimalus teha ka lihtne broneering, mis ei nõua ruumi ja konkreetse laua valikut;
11. keskkonda on võimalik sisse logida;
12. keskkonda sisse loginud töötajal ja juhatajal on võimalik vaadata ainult temaga seotud ettevõtteid;
13. töötajal ja juhatajal on võimalik hallata ettevõttega seotud andmeid;
14. töötajal ja juhatajal on võimalik hallata broneeringuid;
15. juhatajal on süsteemis võimalik hallata ettevõtte ruume;
16. administraatoril on võimalik hallata kõikide ettevõtetega seotud andmeid.

### 3.3 Mittefunktsionaalsed nõuded

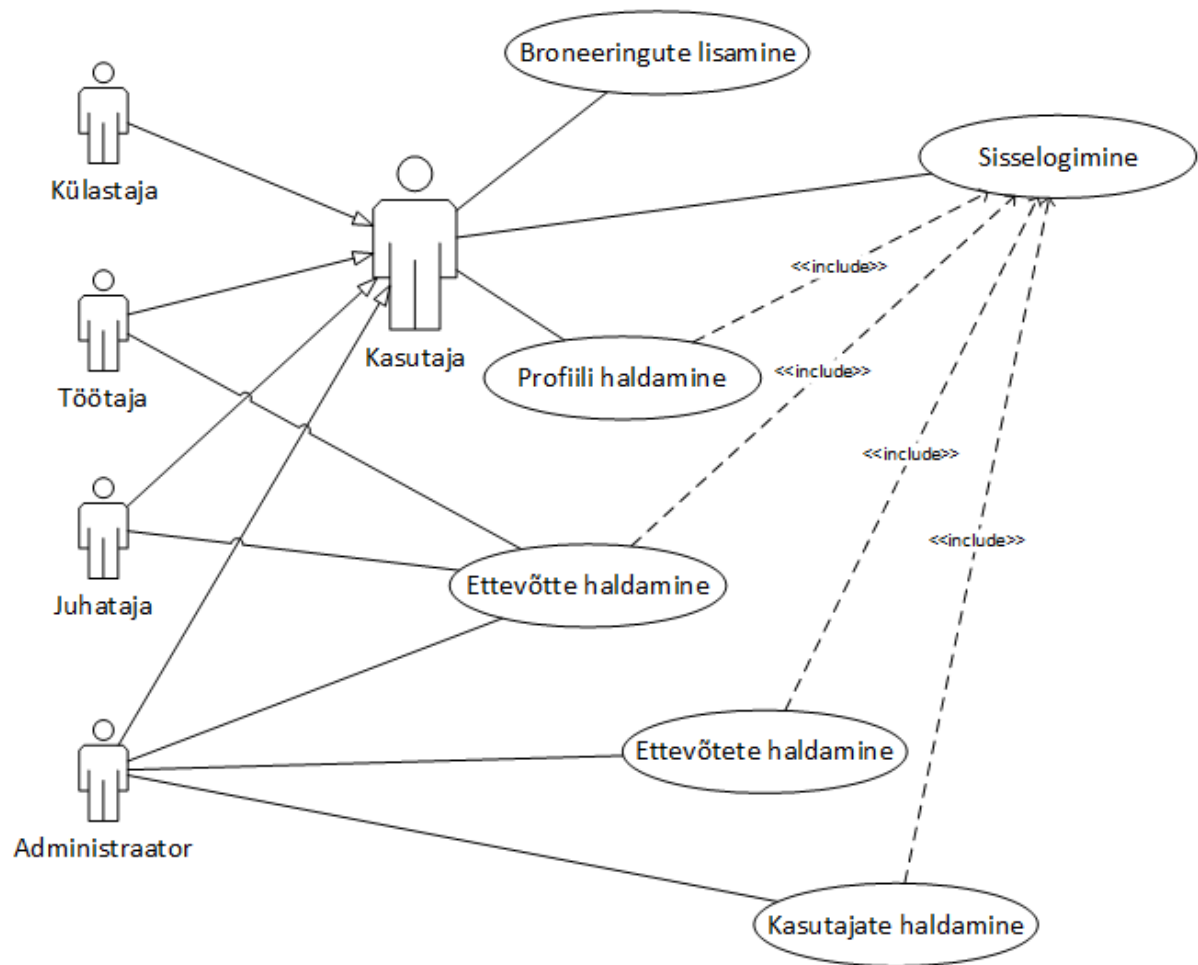
Järgnevalt on toodud ära keskkonna mittefunktsionaalsed nõuded:

1. **Veebilehitsejate tugi** – keskkond peab olema ligipääsetav ja töötav enimlevinud personaalarvuti veebilehitsejatega. Veebilehitsejateks on [13]:
  - a. Apple Safari
  - b. Google Chrome
  - c. Mozilla Firefox
  - d. Microsoft Internet Explorer 11
  - e. Opera
2. **Keskkonna jõudlus** – keskkond peaks olema suuteline teenindama korraga 50 klienti.
3. **Kasutajaliides** – keskkonna kasutajaliides peab olema disainitud intuiivselt, et ka esmakordsed külastajad saaksid hõlpsasti rakenduse kasutamisega hakkama.
4. **Dokumenteeritus** – keskkond peaks olema põhjalikult dokumenteeritud, et edasiarendusi oleks võimalikult lihtne teostada ka arendajate vahetumisel.
5. **Arendusvahendid** – analüüsimisel kasutatavad arendusvahendid on peamiselt UML diagrammide joonistamiseks mõeldud tarkvara. Töö autor on käesoleva töö joonised teinud Microsoft Visio 2013 tarkvaraga. Arendamisel kasutatavate vahendite valik sõltub suuresti arendajate kogemustest ja eelistustest ning saadaval on mitmeid erinevaid tasuta ja tasulisi arendusvahendid PHP, MySQL ja HTML jms tehnoloogiate jaoks.

6. **Keel** – keskkonna dokumentatsioon ja kasutajaliides peavad olema eesti keeles. Keskkond tuleks arendada nii, et muude keelte lisamine keskkonna kasutajaliidesesse oleks võimalikult lihtne ja ei nõuaks kogu keskkonna või selle osade uuesti arendamist.
7. **Serveri tarkvara** – kulude kokkuhoiduks võiks serveri operatsioonisüsteem toimida Linuxi kernelil, kuid kaaluda tuleks ka professionaalse tehnilise ja kasutajatoe võimalusega, mis tuleb kaasa tasuliste operatsioonisüsteemidega. Sama arutluskäiku võib rakendada ka andmebaasisüsteemi valimisel. Kas valida tasuta, populaarne ja või tasulise, aga kindla tehnilise ja kasutajatoe olemasoluga süsteem? Siinkohal on eelistatud MySQL andmebaasisüsteemi.

### **3.4 Kasutusjuhtude mudel**

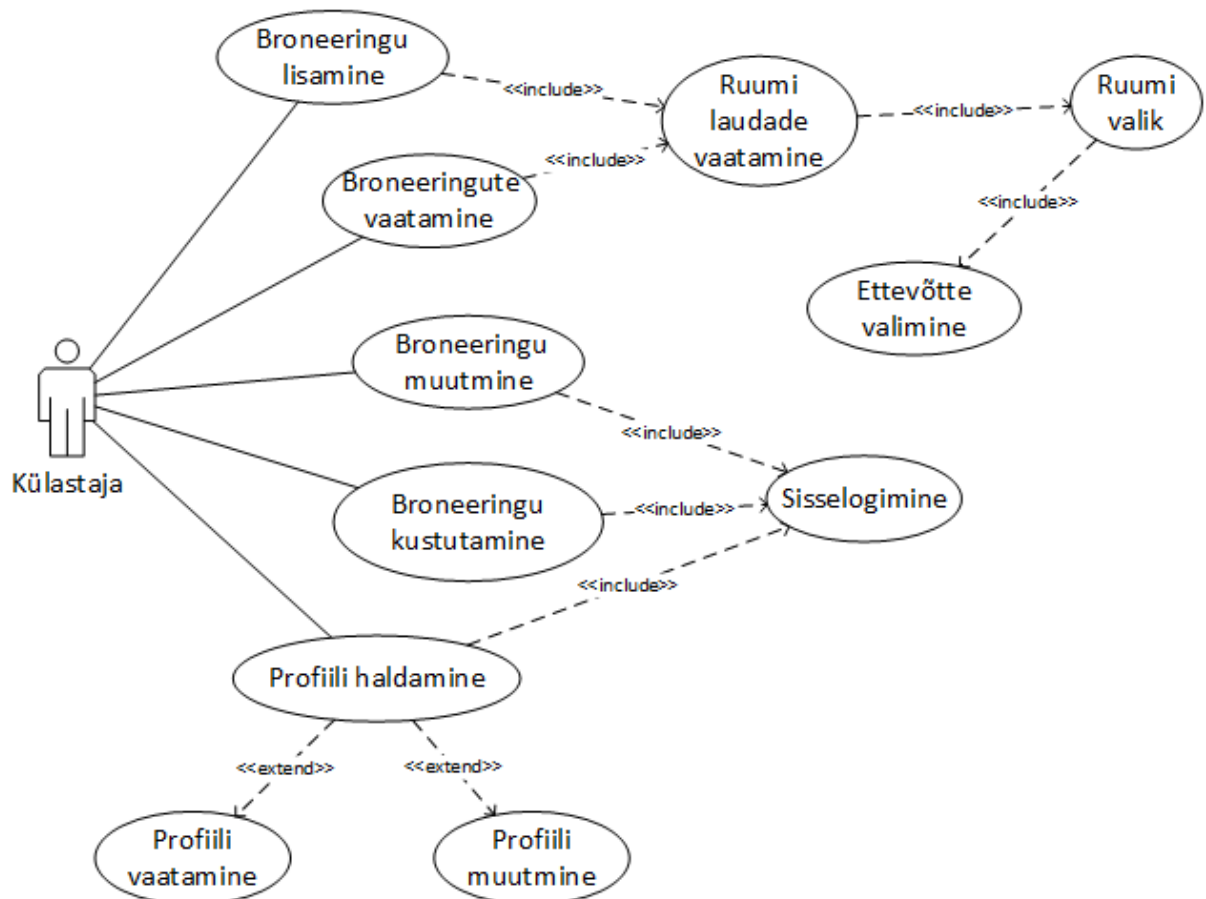
Järgnevalt on kirjeldatud keskkonna kasutusjuhte. Keskkonna kõikide kasutajate kokkuvõttev kasutusjuhtude diagramm on esitatud joonisel 4.



**Joonis 4.** Üldine kasutusjuhtude diagramm

### 3.4.1 Külastaja kasutusjuhud

Järgnevalt on ära kirjeldatud joonisel 5 kujutatud külastaja rolli kasutusjuhud.



**Joonis 5.** Külastaja kasutusjuhtude diagramm

**Nimi:** UC1.1 Ettevõtte valimine

**Tegutsejad:** külastaja, administraator

**Kirjeldus:** Külastaja ning administraator saavad näha keskkonnas leiduvaid ettevõtteid, et nendega sooritada edasisi tegevusi. Külastaja valib ettevõtte broneerimise protsessi tarbeks, administraator ettevõtte haldamisega seotud tegevusteks.

**Nimi:** UC1.2 Ruumi valik

**Tegutsejad:** külastaja, töötaja, juhataja, administraator

**Kirjeldus:** Kasutajal on võimalus näha toitlustusettevõtte söögiruumide nimekirja.

**Nimi:** UC1.3 Ruumi laudade vaatamine

**Tegutsejad:** külastaja, töötaja, juhataja, administraator

**Kirjeldus:** Kasutajale kuvatakse ruumi laudade paigutus graafiliselt.

**Nimi:** UC1.4 Broneeringu vaatamine

**Tegutsejad:** külastaja, töötaja, juhataja, administraator



**Kirjeldus:** Konkreetse laua valimisel kuvatakse kasutajale broneeringute nimekiri kalendrivaates. Kasutajal on võimalus liikuda kalendris edasi või tagasi. Kalender on nädalavaates.

**Nimi:** UC1.5 Broneeringu lisamine

**Tegutsejad:** külastaja

**Kirjeldus:** Külastaja saab valitud laua kalendrisse vabale kohale sisestada broneeringu. Pärast broneeringuaja valimist küsib süsteem kasutaja kontaktandmeid ainult juhul, kui külastaja ei ole süsteemi sisse loginud.

**Nimi:** UC1.6 Kasutaja sisselogimine

**Tegutsejad:** kasutaja

**Kirjeldus:** Kasutaja saab keskkonda sisse logida, kasutades kasutajanime ja parooli. Juhul, kui kasutajanimi või parool on valed, siis keskkonda sisselogimine ebaõnnestub.

**Nimi:** UC1.7 Broneeringu muutmine

**Tegutsejad:** külastaja

**Kirjeldus:** Juhul kui külastaja on teinud broneeringuid oma kasutajaga, siis on tal võimalus oma broneeringuid vaadata ning neid muuta. Teised kasutajad saavad muuta kõiki broneeringuid.

**Nimi:** UC1.8 Broneeringu kustutamine

**Tegutsejad:** kasutaja

**Kirjeldus:** Külastaja saab oma broneeringuid eemaldada, teised kasutajad saavad kõiki broneeringuid eemaldada.

**Nimi:** UC1.9 Profiili vaatamine

**Tegutsejad:** kasutaja

**Kirjeldus:** Kasutaja saab vaadata oma profiili andmeid, mis on keskkonda sisestatud. Samast vaatest on tal võimalus ka andmeid muuta, kaasaarvatud parooli.

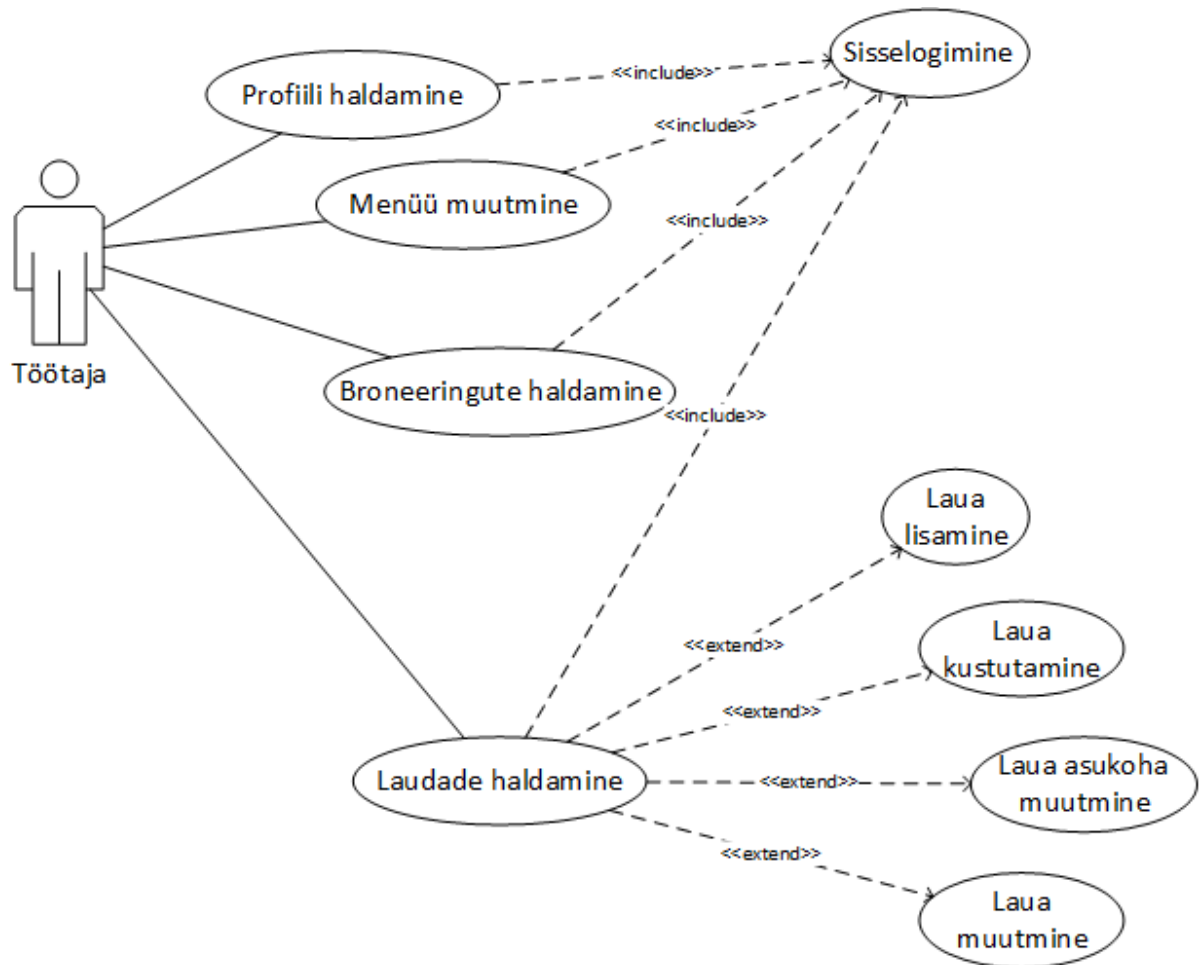
**Nimi:** UC1.10 Profiili muutmine

**Tegutsejad:** kasutaja

**Kirjeldus:** Kasutaja saab muuta oma profiiliga seotud andmeid.

### 3.4.2 Töötaja kasutusjuhud

Järgnevalt on ära kirjeldatud joonisel 6 kujutatud töötaja rolli kasutusjuhud.



**Joonis 6.** Töötaja kasutusjuhtude diagramm

**Nimi:** UC2.1 Menüü muutmine

**Tegutsejad:** töötaja, juhataja, administraator

**Kirjeldus:** Eelnimetud tegutsejad saavad muuta ettevõttega seotud toidukaardi (menüü) sisu.

**Nimi:** UC2.2 Laua lisamine

**Tegutsejad:** töötaja, juhataja, administraator

**Kirjeldus:** Tegutseja saab lisada ruumi laua. Laua lisamisel sisestab tegutseja laua põhiaandmed: nimetus, laius, pikkus, istekohtade arv, kuju ning ruumi.

**Nimi:** UC2.3 Laua muutmine

**Tegutsejad:** töötaja, juhataja, administraator

**Kirjeldus:** Tegutseja saab muuta lauaga seotuid andmeid: nimetus, laius, pikkus, istekohtade arv, kuju ja ruum, kus laud asub.

**Nimi:** UC2.4 Laua kustutamine

**Tegutsejad:** töötaja, juhataja, administraator

**Kirjeldus:** Tegutseja saab kustutada valitud laua.

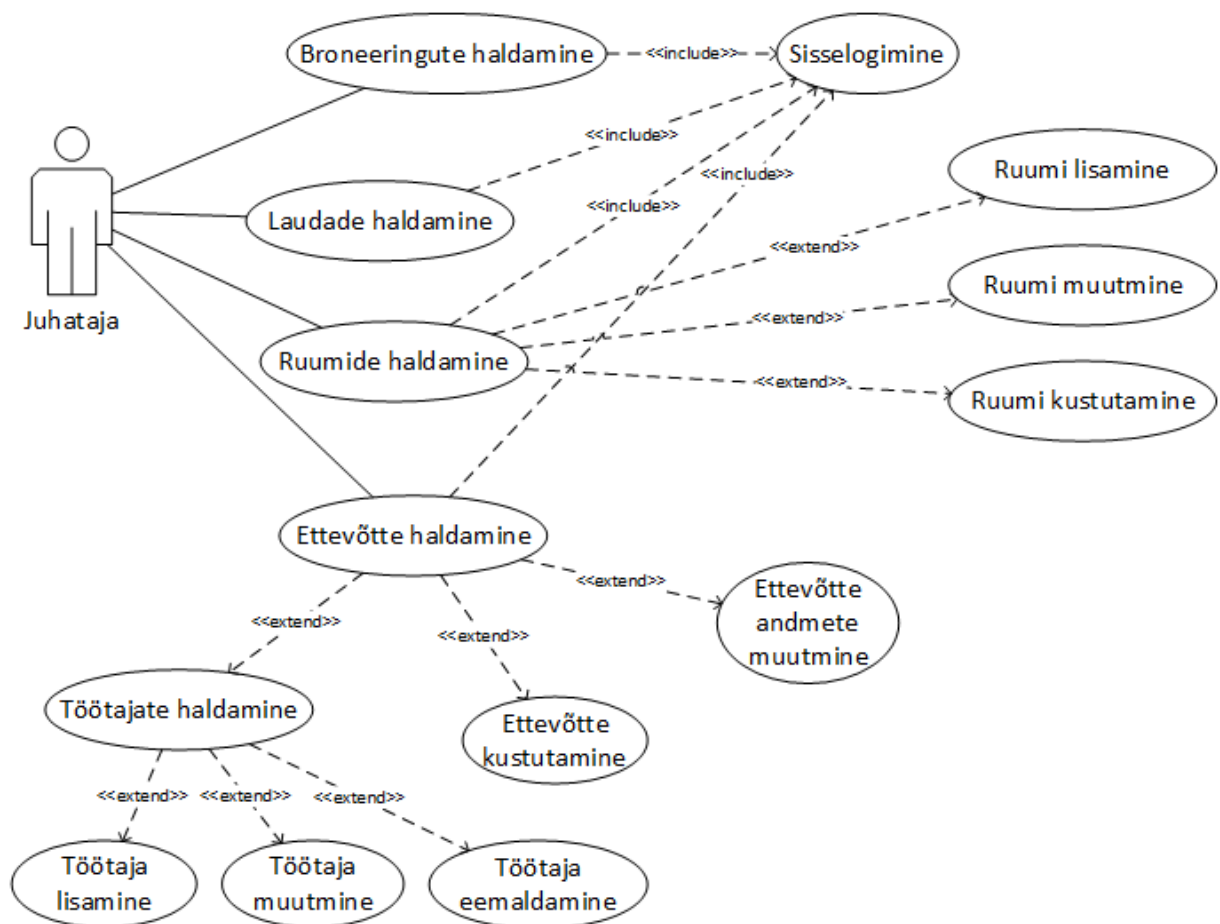
**Nimi:** UC2.5 Laua asukoha muutmise

**Tegutsejad:** töötaja, juhataja, administraator

**Kirjeldus:** Tegutseja saab muuta graafiliselt laua asukohta ruumis. Pärast salvestamist salvestatakse laua asukoht süsteemis ning muudatused on kohe ka külastajatele näha.

### 3.4.3 Juhataja kasutusjuhud

Järgnevalt on ära kirjeldatud joonisel 7 kujutatud juhataja rolli kasutusjuhud.



**Joonis 7.** Juhataja kasutusjuhtude diagramm

**Nimi:** UC3.1 Ruumi lisamine

**Tegutsejad:** juhataja, administraator

**Kirjeldus:** Tegutseja saab lisada ruumi, täites selleks ära põhiandmestiku: ruumi nimetus ja põrandaplaani joonis.

**Nimi:** UC3.2 Ruumi muutmine

**Tegutsejad:** juhataja, administraator

**Kirjeldus:** Tegutseja saab muuta ruumiga seotud andmeid: ruumi nimetus ja põrandaplaani joonis.

**Nimi:** UC3.3 Ruumi kustutamine

**Tegutsejad:** juhataja, administraator

**Kirjeldus:** Tegutseja saab kustutada ruumi, kuid ainult juhul, kui ruumiga ei ole seotud ühtegi lauda. Püüdes kustutada ruumi, kus sees on lauad, kuvab keskkond veateate ning ruumi ei kustutata.

**Nimi:** UC3.4 Ettevõtte andmete muutmine

**Tegutsejad:** juhataja, administraator

**Kirjeldus:** Tegutsejal on võimalus muuta ettevõttega seotud andmeid: nimi, aadress, logo, tutvustus, söögikoha tüüp jne.

**Nimi:** UC3.5 Ettevõtte kustutamine

**Tegutsejad:** juhataja, administraator

**Kirjeldus:** Juhul, kui tegutseja soovib oma ettevõtet eemaldada keskkonnast, on tal võimalus kustutada ettevõtte.

**Nimi:** UC3.6 Töötaja lisamine

**Tegutsejad:** juhataja, administraator

**Kirjeldus:** Peamiselt ettevõtte juhatajale vajalik funktsioon, millega saab ta lisada keskkonda kasutajad, kes on seotud antud ettevõttega. Kasutajad saavad vaikimisi endale töötaja õigused seotud ettevõttega. Töötaja lisamisel sisestab juhataja kasutaja loomiseks vajalikud andmed: eesnimi, perenimi, kasutajanimi, parool ning kasutajaõigused (töötaja või juhendaja).

**Nimi:** UC3.7 Töötaja muutmine

**Tegutsejad:** juhataja, administraator

**Kirjeldus:** Tegutseja saab muuta töötajatega seotud andmeid: eesnimi, perenimi, kasutajanimi, parool ning kasutajaõigused (töötaja või juhendaja).

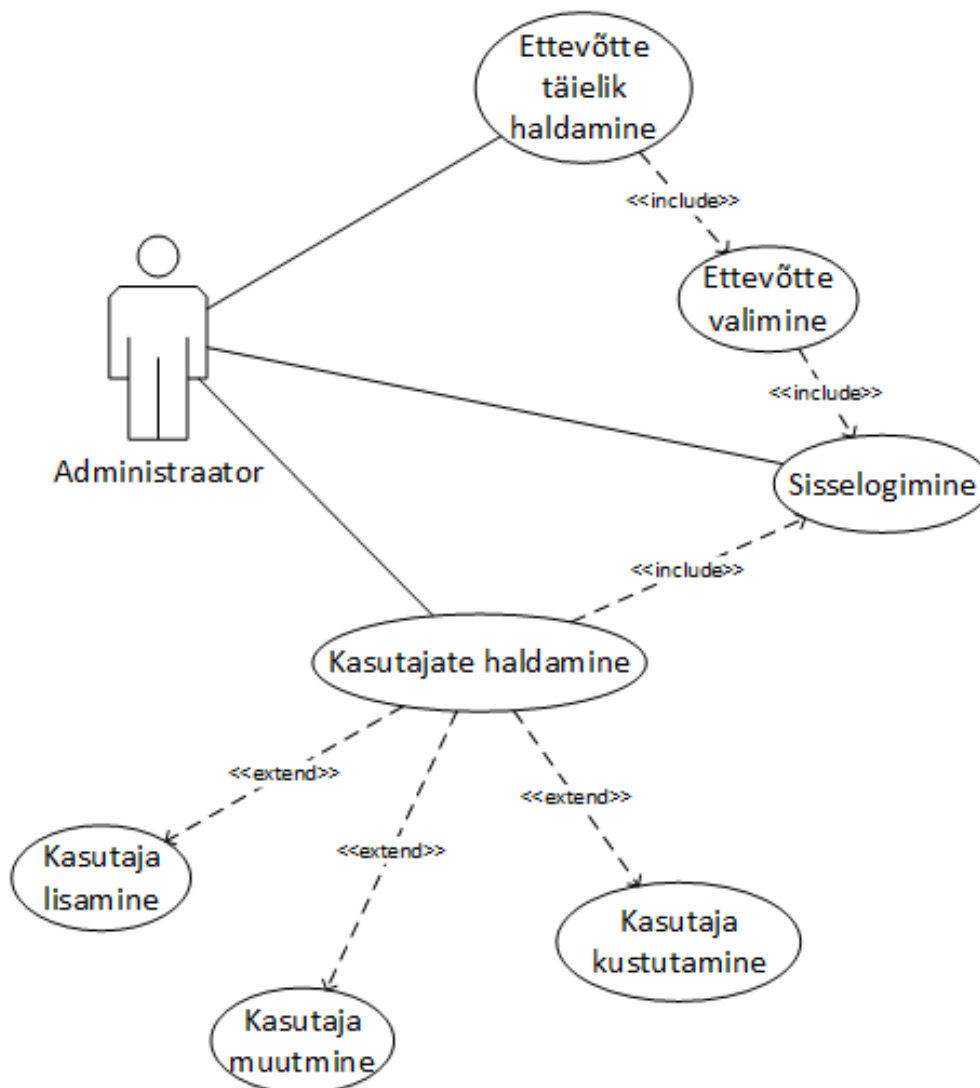
**Nimi:** UC3.8 Töötaja eemaldamine

**Tegutsejad:** juhataja, administraator

**Kirjeldus:** Tegutseja saab eemaldada ettevõttega seotud töötajaid. Antud võimalust peaks vaja minema juhul, kui töötaja lahkub ettevõttest ning kaob vajadus tema kasutaja järele keskkonnas.

### 3.4.4 Administraatori kasutusjuhud

Järgnevalt on ära kirjeldatud joonisel 8 kujutatud administraatori rolli kasutusjuhud.



**Joonis 8.** Administraatori kasutusjuhtude diagramm

**Nimi:** UC4.1 Ettevõtte täielik haldamine

**Tegutsejad:** administraator

**Kirjeldus:** Administraator saab hallata täielikult iga ettevõtte andmeid, täpselt nagu juhataja saab endaga seotud ettevõtte andmeid muuta. Konkreetne kasutusjuht on vajalik peamiselt keskkonna haldamiseks seotud ülesannete täitmiseks.

**Nimi:** UC4.2 Kasutaja lisamine

**Tegutsejad:** administraator

**Kirjeldus:** Administraator saab lisada keskkonda kasutajaid. Tüübilt võib ta lisada kõikide rollidega kasutajaid: külastaja, töötaja, juhataja ning administraatori. Lisaks saab ta siduda töötajat ja juhatajat konkreetse ettevõttega.

**Nimi:** UC4.3 Kasutaja muutmine

**Tegutsejad:** administraator

**Kirjeldus:** Sarnane kasutusjuhule UC4.2, kuid saab muuta keskkonna kasutaja andmeid. Konkreetne kasutusjuht on vajalik peamiselt keskkonna haldamiseks seotud ülesannete täitmiseks.

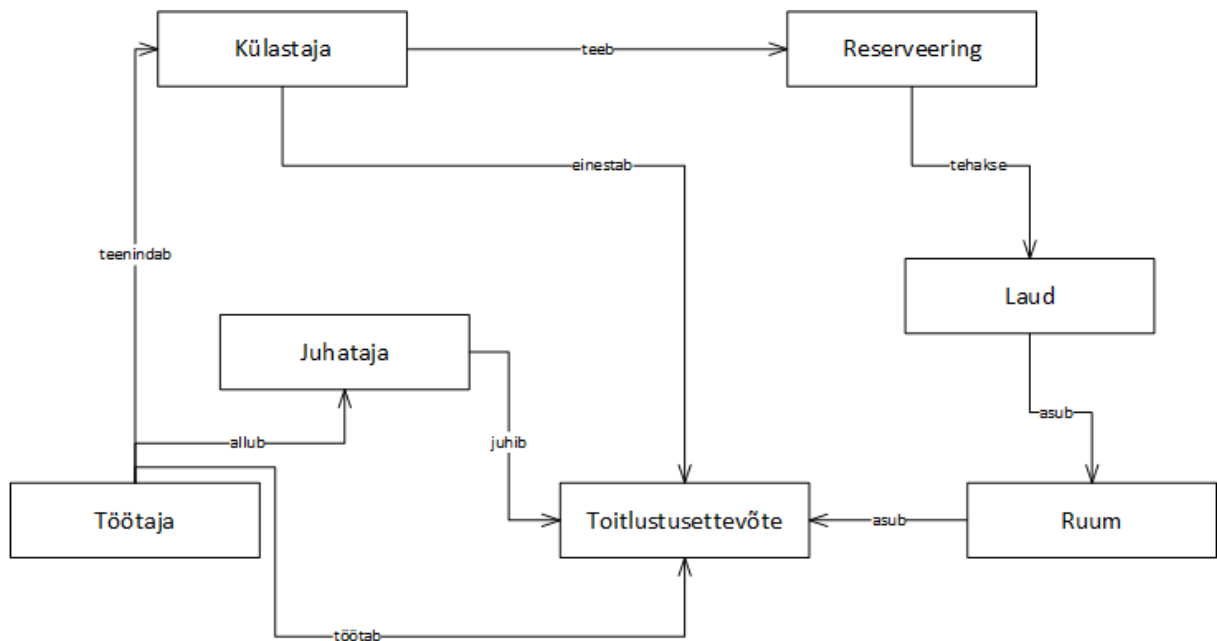
**Nimi:** UC4.4 Kasutaja kustutamine

**Tegutsejad:** administraator

**Kirjeldus:** Administraatoril on võimalus kustutada keskkonnast kasutajaid. Konkreetne kasutusjuht on vajalik peamiselt keskkonna haldamiseks seotud ülesannete täitmiseks.

### **3.5 Kontseptuaalmudel**

Järgnevalt, joonisel 9, on välja toodud keskkonna kontseptuaalmudel:



**Joonis 9.** Keskkonna kontseptuaalmudel

Joonisel 9 on näha, kuidas on kontseptuaalselt seotud keskkonna põhiobjektid. Külastaja teeb reserveeringuid. Reserveering tehakse ühele laule. Laud asub ruumis ning ruumid asuvad toitlustusettevõttes. Ettevõtet juhib juhataja ning talle alluvad ettevõtte töötajad töötavad selles ettevõttes. Töötajad teenindavad külastajat, kui külastaja saabub reserveeritud ajal lauda einestama.

## 4. Disain

### 4.1 Kasutatavad vahendid ja tehnoloogiad

**AJAX** – AJAX on kasutusel FullCalendar pistikprogrammi poolt. Broneeringute andmeid loetakse, luuakse, muudetakse ja kustutatakse baasist lehekülge uuesti laadimata.

**Bootstrap** – Bootstrap võimaldab mugavalt ja kerge vaevaga luua välimuselt silmapaistvaid lehekülgi nii mobiiliseadmetele kui ka lauarvutitele. Bootstrap kasutab CSSi kujunduselementide defineerimiseks.

**FullCalendar** – jQuery't kasutatav pistikprogramm, mis muudab mugavaks läbi veebivormi kuvatavate kalendrite kasutamise [14]. Keskkonnas kuvatakse selle pistikprogrammi abil broneeringuid kalendrivaates.

**JavaScript** – skriptimiskeel, mida täidetakse kliendipoolses veebilehitsejas. See on keskkonnas vajalik mitmete tegevuste (näiteks ettevõtete nimekirja kohesel filtreerimisel jne) jaoks.

**jQuery** – JavaScripti teek, mis laiendab ja lihtsustab tüüpiliste JavaScripti meetodite programmeerimist. Keskkonnas kasutusel just eelnimetatul põhjusel.

**jQuery UI** – JavaScripti ja jQuery peale üles ehitatud teek, mille peamiseks ülesandeks antud keskkonnas on laudade graafilise kuvamisega ja funktsionaalsuse realiseerimine.

**MySQL** – populaarne relatsiooniline andmebaasisüsteem [15], mille ülesandeks on keskkonna jaoks vajalike andmete hoiustamine.

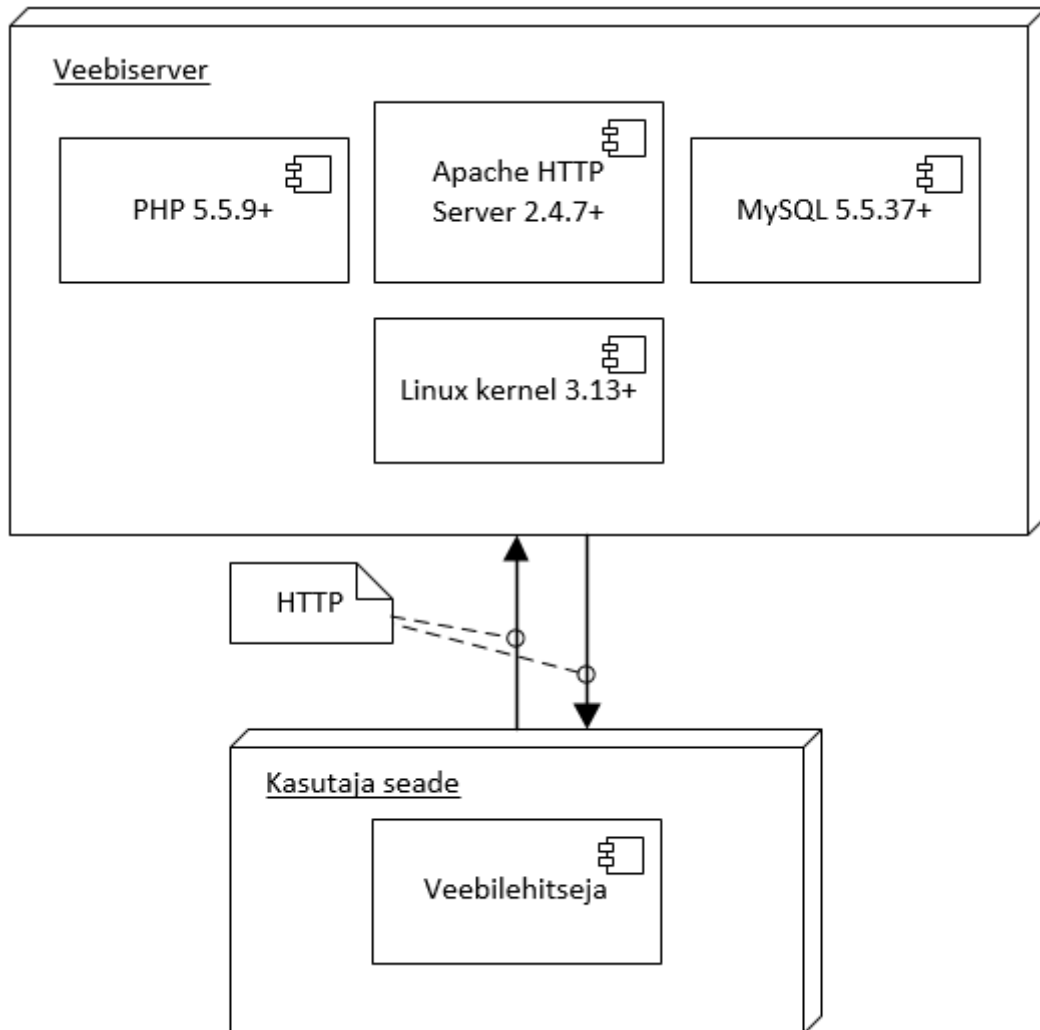
**PHP** – serveripoolne skriptimiskeel, mis on vajalik eelkõige MySQL andmebaasisüsteemiga suhtlemiseks ning dünaamilise sisu kuvamiseks.

**PHPUnit** – PHP skriptikeelele mõeldud testimisraamistik, millega kirjutatakse keskkonna jaoks vajalikud ühiktestid.



## 4.2 Arhitektuur

Keskkonna esialgne arhitektuur on kujutatud joonisel 10.



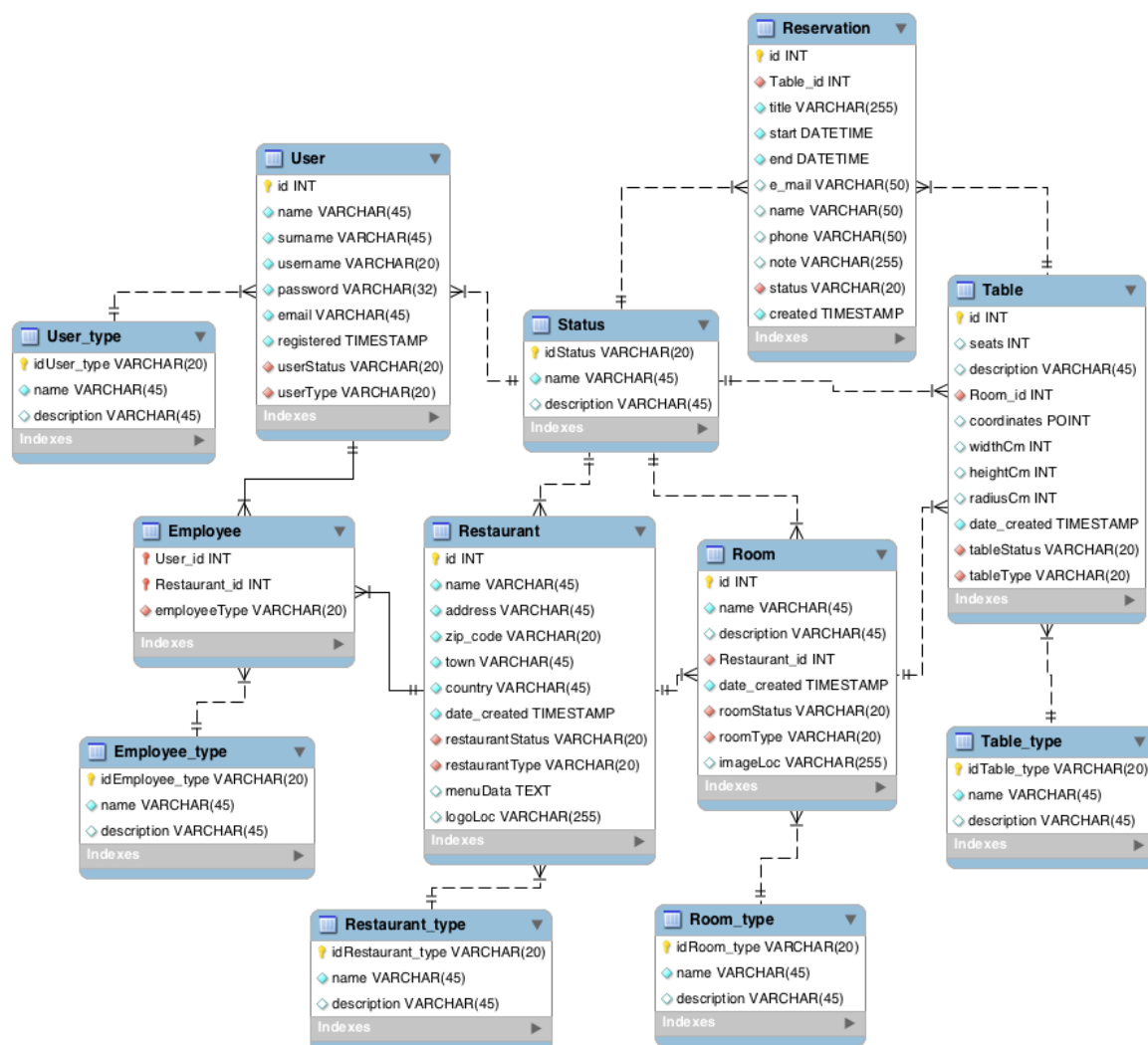
**Joonis 10.** Keskkonna arhitektuuri diagramm

Arhitektuur on üsna lihtne. Veebiserverisse on paigaldatud üsna tüüpiline LAMP tarkvarakomplekt, kuhu kuuluvad Apache Web Server, MySQL, PHP ning mis töötavad mõnel Linux'i distributsioonil. Kogu suhtlus keskkonnaga toimub üle HTTP protokollil.

Kasutaja seadme veebilehitseja peab vastama keskkonna mittefunktsionaalsete nõuete nimekirjas toodud veebilehitseja tingimustele. Kuna suhtlus keskkonnaga toimub üle HTTP protokollil, on kasutajal nõutud ka internetiühendus.

## 4.3 Andmemudel

Järgnevalt on esitatud keskkonna andmemudel koos atribuutide ja nende tüüpidega:



**Joonis 11.** Keskkonna andmemudel koos atribuutide ja nende tüüpidega

Konkreetse andmemudeli eesmärgiks on analüüsist tuleneva funktsionaalsuse andmevajaduste rahuldamine.

Kasutajatega seotud informatsioon salvestatakse nelja tabelisse: *User*, *User\_type*, *Employee*, *Employee\_type*. *User* tabelisse lähevad kõikide kasutajate põhiaandmed, näiteks nimi, perenimi, kasutajanimi, parooli räsiväärtus, e-posti aadress. Lisaks salvestatakse *registered* veergu kasutaja loomise aeg. Veerud *userStatus* ning *userType* sisaldavad välisvõtit, mis viitavad vastavalt tabelitesse *User\_type* ja *Status*. Tabel *User\_type* sisaldab klassifikaatoreid, mis määravad ära kasutajatüübi ja rolli, näiteks administraatori roll või kasutaja roll. *Status*

tabel sisaldab üle keskkonna kasutatavaid klassifikaatoreid, millega saab määrata objektide seisundit, näiteks aktiivne, kustutatud või inaktiivne.

Tabeli *Employee* ülesandeks on siduda omavahel kasutajaid ja ettevõtteid. Lisaks sisaldab tabel veergu, mille väärtused viitavad tabelisse *Employee\_type*. See viide on vajalik selleks, et ettevõttesiseselt saab ära määrata kasutaja õigused ja rolli.

Ettevõtetega seotud informatsioon on salvestatud tabelitesse nimedega *Restaurant*, *Restaurant\_type*, *Room*, *Room\_type*, *Table*, *Table\_type*. Ettevõtte põhilised andmed on salvestatud tabelisse *Restaurant*. Tabel sisaldab ettevõtte nime, aadressi, postindeksit, linna või küla ning riiki, kus ettevõtte asub. Samaselt tabeliga *User*, salvestatakse ka veergu *date\_created* keskkonda sisestamise aeg. Samuti on salvestatud ka *Restaurant* tabelisse toidumenüü info ning viide ettevõtte logole. Veerus *restaurantStatus* on salvestatud välisvõti, mis viitab üle keskkonna kasutatavatele staatuse klassifikaatoritele tabelis *Status*. Veergu *restaurantType* salvestatakse viide tabelisse *Restaurant\_type*, mis sisaldab informatsiooni ettevõtte tüübi kohta, näiteks kas tegemist on sušibaari või Itaalia köögiga.

Tabelisse *Room* salvestatakse ruumi kirjeldavad põhilised andmed – nimetus, kirjeldus ja viide põrandaplaani joonisele. Samuti on tabelis olemas viide ettevõttele, milles see ruum asub ning viide ruumi tüübile (näiteks terrass või siseruum). Kindlasti ei puudu ka tabelist *Room* veerg, mis sisaldab kirje staatust keskkonnas.

Tabelisse *Table* salvestatakse laudadega seotud informatsioon, näiteks laua istumiskohtade arv, laua kirjeldus ning laua asukoha ja suurusega seotud andmed – laua koordinaadid, laua mõõtmed ja raadius. Samuti salvestatakse informatsioon laua kuju kohta tabelis *Table\_type*, millega saab näiteks määrata, kas laud on ristkülik või ümmargune. Informatsioon kuju kohta on tähtis keskkonnas graafiliseks kujutamiseks, et CSSi kujundusega oleks laua kujutamine lihtsustatud.

Broneeringutega seotud andmed salvestatakse tabelisse *Reservation*. Seal on salvestatud viide tabelisse *Table*, mis näitab, millise laua kohta antud broneeringu kirje konkreetselt käib. Lisaks on tabelis kirjas ka mitmesugust broneeringuga seotud informatsiooni, näiteks broneeringu nimi, algusaeg, lõpuaeg. Kindlasti ei puudu tabelist ka broneerija kontaktandmed: e-posti aadress, nimi ning telefoninumber. Kontaktandmed on vajalikud vajadusel broneerijaga ühenduse võtmiseks ning broneerija seostamiseks kasutajaga (juhul, kui broneerija on loonud või loob endale konto keskkonda). Lisaks on olemas *note* veerg, kuhu

broneeriija saab vajadusel sisestada lühikest vabatekstilist kommentaari ja täiendavat informatsiooni broneeringu kohta.

Kokku on esialgse seisuga 12 andmetabelit. Keskkonna funktsionaalsuse kasvamisel või täiendavate andmevajaduste tõttu võib nende tabelite arv veelgi kasvada.

### 4.3.1 Tabelite detailsed kirjeldused

Järgnevalt on detailselt kirjeldatud andmemudelil (Joonis 11) kujutatud andmebaasitabelite veergusid.

**Tabel 2.** Kasutajate tabeli detailne kirjeldus

USER					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
id	integer			Automaatselt genereeritav täisarv, samm = 1.	Jah
name	varchar	45	Eesnimi ei tohi olla tühi string ega tühikutest koosnev string, ei tohi sisaldada numbreid.		Jah
surname	varchar	45	Perenimi ei tohi olla tühi string ega tühikutest koosnev string, ei tohi sisaldada numbreid.		Jah
username	varchar	20	Kasutajanimi ei tohi olla tühi string ega tühikutest koosnev string.		Jah
password	varchar	32	Parooli räsiväärtus.		Jah
email	varchar	45			Jah
registered	timestamp			CURRENT_TIMESTAMP	Jah
userStatus	varchar	20		'active'	Jah
userType	varchar	20			Jah

**Primary Key** (id)

**Foreign Key** (userStatus) REFERENCES Status(idStatus)

**Foreign Key** (userType) REFERENCES User\_type(idUser\_type)

**Tabel 3.** Kasutajatüüpide tabeli detailne kirjeldus

USER_TYPE					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
idUser_type	varchar	20			Jah
name	varchar	45	Nimetus ei tohi olla tühi string ega tühikutest koosnev string.		Jah
description	varchar	45	Kirjeldus ei tohi olla tühi string ega tühikutest koosnev string.		Ei

**Primary Key** (idUser\_type)

**Tabel 4.** Töötaja tabeli detailne kirjeldus

EMPLOYEE					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
User_id	integer				Jah
Restaurant_id	integer				Jah
employeeType	varchar	20			Jah

**Primary Key** (User\_id)

**Alternate Key** (Resturant\_id) REFERENCES Restaurant(id)

**Foreign Key** (employeeType) REFERENCES Employee\_type(id)

**Tabel 5.** Töötajatüüpide tabeli detailne kirjeldus

EMPLOYEE_TYPE					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
idEmployee_type	varchar	20			Jah
name	varchar	45	Nimetus ei tohi olla tühi string ega tühikutest koosnev string.		Jah
description	varchar	45	Kirjeldus ei tohi olla tühi string ega tühikutest		Ei

EMPLOYEE_TYPE					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
			koosnev string.		

**Primary Key** (idEmployee\_type)

**Tabel 6.** Toitlustusettevõtte tabeli detailne kirjeldus

RESTAURANT					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
id	integer			Automaatselt genereeritav täisarv, samm = 1.	Jah
name	varchar	45	Nimetus ei tohi olla tühi string ega tühikutest koosnev string.		Jah
address	varchar	45			Jah
zip_code	varchar	20			Jah
town	varchar	45			Jah
country	varchar	45			Jah
date_created	timestamp			CURRENT_TIMESTAMP	Jah
restaurantStatus	varchar	20			Jah
restaurantType	varchar	20			Jah
menuData	text	65535			Ei
logoLoc	varchar	255			Ei

**Primary Key** (id)

**Foreign Key** (restaurantStatus) REFERENCES Status(idStatus)

**Foreign Key** (restaurantType) REFERENCES Restaurant\_type(idRestaurant\_type)

**Tabel 7.** Toitlustusettevõtete tüüpide tabeli detailne kirjeldus

RESTAURANT_TYPE					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
idRestaurant_type	varchar	20			Jah
name	varchar	45	Nimetus ei tohi olla tühi string ega tühikutest koosnev string.		Jah
description	varchar	45	Kirjeldus ei tohi olla tühi string ega tühikutest koosnev string.		Ei

**Primary Key** (idRestaurant\_type)

**Tabel 8.** Ruumi tabeli detailne kirjeldus

ROOM					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
id	integer			Automaatselt genereeritav täisarv, samm = 1.	Jah
name	varchar	45	Nimetus ei tohi olla tühi string ega tühikutest koosnev string.		Jah
description	varchar	45	Kirjeldus ei tohi olla tühi string ega tühikutest koosnev string.		Ei
Restaurant_id	integer				Jah
date_created	timestamp	45		CURRENT_TIMESTAMP	Jah
roomStatus	varchar	20		'active'	Jah
roomType	varchar	20			Jah
imageLoc	varchar	255			Ei

**Primary Key** (id)

**Foreign Key** (roomStatus) REFERENCES Status(idStatus)

**Foreign Key** (roomType) REFERENCES Room\_type(idRoom\_type)

**Tabel 9.** Ruumitüübi tabeli detailne kirjeldus

ROOM_TYPE					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
idRoom_type	varchar	20			Jah
name	varchar	45	Nimetus ei tohi olla tühi string ega tühikutest koosnev string.		Jah
description	varchar	45	Kirjeldus ei tohi olla tühi string ega tühikutest koosnev string.		Ei

**Primary Key** (idRoom\_type)

**Tabel 10.** Laudade tabeli detailne kirjeldus

TABLE					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
id	integer			Automaatselt genereeritav täisarv, samm = 1.	Jah
seats	integer				Ei
description	varchar	45			Ei
Room_id	integer				Jah
coordinates	point				Ei
width	integer				Ei
height	integer				Ei
radius	integer				Ei
date_created	timestamp			CURRENT_TIMESTAMP	Jah
tableStatus	varchar	20		'active'	Jah
tableType	varchar	20			Jah

**Primary Key** (id)

**Foreign Key** (tableStatus) REFERENCES Status(idStatus)

**Foreign Key** (tableType) REFERENCES Table\_type(idTable\_type)



**Tabel 11.** Lauatüüpide tabeli detailne kirjeldus

TABLE_TYPE					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
idTable_type	varchar	20			Jah
name	varchar	45	Nimetus ei tohi olla tühi string ega tühikutest koosnev string.		Jah
description	varchar	45	Kirjeldus ei tohi olla tühi string ega tühikutest koosnev string.		Ei

**Primary Key** (idTable\_type)

**Tabel 12.** Broneeringute tabeli detailne kirjeldus

RESERVATION					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
id	integer			Automaatselt genereeritav täisarv, samm = 1.	Jah
Table_id	integer				Jah
title	varchar	255			Ei
start	datetime				Jah
end	datetime				Jah
e_mail	varchar	50	E-posti aadress ei tohi olla tühi string ega tühikutest koosnev string.		Ei
name	varchar	50	Nimi ei tohi olla tühi string ega tühikutest koosnev string.		Ei
phone	varchar	50			Ei
note	varchar	255	Märkus ei tohi olla tühi string ega tühikutest koosnev string.		Ei
status	varchar	20			Jah

RESERVATION					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
created	varchar	timestamp		CURRENT_TIMESTAMP	Jah

**Primary Key** (id)

**Foreign Key** (Table\_id) REFERENCES Table(id)

**Foreign Key** (status) REFERENCES Status(idStatus)

**Tabel 13.** Staatuste tabeli detailne kirjeldus

STATUS					
Veeru nimi	Tüübi nimi	Pikkus	Võimalikud väärtused	Vaikeväärtus	Kohustuslik
idStatus	varchar	20			Jah
name	varchar	45	Nimetus ei tohi olla tühi string ega tühikutest koosnev string.		Jah
description	varchar	45	Kirjeldus ei tohi olla tühi string ega tühikutest koosnev string.		Ei

**Primary Key** (idStatus)

## 4.4 Prototüüp

Käesoleva töö autor on osalenud sarnase funktsionaalsusega prototüübi arendamise meeskonnas arendajana. Siinkohal soovib autor avaldada suurt tänu ja tunnustust Tallinna Tehnikaülikooli informaatikatudengile Eerik Mägile ja äriinfotehnoloogia tudengitele Rain Laansalule ja Daniel Leiferile.

Nelja inimese koostööna valmis, kasutades Scrum arendusmetoodikat ning versioonihaldusena Git, sarnast funktsionaalsust omav rakenduse prototüüp. Prototüüp ei realiseeri kogu käesolevas töös esitatud ja väljatöötatud funktsionaalsust. Kuna prototüübi arendamisel puudus dokumentatsioon ja analüüs, on käesoleva töö kirjutamise üks kaudseid põhjuseid ka vajadus eelnimetatud dokumentide järgi.

Lisas 1 on toodud esialgse prototüübi ekraanitõmmised, mille disain saab olema teejuhiks ka uuele arendatavale keskkonnale.

Järgnevalt on kujutatud ekraanitõmmis prototüübist laudade kujutamisest ruumis:

Restoran "Kalleke"

Info

Restorani andmed

Lauad

Laudade asukohad ja broneeringud

Menüü

Töötajad

Kustuta restoran

### Restoran "Kalleke"

#### Laudade asukohtade ja broneeringute haldus

Topeltklõps laual, et näha broneeringuid

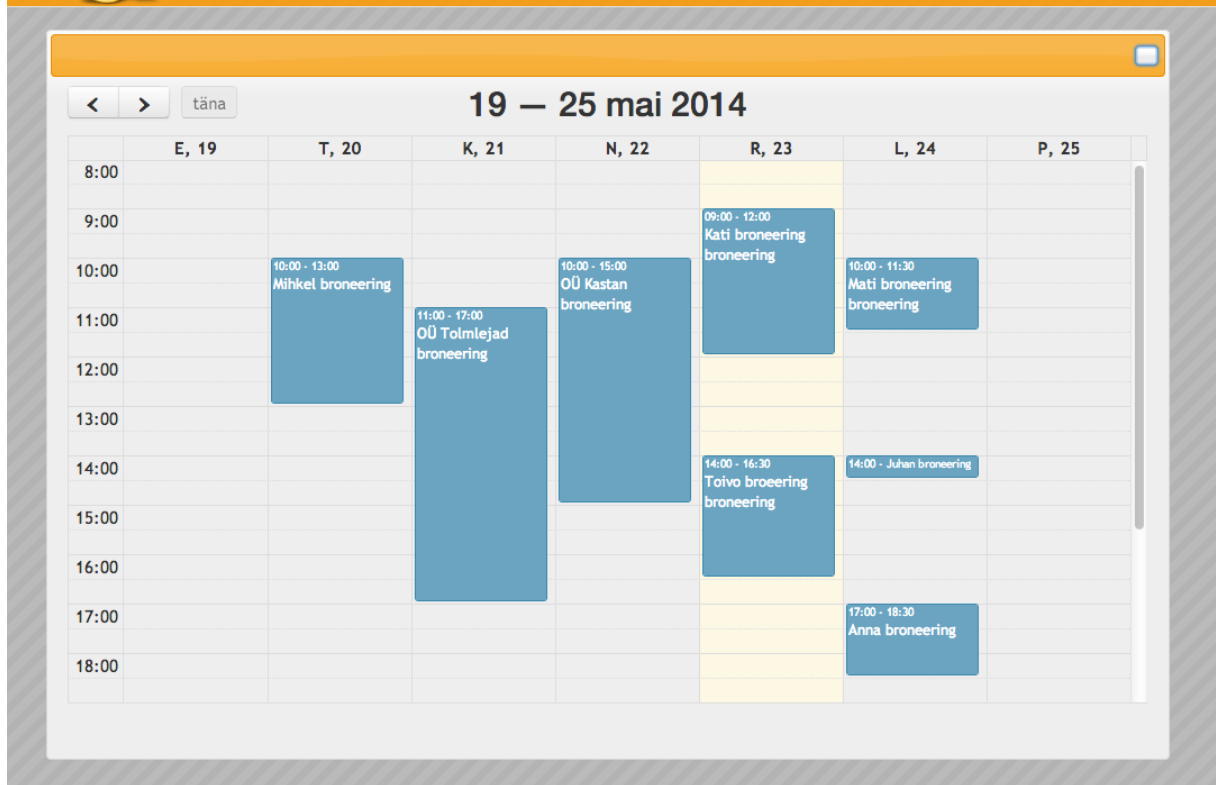
Salvesta paigutus Katkesta

The screenshot displays a restaurant management interface for "Kalleke". On the left is a sidebar menu with options: Info, Restorani andmed, Lauad, Laudade asukohad ja broneeringud, Menüü, Töötajad, and Kustuta restoran. The main area shows a floor plan titled "Laudade asukohtade ja broneeringute haldus" (Management of table locations and reservations). Below the title are two buttons: "Salvesta paigutus" (Save layout) and "Katkesta" (Cancel). The floor plan contains several tables: "No. 4" (8 seats), "No. 3" (6 seats), "No. 1" (2 seats), and "No. 21" (5 seats). There are also decorative plants and a trash can icon on the plan.

**Joonis 12.** Ekraanitõmmis prototüübist, kus saab hallata laudade asukohti ja broneeringuid.

Nagu ka ekraanitõmmiselt (Joonis 12) näha, püüab keskkond muuta laudade kujutamise kasutaja veebilehitsejas võimalikult lihtsaks. Soovitud laua valimisel avaneb konkreetse laua broneeringute kalendrivaade (Joonis 13).

Järgnevalt on näha ühe laua broneeringute vaate ekraanitõmmis:



**Joonis 13.** Valitud laua broneeringute vaatamine prototüübis.

Broneeringute kalendrivaates (Joonis 13) on näha valitud lauale tehtud aktiivsed broneeringud. Kasutaja saab selles vaates sisestada ka enda soovitud broneeringu.

## **5. Võrdlus teiste keskkondadega**

Selles peatükis uuritakse põgusalt teisi internetis sarnast teenust pakkuvaid keskkondi, mis olid leitavad Google otsingumootori abil. Otsingu tulemuseks sai töö autor kaks veebikeskkonda: <http://www.naudi.ee> ja <http://www.vabalaud.ee>.

### **5.1 Naudi.ee**

Naudi.ee reklaamib ennast koduleheküljel järgnevalt: „Enam pole Sul vaja oma restorani laua broneerimiseks kohale minna või telefonitsi jamada – Naudi.ee kaudu saad oma lemmikrestorani laudu broneerida mugavalt läbi interneti. Selleks pole vaja teha muud, kui valida välja meelepärane restoran ja aeg, broneerida laud internetis ning seejärel saadki juba nautida suurepäraseid maitseelamusi parimates restoranides.“ [16]

Broneerimisprotsessi uurides selgub, et tuleb valida restoran, seejärel sisestada einestavate inimeste arv ning valida reserveeringu algusaeg poole tunnise täpsusega. Järgmiseks tuleb kasutajal ennast identifitseerida näiteks läbi sotsiaalvõrgustike ja edasi saab juba broneeringut kinnitada. Sisuliselt tundub tegemist olema veebipõhise vormi täitmisega, misjärel saadetakse info restorani.

### **5.2 Vabalaud.ee**

Vabalaud.ee tutvustab ennast oma koduleheküljel järgnevalt: „Vabalaud.ee portaal võimaldab teha lihtsalt ja kiiresti sobivasse restorani reserveering. Lisaks reserveeringutele saab profiilil kommenteerida toidukohti, avaldada teistele pikemalt arvamust konkreetsest söögikohast ning loomulikult soovitada restorane kui endal puudub nägemus millist kohta valida.“ [17]

Broneerimise protsess on väga sarnane naudi.ee keskkonnale, kus kõigepealt tuleb valida restoran ning seejärel sisestada soovitud suurusega laud ning valida aeg poole tunnise täpsusega. Identifitseerimine otseselt ei ole vajalik, piisab kontaktandmete (nimi, telefoninumber, e-posti aadress) sisestamisest.

### **5.3 Keskkondade puudused ja tugevused**

Peamine puudus eelnimetatud keskkondadel, mis ajendas ka antud lahendust välja töötama, oli broneeringuprotsessi vähene interaktiivsus ning külastaja vähene otsustusvõime protsessi juures. Käesoleva töö poolt välja pakutud lahenduse peamiseks eeliseks on just võimalus valida endale sobiv laud või läbida lihtsustatud broneerimisprotsess.

Mõlema keskkonna tugevuseks on kindlasti külastaja jaoks kuvatavad ettevõtte eripakkumised. Nimelt saab läbi nende keskkondade broneeritud laudade korral näiteks menüüle soodustust või muid eripakkumisi.

Vabalaud.ee tugevuseks on ka kommentaaride kirjutamise võimalus toitlustusettevõtetele. See kindlasti tuleb kasuks teistele külastajatele, kes üritavad leida informatsiooni antud ettevõtte kohta.

Naudi.ee tugevuseks on aga toitlustusettevõtete hinnaklassi näitamine. Hinnaklassi saab määrata otsinguga ning võimalikud väärtused on 1 pallist kuni 4 pallini. Hinnaklassi näitamine on väga hea informatsioon just hinnatundlikule külastajale.

## 6. Võimalikud edasiarendused

Käesoleva töö raames välja töötatud keskkonna peamiseks ülesandeks on toitlustusettevõttesse broneeringu tegemise protsessi kaasajastamine ning mugandamine, mis paneks võimaliku külastaja just protsessi keskmesse ja annaks talle suurema valikuvabaduse pakutavate võimaluste suhtes.

Suurimaks edasiarenduse võimalusteks on integreerida väljatöötatud keskkonda veel erinevat funktsionaalust, mis pakuks rohkem informatsiooni ja võimalusi keskkonna kasutajatele. Näiteks arendada keskkonda lojaalsusprogramm, millega oleks võimalus pikaajalisi kasutajaid siduda ning premeerida keskkonna kasutamise lojaalsuse eest, pakkudes neile läbi punktisüsteemi soodustusi, tasuta söögikordi ning võimalus saada kingitusi. Heaks näiteks on Solarise ja Coca-Cola Plaza kliendi- ja lojaalsusprogrammid vastavalt „Tähtede klubi“ [18] ja „Kinoklubi“ [19].

Teise edasiarendusena võiks lisada külastajatele võimaluse näiteks restorani menüüst toitusid ette tellida ning nende eest ette ära tasuda. Pakutud võimalusel on palju positiivseid külgi – aitab hoida külastaja kulusid kontrolli all; aidata ettevõttel planeerida ressursside efektiivsemat kasutamist; vähendada toidu ooteaega.

Kolmanda edasiarendusena tuleks arendada keskkonna integreerimine mõne ERP rakendusega, kasutades selleks veebiteenuseid, kas SOAP sõnumid, REST teenused või mõni muu tehnoloogia. Näiteks populaarne ja vabavaraline ERP rakendus Odoo (endise nimega OpenERP [20]), kasutab XML-RPC protseduurkütseid andmete vahetamiseks. [21] [22]

Neljanda edasiarendusena võiks keskkonda arendada külastajapoolse tagasisidesüsteemi, kus külastajad saavad anda pärast reserveeringu toimumist konkreetsele ettevõttele hinnanguid, kirjutada arvustusi ja tagasisidet. Sellisel võimalusel oleks palju kasu ettevõtetele, kes selle informatsiooniga töötades saaksid viia sisse muudatusi, et edaspidi oleks külastajakogemus positiivsem ja ettevõtte maine paraneks. Lisaks võimaldaks tagasisidesüsteemiga varustatud keskkond ka kuvada ettevõtte punktiskoori iga ettevõtte nime kõrval. Punktiskoori ja kommentaaride põhjal saab samuti külastaja langetada otsust, millisesse toitlustusettevõttesse planeerida kohtingut, sünnipäevapidu või mõnda muud üritust.

Viiendaks võiks keskkond müüa reklaamikohti ning võimaluse kuvada mõnda ettevõtet otsingutulemustes eespool.



## Kokkuvõte

Käesoleva töö eesmärgiks oli kaasajastada toitlustusettevõttesse laua broneerimise protsessi ning välja töötada seda protsessi realiseeriva keskkonna analüüs ja disain. Tulemuseks saadi dokument, mis sisaldab vajalikku informatsiooni selleks, et hakata keskkonda arendama.

Kõigepealt analüüsiti praegust, peamiselt meili või telefoni teel toimuvat broneerimisprotsessi ning selle põhjal töötati välja uus äriprotsess, mis on tihedalt seotud infotehnoloogiliste võimalustega.

Seejärel töötati välja uut äriprotsessi realiseeriva keskkonna analüüs ja disain, mille peamised eelised on järgnevad:

- külastaja saab mugavalt ja interaktiivselt broneerida laua mõnda toitlustusettevõttesse, samuti on võimalus broneerida ka aega ilma lauda või ruumi valimata, mis tagab protsessi lihtsuse kõikide kasutajate jaoks;
- teenindaja saab mugavalt ja ülevahtlikult näha broneeringute ajakava ja broneeringuid hallata;
- ettevõtte juhataja õigustes olev kasutaja saab lihtsalt hallata oma ettevõttega seotud informatsiooni ning lisada, muuta ja kustutada töötajate kasutajakontosid;
- keskkond võimaldab korraga mitme broneerimisprotsessi läbiviimist;
- administraator saab mugavalt kogu keskkonda administreerida.

Käesolev töö keskendus peamiselt ainult broneerimise protsessi lihtsustamisele ja kaasajastamisele ning vaadeldi ka ettevõtte juhatajapoolseid tegevusi (informatsiooni haldamine, töötajate haldamine jms). Seetõttu on keskkonnale võimalik välja töötada ja arendada hulgaliselt lisafunktsionaalsust, mis keskkonna võimalusi nii külastajatele kui ka ettevõtetele veel kindlasti suurendaksid.

## Summary

The purpose of this thesis was to provide a better solution for catering companies in reservation process. Based on the provided solution, a system analysis and design was developed and documentation was created.

Main advantages of the developed system:

- the customer is now able to reserve a table in a catering company more comfortably and there is also a way to use simple reservation process which involves only selecting reservation time and number of seats;
- waiters are able to conveniently see and manage reservations;
- the manager can easily manage company-related information and employees' user accounts;
- the system is able to serve multiple reservation processes concurrently;
- the administrator can conveniently administrate the whole system.

This thesis focused primarily on simplifying the reservation process. Some of the functionality used by waiters and managers were also observed and analysed. Therefore, it is possible to develop a host of additional functionality for current system to greatly increase the benefits to its users.

## Kasutatud kirjandus

- [1] H. Vallaste, „e-Teatmik: IT- ja sidetehnika seletav sõnaraamat,“ 2014. [Võrgumaterjal]. Saadaval: <http://www.vallaste.ee>. [Kasutatud 22 mai 2014].
- [2] „Bootstrap,“ 2014. [Võrgumaterjal]. Saadaval: <http://www.getbootstrap.com>. [Kasutatud 22 mai 2014].
- [3] Object Management Group, Inc, „BPMN Specification - Business Process Model and Notation,“ 2014. [Võrgumaterjal]. Saadaval: <http://www.bpmn.org>. [Kasutatud 22 mai 2014].
- [4] Software Freedom Conservancy, „Git,“ 2014. [Võrgumaterjal]. Saadaval: <http://git-scm.com>. [Kasutatud 22 mai 2014].
- [5] The jQuery Foundation, „jQuery,“ 2014. [Võrgumaterjal]. Saadaval: <http://www.jquery.com>. [Kasutatud 22 mai 2014].
- [6] The jQuery Foundation, „jQuery UI,“ 2014. [Võrgumaterjal]. Saadaval: <http://www.jqueryui.com>. [Kasutatud 22 mai 2014].
- [7] Ecma International, „The JSON Data Interchange Format,“ Oktoober 2013. [Võrgumaterjal]. Saadaval: <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>. [Kasutatud 22 mai 2014].
- [8] Canonical Ltd, „Ubuntu official documentation,“ 2014. [Võrgumaterjal]. Saadaval: <https://help.ubuntu.com/14.04/serverguide/lamp-overview.html>. [Kasutatud 26 mai 2014].
- [9] S. Bergmann, „PHPUnit - The PHP Testing Framework,“ 2014. [Võrgumaterjal]. Saadaval: <http://www.phpunit.de>. [Kasutatud 22 mai 2014].
- [10] R. T. Fielding, „Representational State Transfer (REST),“ 2000. [Võrgumaterjal]. Saadaval: [http://www.ics.uci.edu/~fielding/pubs/dissertation/rest\\_arch\\_style.htm](http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm). [Kasutatud 27 mai 2014].
- [11] Scrum.org, „What is Scrum?,“ 2014. [Võrgumaterjal]. Saadaval: <https://www.scrum.org/Resources/What-is-Scrum>. [Kasutatud 22 mai 2014].
- [12] M. Fowler, UMLi kontsentraat. 3.readktsioon., Tallinn: Cybernetica AS, 2006.

- [13] Awio Web Services LLC, „W3Counter: Global Web Stats - April 2014,“ 2014. [Võrgumaterjal]. Saadaval: <http://www.w3counter.com/globalstats.php?year=2014&month=4>. [Kasutatud 22 mai 2014].
- [14] A. Shaw, „FullCalendar,“ 2013. [Võrgumaterjal]. Saadaval: <http://www.arshaw.com/fullcalendar/>. [Kasutatud 23 mai 2014].
- [15] Solid IT, „DB-Engines Ranking,“ 2014. [Võrgumaterjal]. Saadaval: <http://db-engines.com/en/ranking>. [Kasutatud 23 mai 2014].
- [16] Booknjoy OÜ, „Naudi.ee,“ 2014. [Võrgumaterjal]. Saadaval: <http://www.naudi.ee/et/kuidastootab>. [Kasutatud 23 mai 2014].
- [17] Vabalaud OÜ, „Vabalaud.ee,“ 2014. [Võrgumaterjal]. Saadaval: <https://www.vabalaud.ee/et/tutvustus>. [Kasutatud 23 mai 2014].
- [18] Solaris Kino, „Mis on tähtede klubi?,“ 2014. [Võrgumaterjal]. Saadaval: <http://www.solariskino.ee/et/klubi/>. [Kasutatud 23 mai 2014].
- [19] Forum Cinemas AS, „Kinoklubi,“ 2014. [Võrgumaterjal]. Saadaval: <http://www.forumcinemas.ee/Club/>. [Kasutatud 23 mai 2014].
- [20] OpenERP S.A., „OpenERP becomes Odoo,“ 2014. [Võrgumaterjal]. Saadaval: <https://www.odoo.com/blog/Odoo-Blog-1/post/Odoo-The-New-OpenERP-156>. [Kasutatud 23 mai 2014].
- [21] OpenERP S.A., „About Us,“ 2014. [Võrgumaterjal]. Saadaval: <https://www.odoo.com/page/about-us>. [Kasutatud 23 mai 2014].
- [22] OpenERP S.A., „XML-RPC Web Services,“ 2014. [Võrgumaterjal]. Saadaval: [https://doc.openerp.com/6.0/developer/6\\_22\\_XML-RPC\\_web\\_services/](https://doc.openerp.com/6.0/developer/6_22_XML-RPC_web_services/). [Kasutatud 23 mai 2014].


# Lisa 1

Apelsin Pealeht Kontakt Administratsioon

Vali restoran, kus soovid laua broneerida.

Filtreeri

Restoran "Kalleke"
Söökla "Pille"
Sushi Baar
Mäkk Barbequu
Häck Bar-B-Que
Kapiano
Sapiano
TTÜ söökla
Sushi hauss



Apelsin is the best thing that ever happened to mankind!

— Someone famous in Source Title

## Joonis 14. Keskkonna prototüübi avaleht

**Pizzaresto**

- Info
- Menüü
- Broneeri laud
- Tagasi

## Pizzaresto

**Pizzaresto**  
 Läti tee 47  
 Morna, Estonia

Restorani tüüp: Itaalia

Logo

Joonis 15. Keskkonna prototüübi ettevõtte informatsiooni vaade

**Pizzaresto**

- Info
- Menüü
- Broneeri laud
- Tagasi

## Pizzaresto

Menüü:

# Pizza

**Pizza Italiana:**

<b>Margarita</b> <i>tomato, cheese</i>	<b>little big</b> <b>2.90 3.55</b>
<b>Prosciutto</b> <i>tomato, cheese, ham</i>	<b>3.85 4.50</b>
<b>Salsiccia</b> <i>tomato, cheese, sausage</i>	<b>3.85 4.50</b>
<b>Tonno e cipolle</b> <i>tomato, cheese, tuna fish, onion</i>	<b>4.15 4.80</b>
<b>Acciughe</b> <i>tomato, cheese, anchovy</i>	<b>4.15 4.80</b>
<b>Vegetariana</b> <i>tomato, cheese, zucchini, egg-blant</i>	<b>3.85 4.50</b>
<b>4 Formaggi</b> <i>tomato, 4 different cheeses</i>	<b>4.15 4.80</b>

Joonis 16. Keskkonna prototüübi ettevõtte toidumenüü vaade