

TALLINNA TEHNIKAÜLIKOOL  
Infotehnoloogia teaduskond

Harri Martin Mölder 179627IADB

# **Toiduvarude planeerimise infosüsteemi analüüs ja arendamine**

Bakalaureusetöö

Juhendaja: Toomas Lepikult  
PhD

Tallinn 2022

## **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: Harri Martin Mölder

31.12.2021

## **Annotatsioon**

Käesoleva bakalaureusetöö eesmärgid on analüüsida toiduvarude raiskamist Eesti kodudes ja kaasa aidata toiduvarude raiskamise vähendamisele, luues selleks veebipõhise infosüsteemi, mis võimaldab retsepte otsida ja toiduvarusid planeerida. Antud infosüsteem võimaldaks kasutajal leida oma valitud toiduainetele kasutusviise toidu valmistamisel või muul viisil ja aidata paremini planeerida vajalike toiduainete ostmist.

Töö esimeses pooles antakse ülevaade toiduainete raiskamisest ja sellest tulenevatest probleemidest globaalselt kui ka Eestis. Analüüsitakse bakalaureusetöö jaoks koostatud küsimustiku tulemusi. Määratakse loodavale süsteemile kriteeriumid ja võrreldakse olemasolevaid sarnaseid infosüsteeme, tuues välja nende head omadused ja puudujäägid.

Töö teises pooles tuuakse välja infosüsteemi loomise käigus kasutatud arenduskeskkonnad ja –vahendid ning kasutatud tarkvaratehnikad. Tuuakse välja loodud andmebaasi mudel. Kirjeldatakse ka loodava süsteemi kasutajalood ja esmast testimise protsessi. Lõpetuseks analüüsitakse loodud süsteemi ja mainitakse võimalikke edasiarendusi.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 32 leheküljel, 7 peatükki, 1 joonist, 12 tabelit.

**Abstract**

**Analysis and Development of a Food Supply Planning  
Information System**

The aims of the given thesis are to analyse food waste in Estonian households and to help prevent food waste in Estonia by developing a web system that allows users to search for recipes and plan the use of their food products. The web system would allow users to find uses for their chosen food items and to help them to better plan the purchase of necessary food products.

The first half of the thesis gives an overview of food waste and the resulting problems it causes both globally as well as in Estonia. It then analyses the results of the questionnaire that was prepared for the thesis. Then the criteria for the web system are established and the strengths and weaknesses of similar existing web systems are analysed.

The second half of the thesis presents the development environments used during the development process as well as the software techniques used when developing the web system. It also provides the database model that the system uses. Then the expected user stories are described as well as the initial minimal viable product testing phase and the resulting changes. Finally, the developed system is analysed, and possible future developments are mentioned.

The thesis is in Estonian and contains 32 pages of text, 7 chapters, 1 figure, 12 tables.

## Lühendite ja mõistete sõnastik

<i>API</i>	Rakendusliides ehk inglise keeles <i>Application Programming Interface</i> Rakendusliides võimaldab määratud reeglistikule vastaval süsteemil edastada käsked ja vahetada andmeid rakendusliideseaga.
<i>Controller</i>	Süsteemi osa, mis tegeleb kasutaja sisendite vastu võtmisega ja töötlemisega ning vastuste tagastamisega.
Eesrakendus	Kasutajapoolne liides ehk inglise keeles <i>front-end application</i> .
<i>HTTPS</i>	Turvaline protokoll, mis võimaldab andmeid edastada arvutivõrkudes. Inglise keeles HyperText Transfer Protocol Secure.
<i>Model</i>	Süsteemi osa, mis haldab infot ja loogikat. Antud kihis kirjeldatakse süsteemi domeeni objektid.
<i>MVC</i>	Inglise keeles <i>Model-View-Controller</i> on tarkvara arhitektuurimuster. Antud muster jaotab loodava rakenduse kolmeks osaks: mudel, vaade ja kontrollid.
Repositoorium	Tarkvaradele mõeldud veebipõhine varamu ehk inglise keeles <i>software repository</i> .
<i>REST</i>	Veebiteenuse ehitust kirjeldavad piirangud. Inglise keeles <i>REpresentational State Transfer</i> .
Süntaks	Programmeerimise keelele vastav reeglite kogum, mis kirjeldab ära millised on õigesti struktureeritud avaldused ja väljendid. Inglise keeles <i>syntax</i> .
Tagarakendus	Süsteemi poolne liides ehk inglise keeles <i>back-end application</i> .
<i>View</i>	Süsteemi osa, mis tegeleb lõppkasutajale info esitamisega.
<i>XML</i>	Masin- ja inimloetav märgistuskeel ehk inglise keeles <i>eXtensible Markup Language</i> .

## Sisukord

1 Sissejuhatus .....	10
2 Teema tutvustus .....	12
2.1 Globaalne probleem.....	12
2.2 Olukord Eestis .....	13
2.3 Toidu raisku minemise põhjused .....	13
2.4 Vastumeetmed .....	14
3 Detailsem ülesandepüstitus.....	15
4 Analüüs.....	17
4.1 Küsimustiku tulemused .....	17
4.2 Loodava süsteemi kriteeriumid .....	18
4.3 Võrdlus sarnaste lahendustega.....	19
4.3.1 Supercook.com .....	19
4.3.2 Myfridgefood.com.....	19
4.3.3 Realfood.tesco.com .....	20
4.3.4 Yummly .....	20
4.3.5 Rimi .....	20
4.3.6 Fitlap.....	21
4.3.7 Erik Orgu .....	21
4.3.8 Võrdlus .....	21
4.4 Arenduskeskkondade ja -vahendite võrdlus .....	22
4.4.1 Java .....	22
4.4.2 Gradle .....	23
4.4.3 Spring .....	24
4.4.4 MariaDB .....	25
4.4.5 VueJS.....	26
4.4.6 Kasutatud tarkvaratehnikad .....	27
4.5 Andmebaasi mudel .....	27
5 Lahendus.....	32
5.1 Kasutajalood .....	32

5.1.1 Toiduainete valimine .....	32
5.1.2 Retseptide otsimine ja filtreerimine.....	32
5.1.3 Retseptide lisamine planeerijasse .....	33
5.1.4 Konto loomine .....	33
5.1.5 Retsepti lisamine .....	33
5.1.6 Toiduainete külmkappi lisamine .....	33
5.2 Esmane testimine ja selle tulemused .....	34
5.2.1 Testimise tulemused .....	34
5.2.2 Testimisest tulenevad muudatused .....	35
6 Tulemused .....	36
6.1 Tulemuste analüüs .....	36
6.2 Loodud süsteem.....	36
6.2.1 Loodud süsteemi vaated .....	37
6.3 Võimalikud edasiarendused.....	38
7 Kokkuvõte .....	40
Kasutatud kirjandus .....	42
Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks .....	44
Lisa 2 – Küsimustik ja vastuse variandid .....	45
Lisa 3 Küsimustiku tulemused .....	47

## **Jooniste loetelu**

Joonis 1. Andmebaasi skeem.....	28
---------------------------------	----



## Tabelite loetelu

Tabel 1. Alternatiivide võrdlus.....	22
Tabel 2. Retsept (Recipe). ....	28
Tabel 3. Toiduaine (Grocery). ....	29
Tabel 4. Retsepti toiduaine (Recipe_Grocery). ....	29
Tabel 5. Toiduaine kategooria (Grocery_Category).....	29
Tabel 6. Toidukord (Meal). ....	29
Tabel 7. Dieet (Diet). ....	29
Tabel 8. Kommentaar (Comment).....	30
Tabel 9. Lemmikud (Favourite). ....	30
Tabel 10. Kasutaja (App_User). ....	30
Tabel 11. Roll (Role). ....	30
Tabel 12. Külmkapp (Fridge_Grocery).....	31

# 1 Sissejuhatus

Käesoleva bakalaureusetöö teemaks on toiduvarude veebipõhise infosüsteemi analüüs ja arendus. Antud teema on aktuaalne, sest Eesti elanikud raiskavad toiduaineid suhteliselt suures ulatuses. Bakalaureusetöös loodav veebipõhine infosüsteem võimaldab toiduainete raiskamist vähendada, aidates kasutajatel oma toidu valmistamist planeerida. Maailmas on sarnaseid lahendusi loodud, kuid eesti keelsena sarnaseid rakendusi praktiliselt pole.

Bakalaureusetöö eesmärgiks on luua veebipõhine infosüsteem, mis võimaldab infosüsteemi kasutajal leida oma olemasolevatele toiduainetele kasutusviise toidu valmistamisel ja aidata kasutajal paremini planeerida toidu valmistamist ja toidukaupade ostmist. Lisaks on eesmärgiks infosüsteemi abil vähendada toiduvarude rikkumist Eesti kodudes.

Loodav veebipõhine infosüsteem peab vastama järgnevatele tingimustele:

- Peab olema mõeldud eraisikutele toiduvalmistamise ja toiduvarude planeerimiseks;
- Peab võimaldama süsteemi kasutajal valida oma olemasolevaid toiduaineid ja otsida ning filtreerida retsepte, võttes arvesse olemasolevaid toiduvarusid;
- Peab võimaldama süsteemi kasutajal luua retsepte;
- Peab võimaldama kasutajatunnusega kasutaja valitud toiduaineid meelde jätta;
- Peab olema kasutatav nii arvutis kui ka mobiiltelefonis läbi veebi.

Bakalaureusetöö koosneb neljast sissejuhatusest. Esimeses peatükis kirjeldatakse toidu raiskamist kui probleemi globaalsel ja Eesti tasandil ning võimalikke lahendusi toidu rikkumise vähendamiseks. Teises peatükis esitatakse täpsem teema tutvustus. Kolmandas peatükis esitatakse bakalaureusetööga seotud analüüsid. Muuhulgas kirjeldatakse lähemalt käesoleva töö autori poolt teostatud küsitluse tulemusi, loodava veebipõhise

infosüsteemi kriteeriume, olemasolevaid sarnaseid infosüsteeme ja infosüsteemi loomisel kasutatud meetodeid ning tehnoloogiaid ja nende võimalikke alternatiive. Kolmandas peatükis esitatakse loodud veebipõhise infosüsteemi lahenduse kirjeldus. Neljandas peatükis kirjeldatakse tehtud töö tulemusi, sealhulgas tulemuste analüüse, loodud süsteemi ja selle võimalikke edasiarendusi.

## 2 Teema tutvustus

Antud peatükis kirjeldatakse lähemalt toidu raiskamist kui probleemi globaalsel ja Eesti tasandil. Samas antakse ülevaade miks lähevad toidukaubad raisku ja mis vastumeetmeid on toidukaupade raiskamise vähendamiseks.

### 2.1 Globaalne probleem

Toidu raiskamine ja sellest tulenevalt toidukaupade riknemine on tõsine globaalne probleem, millel on mitmeid tagajärgi, mis pole koheselt arusaadavad.

Toidu raiskamine on üks tõsine eetiline probleem. Maailmas leidub mitmeid riike, kaasaarvatud arenenud riike, kus märkimisväärsel osal rahvastikust puudub piisav ligipääs toidukaupadele. Ühinenud Rahvaste Organisatsioon arvestab, et üheksast inimesest puudub ühel inimesel normaalseks eluks piisav ligipääs toidukaupadele. Iga aasta sureb ligi 9 miljonit inimest nälga või näljast tulenevatesse haigustesse [1]. Seda on rohkem kui AIDSi, malaariasse ja tuberkuloosi hukkunute arv kokku. Aasta jooksul raisku läinud toiduga oleks võimalik ära toita kogu alatoidetud osa inimkonnast.

Ligi üks kolmandik kogu maailmas toodetud toidukaupadest läheb igal aastal raisku. See on ligi 1,3 miljardit tonni toidukaupasad nagu näiteks liha, puu- ja juurviljad ning paljud muud toidugrupid [2]. Raisku läinud toiduained jõuavad prügilasse, kus neist märkimisväärses koguses metaani eraldub. Atmosfääri jõudnud metaan osaleb keemilistes reaktsioonides ning annab oma panuse kasvuhooneefektile. Samas kõikide toidukaupade tootmine, kaasaarvatud ka raisku minevad toidukaubad, annavad suure panuse süsinikujalajäljele (hinnanguliselt 3,3 gigatonni CO<sub>2</sub>-ekvivalenti) [3]. Kaua aega ei tahtud metaani mõju kasvuhooneefektile arvestada, sest metaan ei püsi atmosfääris pikas perspektiivis väga kaua. Kuid tänaseks on teadlased leidnud, et metaani vähendamine on kriitiline. Võrreldes süsihappegaasiga, on metaani soojendusvõime 80 korda suurem ja see on on 28 korda efektiivsem infrapunakiirguse kinni hoidmisel atmosfääris [4]. Võrreldes süsihappegaasiga ei püsi metaan atmosfääris kaua, ligi 8-10 aastat võrreldes süsihappegaasi 300-1000 aastaga. Peale seda perioodi metaan

oksüdeerub süsihappegaasiks ja veeks. Toidu raiskamist vähendades või isegi vältides, on võimalik ligi 6%-8% inimpõhjustatud kasvuhoonegaasidest vähendada.

Toidu raisku minemine ei ole üksnes keskkonna probleem. Võib öelda, et toidukaup, mis on riknenud, on olnud tühi kulu antud riigile, kus toidukaupa toodeti. Toidukauba tootmiseks kulub aega, vett, energiat ja tööjõudu ning edaspidiseks tuleb toidukaubad vajadusel pakendada ja transportida poodidesse. Kõik see tekitab kulusid ja kui toidukaup ei leia kasutust, siis puudub ka otsene tulemus ja tulu. Globaalselt kulub põllumajanduses 70% inimeste poolt kasutatud veest. Ühe klaasitäie piima tootmiseks, kulub juba ligi 1000 liitrit vett [5]. Riknenud toidukaupade puhul on tegu on tühja kuluga riigile, mida oleks võinud vältida ja millele kulunud ressursid oleks võinud mujale suunata.

## **2.2 Olukord Eestis**

Ka Eestis on toiduainete raisku minemine suur probleem. Aastas tekib Eestis ligikaudu 167 000 tonni toidujäätmeid, millest ligi pool on raisku läinud toidukaup [6]. Antud poole raisku läinud toidukauba väärtuseks on aasta kohta 164 miljonit eurot. Enamik toidukaupasid läheb siiski raisku Eesti kodupidamistes [7]. Üks Eesti leibkond võib aastas ära visata keskmiselt 180 euro väärtuses toitu. Lastega peredes võib summa ulatuda isegi 220-euroni. Kõikidest toidujäätmetest on Eestis ligi 70% tulnud kodumajapidamistest [3]. Samas leidub Eestis ka palju neid inimesi, kellel ei ole piisavalt vahendeid endale toidukaupasid ja esmatarbekaupasid osta.

Ligi neli inimest kümnest viskab igal nädalal toitu ära. Ühes kuus tähendab see 1-5 kilogrammi raisku läinud toidukaupasid, ehk umbes 54 kilogrammi aasta kohta [8]. Üldiselt on inimesed väidetavalt vastutustundlikud ja Eestis on küllalt levinud arvamus, et toiduainete raisku minemine ei ole suur probleem või, et probleemi pole üldse olemaski. Enamasti ollakse nõus, et toidu raisku minemise vähendamiseks peab igaüks oma panuse andma.

## **2.3 Toidu raisku minemise põhjused**

Üheks suuremaks toiduainete raisku minemise põhjuseks on inimeste poolt vähene toidu planeerimine. Enamasti ei planeerita toidu valmistamist ja toidukaupade ostmist ajapuuduse tõttu. Poes käies ostetakse toiduaineid palju kas mugavuse, soodustuste või

kampaaniate tõttu. Toidukaupasad visatakse ära kõige rohkem nende kõlbmatuks muutumise pärast. Ostetud toidukaupasad võib olla liiga palju ning hiljem võib nende olemasolu ununeda ja lõpuks nende avastamisel võib toidukauba tähtaeg juba läbi olla, mistõttu need ära visatakse.

Samas paljud inimesed eelistavad igapäevaselt värskelt valmistatud toitu süüa. Eelmisel päeval valminud ja alles jäänud toit visatakse tihtipeale ära, sest seda ei soovita enam tarbida või ei leita viise selle kasutamiseks. Tihtipeale peetakse ka toidu ülejääke väheväärtuslikuks.

## **2.4 Vastumeetmed**

Nagu sai varem mainitud, on üheks suuremaks toiduainete riknemise põhjuseks vähene planeerimine toidukaupade ostmisel ja vähene toidu valmistamise planeerimine. Toidu valmistamise planeerimine ei pea koheselt olema keeruline tegevus, vaid seda saab väikestes etappides ette võtta. Kui juba jälgida, mida ja kui palju ostetakse ning ära visatakse, siis on võimalik edaspidi neid teadmisi kasutada ostude tegemisel. Edaspidi on võimalik hakata arvestama iganädalase toidu valmistamisega ja milliseid toiduaineid on selle jaoks vaja. Kui tõesti toidu tegemise planeerimiseks aega ei ole, siis on võimalik netist otsida nädala kohta menüüsid, mille järgi oma vajalikud toidukaubad osta.

Toidujäätmete probleemi lahendamiseks ei ole oluline ainult oma toidukaupade ostmist ja toidu valmistamist paremini planeerida, vaid ka enda olemasolevaid toidujääke mõistlikult ära kasutada. Eelnevast päevast alles jäänud toit ei ole koheselt pahaks läinud toit ja seda saab veel ära kasutada. Toidujääke saab uuesti järgmisel toidutegemisel ära kasutada või hoopiski järgnevas päevaks endale kooli või tööle kaasa pakkida.

Samas on võimalik oma toiduaineid või toidujääke annetada, andes sedasi toiduabi selleks vajavatele inimestele. Toidu annetamine pole üldsegi nii keeruline kui võiks arvata ja läbi vahendajate on seda mugav teha. Toidu annetamiseks on kõige parem kasutada selle jaoks mõeldud usaldusväärseid organisatsioone. Näiteks saab toitu annetada lehel [www.toiduannetamine.ee](http://www.toiduannetamine.ee).

### 3 Detailsem ülesandepüstitus

Toiduainete äraviskamine ja seega nende raisku minemine on tänapäeval tõsine probleem. Inimesed üldiselt ei planeeri oma toidu valmistamist ja ostetakse toiduaineid hulgi. Toidu valmistamisest alles või üle jäänud toiduaineid ei osata või ei raatsita ära kasutada. Sedasi visatakse lõpuks toiduained ära, mistõttu lähevad toiduained raisku.

Leidub veebipõhiseid infosüsteeme, mis proovivad aidata kasutajatel leida kasutusviise oma olemasolevatele toiduainetele ja jääkidele. Osad taolised süsteemid isegi lasevad kasutajatel jagada oma toidutegemise mõtteid või retsepte. Samas on neid süsteeme võimalik kasutada oma toidukaupade ostmise planeerimiseks, võttes arvesse kui palju milliseid toidukaupasid toidu tegemisel vaja oleks. Kahjuks ei võimalda suur enamus sarnaseid infosüsteeme kasutaja enda toiduaineid meelde jätta. Probleemne on lisaks see, et sarnased veebipõhised infosüsteemid pole kuigi tuntud ja sarnaseid eesti keelseid infosüsteeme leidub vähe. Enamus olemasolevatest sarnastest eesti keelsetest infosüsteemidest on tasulised.

Eesmärgiks on luua tasuta kättesaadav veebipõhine infosüsteem, mis võimaldaks süsteemi kasutajatel leida oma olemasolevatele toiduainetele kasutusviise toidu valmistamisel või muul viisil. Veebipõhine infosüsteem pakub võimalust kasutajate poolt loodud retsepte otsida ja filtreerida ning samas lisada retsepte planeerijasse, aidates sedasi paremini vajalike toiduainete ostmist planeerida. Süsteem võimaldaks ka lisaks kasutajatunnusega kasutajal retsepte luua ja oma olemasolevaid toiduaineid salvestada. Loodava infosüsteemiga on samas eesmärk aidata kaasa toiduainete riknemise vähendamisele kodudes.

Veebipõhine infosüsteem peab võimaldama retseptide filtreerimist kategooriate järgi, nagu näiteks dieetid või toidukorrad. Samas peab süsteem võimaldama retseptide otsimist, võttes ka arvesse kasutaja poolt valitud toiduaineid. Süsteem peab lisaks võimaldama kasutaja poolt valitud retsepte lisada planeerijasse, et saada täpsemat ülevaadet kõikidest vajalikest toiduainetest ja nende kogustest, mida antud retseptide valmistamiseks vaja läheb. Loodav süsteem peab lisaks võimaldama süsteemi kasutajatel luua oma isiklik konto, millega on võimalik kasutajal oma retsepte süsteemi lisada ja oma olemasolevaid toidukaupasid koos kogustega salvestada. Süsteem peab kasutajatunnusega kasutajal võimaldama retseptide jälgimist ja hindamist.

Loodava veebipõhise infosüsteemi sihtrühmaks on Eesti elanikud, kes tegelevad toidu ostmisega ja valmistamisega.



## 4 Analüüs

Käesolevas peatükis tutvustatakse tööga seotud analüüsi ja nende tulemusi. On kirjeldatud läbiviidud küsitlust ja esitatud selle tulemused. Sõnastatakse loodava süsteemi kriteeriumid, võrdlused sarnaste lahendustega ja töös kasutatud arenduskeskkonnad ja arendusmeetodid ning nende alternatiivid.

### 4.1 Küsimustiku tulemused

Antud bakalaureusetöö osas koostati küsimustik, uurimaks täpsemalt Eesti elanike toidu planeerimist ja allesjäänud toidu rikkemist. Küsimustik koosnes seitsmest valikvastustega küsimusest. Eesmärgiks oli saada vastuseid vähemalt viiekümnelt inimeselt. Küsimustikule vastasid 52 inimest. Küsimustikus esinenud küsimused koos vastuste variantidega leiab käesoleva töö lisa 2.

Küsimustele antud vastustest (esitatud graafiliselt lisa 3) selgus, et peaaegu 30% vastanutest ei kasuta toidu ostmisel kunagi toidulist ja sama paljud inimesed kasutavad toidulist keskmise sagedusega. Kümnendik vastanutest ostab toitu alati koos toidulistiga. Natuke enam kui pooled vastanutest planeerivad toidu tegemist paar päeva ette. Peaaegu 30% vastanutest planeerib toidu tegemist ette terveks nädalaks. Peaaegu pooled vastanutest väitsid, et allahindlused ja kampaaniad mõjutavad nende toidu ostmist väga sageli. Enam kui kolmandikel vastajatest on allahindlustel ja kampaaniatel mõju toidu ostmisele sage. Vastajate seas oli suhteliselt ühe palju neid, kes ostavad toitu tavaliselt ühele inimesele, kahele inimesele, kolmele inimesele ning neljale või enamale inimesele. Ligikaudu 40%-l vastanutest jääb toiduaineid suhteliselt harva üle ja nad ei oska neid toiduaineid ära kasutada. Ligikaudu 30% vastanutest jääb toiduaineid üle keskmiselt sageli. Peaaegu 15%-l vastanutest jääb kodus toiduaineid üle sageli või väga sageli.

Kokkuvõtteks näitasid küsitluse vastused, et keskmiselt sageli, sageli või väga sageli jääb toiduaineid üle ligikaudu 45%-l vastanutest. Sellise olukorra põhjuseks võib muuhulgas olla asjaolu, et inimesed ostavad palju toiduaineid allahindluste ja kampaaniate ajal, sest toidu üle jäämist väitsid valdavalt need vastajad, kes ostavad toiduaineid sageli või väga sageli allahindluste ja kampaaniate mõjul. Küsimustiku vastustest saab järeldada, et käesolevas töös loodav toiduvarude planeerimise veebipõhine infosüsteem on vajalik, sest aitab vähendada toidu raiskamise ja rikkemise probleemi, mis esineb suhteliselt

paljudel inimestel. Samas on selge, et küsimustiku valim on liiga väike, et selle alusel teha suuremaid järeldusi.

## 4.2 Loodava süsteemi kriteeriumid

Antud peatükis räägitakse lähemalt loodavale infosüsteemile seatud kriteeriumitest. Antud kriteeriumid olid ka sarnaste lahenduste võrdlemisel aluseks.

Loodavale infosüsteemile seati järgnevad kriteeriumid:

1. loodav rakendus peab võimaldama süsteemi kasutajal valida toidukaupasid, mille järgi retsepte otsitakse. Valitud toidukaubad on mõeldud olema kasutaja hetkel olemasolevad toidukaubad, kuid võivad samahästi olla ka toiduained, mida kasutaja soovib toidutegemisel kasutada;
2. loodav rakendus peab võimaldama süsteemi kasutajal retsepte otsides neid filtreerida näiteks toidukorra või dieedi alusel. Lisaks peaks süsteemis olema otsingu võimalus, et otsida retsepte nime järgi;
3. loodav rakendus peab retseptide otsingus arvestama kasutaja poolt valitud toidukaupasid ja kuvama retseptide puhul, mis toidukaubad on kasutaja valikus olemas ja mis on puudu;
4. loodav rakendus peab võimaldama süsteemi kasutajal valida mingi arv retsepte ning lisada need retseptide planeerijasse. Retseptide planeerija peab kuvama kasutajale antud retseptide valmistamiseks vajaminevaid koostisosasid kokku;
5. loodav rakendus peab võimaldama süsteemi kasutajal luua omale konto. Loodud kontoga peab kasutaja saama retseptile hinnangut lisada ja retsepti oma lemmikutesse lisada;
6. loodav rakendus peab võimaldama kasutajatunnusega kasutajal luua retsepte ning määrata loodavale retseptile selle valmistamiseks vajaminevaid toiduaineid;
7. loodav rakendus peab võimaldama kasutajatunnusega kasutajal salvestada oma valitud toiduaineid ning võimaldama neid hiljem muuta või eemaldada;
8. loodav rakendus peab olema kasutatav nii arvutis kui ka mobiilis.

### **4.3 Võrdlus sarnaste lahendustega**

Antud peatükis võrreldakse loodava veebipõhise infosüsteemiga juba olemasolevaid sarnaseid infosüsteeme.

Mõningad kriteeriumid, mida võeti arvesse erinevate lahenduste võrdlemisel:

1. saab retsepte otsida toiduainete järgi;
2. saab retsepte filtreerida:
  - a. kasutajal võiks olla võimalik retseptide otsingus filtreerida retsepte dieedi või näiteks toidukorra järgi;
3. saab retsepte lisada;
4. saab retsepte jälgida:
  - a. kasutajatunnuse olemasolul võiks kasutajal olla võimalik lisada retseptid oma lemmikute listi;
5. süsteemi on mugav kasutada ka mobiilis;
6. süsteem võimaldab kasutajal salvestada oma toiduaineid ja neid hallata.

#### **4.3.1 Supercook.com**

Tegu on veebipõhise infosüsteemiga, mis sisaldab endas mitmeid retsepte teistest veebi saitidest. Kasutajal on võimalik toiduaineid külmenüüst lisada või otsida, mille tõttu kuvatakse aina rohkem retsepte, mida kasutaja saab valitud toiduainetega luua. Võimalus on ka filtreerida retsepte, kus kindlasti leidub mingit toiduainet. Retsepti lähemalt uurimiseks tuleb lingist antud veebi sait, kust retsept on võetud. Veebipõhist infosüsteemi on mugav kasutada arvutis. Mobiili kasutajatega pole süsteemi loomisel arvestatud.

#### **4.3.2 Myfridgefood.com**

Tegu on veebipõhise infosüsteemiga, mis võimaldab kasutajal otsida retsepte toiduainete järgi ja luua ka retsepte. Retsepte saab kasutaja luua piiramatu arv, täites retsepti vormi ja saates selle ülevaatuseks. Retsepti täitmisel ei kontrollita, kas kõik vajalikud väljad on

täidetud või mitte. Retseptide otsimisel valib kasutaja toiduained ja saab retsepte otsida. Otsingu vaates on võimalus tulemusi filtreerida. Infosüsteem ei ole väga mugav kasutada, sest lehel leidub rohkelt reklaame ja saidi elemendid ei ole mugavalt ning arusaadavalt paigutatud. Mobiili kasutajatega pole ka arvestatud.

#### **4.3.3 Realfood.tesco.com**

Tegu on Briti suure toidukaubanduskeskuse, Tesco, poolt loodud süsteemiga. Süsteem võimaldab kasutajal otsida süsteemis olemasolevaid retsepte ning filtreerida neid toiduainete kui ka mitme muu välja järgi. Olemasolevate toiduainete järgi filtreerimisel peab kasutaja otsingusse kirjutama ja sealt lisama toiduained, mida retseptid võiksid sisaldada ning olemas on ka võimalus valida koostisosa, mida retseptides võiks kindlasti esineda. Veebi infosüsteemi on mugav kasutada nii mobiilis kui ka arvutis ja reklaame välistades on sisu puhas ja loetav.

#### **4.3.4 Yummly**

Antud veebipõhine infosüsteem võimaldab lisaks retseptide otsimisele oma toidutegemist planeerida, vaadata õpetustega retsepte, lugeda artikleid ja muud. Retsepte otsitakse valitud toiduainete alusel ja tagastatakse retseptid, mida antud toiduainetega saab koheselt luua. Süsteemi on üldiselt lihtne ja mugav kasutada ning mobiili kasutajatega on ka arvestatud. Süsteemi miinuseks on mingil määral tema sisu rohkus. Esmasele saidi külastajale võib arusaamatuks jääda, kus mõningad valikud asuvad. Üheks suureks plussiks on võimalus toiduaineid läbi interneti tellida, kui otsitakse retsepte toiduainete järgi, mida kasutajal ei ole. Toiduainete tellimine töötab ainult kindlates regioonides.

#### **4.3.5 Rimi**

Riia jaemüügioperaator Rimi on loonud oma leheküljel „Tühjenda oma külmik“ süsteemi, mis sarnaneb oma loomult loodava veebipõhise infosüsteemiga. Süsteem võimaldab kasutajal valida mõninga toiduaine vahel, mida arvestades pakub retsepte, mis kasutavad antud toiduaineid. Olles Rimi poolt loodud süsteem, on võimalik retsepti vaates vajalikud toiduained ka Rimi e-poest tellida või välja vahetada. Rimil on ka eraldi retseptide süsteem, kus on võimalik retsepte otsida.

#### **4.3.6 Fitlap**

Fitlap on tasuline toitumiskava süsteem, mis aitab kasutajal oma toitumist tervislikumalt ja paremini planeerida. Fitlapil on loodava süsteemiga sarnasusi, kuid Fitlap keskendub rohkem toitumisele mitte toiduvarude ärakasutamisele. Fitlap võimaldab otsida retsepte märksõna (mis võib olla koostisosa nimetus) ja kategooria järgi, kuid retseptide vaatamiseks on vaja konto luua. Samas on kasutajal võimalik oma retsepte luua, mis on nähtavad ainult retseptide autorile. Retsepte saab maksimaalselt 20 luua. Süsteemi saab limiteeritud aja tasuta kasutada ning peale seda on võimalik osta süsteemi kasutusaega üheks, kolmeks või kaheteistkümneks kuuks. Fitlapil on oma mobiili rakendus ning veebi vaade.

#### **4.3.7 Erik Orgu**

Erik Orgu poolt on olemas tasuline toitumiskava süsteem nii veebis kui ka mobiilis. Süsteem ei sarnane loodava veebipõhise infosüsteemiga kuid on siiski üks alternatiiv retseptide saamiseks. Süsteemi kasutamiseks on vaja konto luua ja telefoni rakenduse kasutamiseks on eraldi vaja tasulist paketti. Süsteem pakub päeva kohta oma loodud menüü ja tasuta kontoga on võimalik tänast ning eelnevate päevade menüüsid vaadata. Retseptide vaatamiseks pole vaja maksta. Retseptide otsimist ja filtreerimist siiski ei võimaldata. Kasutajal on võimalik hankida omale sobiv pakett. Pakettidel on erinevad boonused nagu näiteks treeningkavad, ekspertide nõuanded ja telefonikõne Erik Orguga.

#### **4.3.8 Võrdlus**

Antud alapeatükis on välja toodud alternatiivide uurimisel saadud tulemused eelnevalt mainitud kriteeriumite kohta. Tabelis tähendab „+“ märk kriteeriumile vastavust, „-“ märk kriteeriumile mittevastavust ja „?“ märk tulemuse info puudumist.

Tabel 1. Alternatiivide võrdlus

Süsteem	Saab otsida retsepte	Saab tulemusi filtreerida	Saab retsepte luua	Saab retsepte jälgida	Süsteemi on mugav kasutada ka mobiilis	Võimaldab toiduainete salvestamist ja haldamist
Supercook	+	+	-	+	-	-
Myfridgefood	+	+	+	-	-	-
Realfood	+	+	-	+	+	-
Yummly	+	-	?	+	+	-
Rimi	+	+	-	+	+	-
Fitlap	+	+	+	+	+	-
Orgu	-	-	-	+	?	-

#### 4.4 Arenduskeskkondade ja -vahendite võrdlus

Käesolevas peatükis kirjeldatakse veebipõhise infosüsteemi loomisel kasutatud tehnoloogiaid ja nende alternatiive.

##### 4.4.1 Java

Antud veebipõhise infosüsteemi tagarakenduse (ingl. *back-end application*) loomisel kasutati programmeerimiskeelt Java, mis on mitmeotstrabeline objektorienteeritud (Object-Oriented) programmeerimiskeel, mida kasutatakse nii veebipõhiste infosüsteemide, arvuti rakenduste kui ka Android mobiili äppide väljaarendamisel. Java sai valitud sellepärast, et tegemist on ühe populaarsema ja IT valdkonnas enim nõutuma programmeerimiskeelega ja lõputöö koostaja soovis antud keelega rohkem tutvust teha.

Veebipõhise infosüsteemi tagarakenduse väljaarendamiseks võib kasutada ka muid programmeerimiskeeli. Java ühed tuntumad alternatiivid veebisüsteemi tagarakenduse arendamisel on .NET ja PHP. Väga suur osa maailma ettevõtetest kasutavad oma infosüsteemide tagarakenduste loomisel PHP-d, mis on just suunatud infosüsteemide loomisele. Paljud suured veebipõhised infosüsteemid, nagu näiteks Facebook või Amazon, ehitati PHP põhjal. PHP-l on lisaks loodud raamistikud, mis aitavad süsteemi loomise protsessi lihtsustada ja kiirendada. Sellisteks raamistikeks on näiteks Laravel või Symphony. Üldiselt eelistatakse suuremate projektide puhul kasutada PHP-d, et ei tekiks süsteemi skaleerimisel probleeme. Väiksema projekti puhul on Java kasutamine sobilik ja eelisteks on kohati turvalisem kood, parem programmeerimise liides ning arendamise protsessis kasutatavad silumise tööriistad [9]. .NET, mis loodi sisuliselt Java-ga konkureerimiseks, kasutab keeli VB NET ja C#, mis on oma koodi ja süntaksi (programmeerimise keelele vastav reeglite kogum, mis kirjeldab ära millised on õigesti struktureeritud avaldised ja väljendid. Ingl. *syntax*) poolest üsna sarnane Javaga. Mõlemad keeled on modulaarsed ja pakuvad väga sarnaseid võimalusi. Võrreldes C#-ga on Java vanem ja sellest tulenevalt on ta siiani veebipõhiste infosüsteemide arendamiseks rohkem kasutatud kui C# [10] [11]. Java sai valitud, sest lõputöö koostajal on antud programmeerimiskeelega rohkem kokkupuudet, mis vähendas kohati süsteemi arendamiseks kulunud aega.

#### **4.4.2 Gradle**

Antud veebipõhise infosüsteemi tagarakenduse loomisel kasutati Gradle automatiseerimise tööriista. Gradle sai valitud tema uudsuse, mugavuse ja kiiruse pärast. Samas on lõputöö koostajal Gradle-ga kõige rohkem kokkupuudet.

Gradle on vabavaraline ehituse automatiseerimise tööriist, mis võimaldab luua, erinevates programmeerimiskeeltes, erinevateks otstarveteks projekte. Gradle pakub omalt poolt mitmeid funktsionaalsusi, mis aitavad tarkvara ehitamist lihtsustada, käivitada ja struktureerida. Gradle võimaldab omalt poolt süsteemi jaoks vaja minevate sõltuvuste lihtsat haldamist läbi Maven või Ivy repositooriumite (tarkvarade varamu. Ingl. *software repository*). Samas on üheks peamiseks funktsionaalsuseks Gradle ehitusskript, mis on Groovy programmeerimiskeele põhjal loodud domeenispetsiifiline keel [12]. Gradle on sobilik erineva suurusega projektide arendamiseks. Kasutades Gradle Wrapper'it on võimalik tarkvara komponente käivitada ka Gradle'it installeerimata.

Java puhul on Gradle alternatiivideks näiteks Apache Ant ja Apache Maven, mis on kohati populaarsemad. Antud valikutest on Apache Ant kõige vanem. Ant ehitusskriptid on kirjutatud *XML*-is (masin- ja inimloetav märgistuskeel. Ingl. *eXtensible Markup Language*.) ja iga ehitusfail sisaldab endas ühte projekti. Algselt ei võimaldanud Ant sõltuvuste haldamist, kuid nüüdseks on see lisatud Apache Ivy kaudu. Ant suureks eeliseks on tema paindlikkus. See tähendab, et projektile ei kirjutata ette kindlat struktuuri ega koodimise tavasid. Selline paindlikkus võib muidugi ka miinuseks olla, sest see tähendab, et projektides ei leidu ühtset standardit, kui seda juhuslikult hoida ei proovita. Maven, nagu ka Ant, kasutab *XML*-i oma ehitusfailides. Suureks erinevuseks on see, et Maven vähendab projekti paindlikkust, nõudes kindlat projekti struktuuri. Selline lähenemine võimaldab igas projektis kasutada olemasolevaid käsklusi projekti ehitamiseks. Maven pakub omalt poolt sõltuvuste haldust. Sõltuvused kirjeldatakse ehitusfailis ja seejärel otsib Maven antud sõltuvust kasutaja oma masinast, kohalikust repositooriumist või globaalsest repositooriumist üle interneti. Gradle on mainitud automatiseerimise tööriistadest kõige uuem ja proovib kombineerida Ant ja Maven häid külgi. Gradle pakub omalt poolt paindlikkust, hoides Maven-iga sarnast kindlat struktuuri. Ehitusskriptid ei kasuta *XML*-i vaid domeeni spetsiifilist keelt. Gradle ehitusskript võib hallata ühte projekti või ühte põhiprojekti ja mitut alamprojekti, mis kõik koosnevad oma käsklustest. Tänu Gradle domeeni spetsiifilisele keelele on ehitusfaili sisu palju puhtam ja arusaadavam. Samas jälgib Gradle käsklusi mida on projektis käivitatud ja mis tulemusi antud käsklused on andnud, mis tähendab, et käivitatakse ainult neid käsklusi, mida on vaja uuesti käivitada. See muudab Gradle kohati kiiremaks kui tema alternatiivid [13].

#### **4.4.3 Spring**

Antud veebipõhise infosüsteemi tagarakenduse loomisel kasutati Java jaoks mõeldud Spring Boot raamistiku, koos teiste Spring moodulitega.

Spring on üks populaarseim Java rakenduste ja süsteemide loomiseks mõeldud raamistik. Üks tähtsaim tunnus Spring raamistikul on sõltuvuste süstimine, programmeerimise tehnika, mis võimaldab koodis vajalike sõltuvusi luua ja seejärel sisestada objektidesse, kus neid vaja on. Sõltuvusi sisestatakse näiteks läbi objektide konstruktorite. Sõltuvuste sisestamine vähendab üleliigset koodi ja lihtsustab koodi testimist [14]. Spring Boot on Spring raamistiku üks moodul, mida kasutatakse suuresti veebipõhiste infosüsteemide



loomiseks. Spring Boot vähendab suuresti kirjutatavat koodi, lihtsustades sedasi süsteemi arendamist, ja pakub omalt poolt automaatset konfigureerimist, mida on võimalik hiljem üle kirjutada [15].

Spring Boot alternatiivideks oleks Spring või näiteks Grails. Grails on Spring Boot põhjal loodud raamistik. Võrreldes Spring ja Spring Boot-iga, pole Grails nii levinud, kuid pakub omalt poolt uusi võimalusi, mis on Spring Boot raamistikule juurde ehitatud. Spring Boot sai valitud selle õppematerjalide kättesaadavuse tõttu ja samas on tegu ühe vägagi levinud raamistikuga taoliste süsteemide loomiseks. Spring kasutades toimetab arendaja otse Spring raamistikuga, mis võimaldab paindlikumalt oma keskkonda üles seada. Võrreldes Spring raamistikuga, valiti Spring Boot moodul suuresti arendamise lihtsustamise tõttu. See tähendab, et tehti kompromiss süsteemi paindlikkuse arvel, mis pole otseselt probleem. Loodav süsteem ei ole oma loomult väga unikaalne ega vaja väga täpset korrigeerimist, mistõttu on Spring Boot mooduli kasutamine igati õigustatud [16].

#### **4.4.4 MariaDB**

Antud veebipõhise infosüsteemi andmebaasi haldussüsteemina kasutati MariaDB.

MariaDB on vabavaraline relatsioonilise andmebaasi haldussüsteem, mis loodi MySQL baasil peale seda kui Oracle võttis üle ja ostis ära MySQL 2009. aastal. Selle tõttu on mõlemad andmebaasisüsteemid väga sarnased. Tänapäevaks hoitakse MariaDB arendust MySQL uuendustega kursis, veendumaks, et kõik vajalikud parandused oleks kättesaadavad. MariaDB on klient/server tüüpi andmebaasisüsteem, mis tähendab, et klient rakendus ja server võivad eraldi majutatud olla. Andmete kättesaamiseks tehakse päring serverile, mille peale saadetakse vastuseks soovitud andmed [17].

MariaDB ühtedeks alternatiivideks on MySQL või PostgreSQL. MariaDB on MySQL põhjal loodud, mis tähendab, et mõlemad süsteemid on omavahel sarnased ning kasutavad sama süntaksi. MariaDB loomisel on proovitud MySQL kitsaskohti parendada. Võrreldes MySQL-ga on MariaDB üldiselt kiirem ja võimaldab luua rohkem ühendusi [18]. MariaDB on samas turvalisem, pakkudes meetmeid andmebaasi krüpteerimiseks ja paroolide kontrolliks [19]. PostgreSQL on objekt-relatsiooniline andmebaasi haldussüsteem, mis loodi peale MySQL ning enne MariaDB loomist. Objekt-relatsioonilises andmebaasis hoitakse andmeid objektidena. Sarnaselt objektorienteeritud programmeerimisele võimaldab objekt-relatsiooniline andmebaas kasutada

objektorienteeritud programmeerimises kasutatavaid tehnikaid nagu näiteks pärimist või kapseldamist ja kasutada ka keerukamaid andmestruktuure nagu näiteks liste, kollektsioone ja kaarte [20]. PostgreSQL pakub võrreldes MariaDB-ga paremat andmete kirjutamise ja lugemise kiirust ning on üldiselt võimsam, mistõttu selle kasutamist tihti peale ka eelistatakse. MariaDB sai valitud, sest see on oma mahu poolest väiksem ja paindlikum andmete tüüpide poolest, võimaldades automaatset korrigeerimist, kui proovitakse sisestada valet tüüpi andmeid [21].

#### 4.4.5 VueJS

Antud veebipõhise infosüsteemi esirakenduse (ingl. *front-end application*) loomisel kasutati VueJS raamistikku. VueJS on JavaScripti raamistik, mis on mõeldud üheleheliste (SPA – Single Page Application) veebileidete loomiseks. VueJS sai valitud sellepärast, et lõputöö koostajal on antud raamistikuga kõige rohkem kogemust, mis vähendas esirakenduse loomisele kulutatud aega.

Üheleheline veebirakendus tähendab, et veebilehe sisu serveritakse kasutajale ühe korra ja edaspidine interaktsioon veebilehega on dünaamiline, mis on suhteliselt kiirem võrreldes vanade traditsiooniliste veebirakendustega. Veebilehel uue päringu tegemise ja andmete muutumise puhul serveritakse ainult seda, mis muutus ja mida on vaja kuvada ning veebilehte ei värskendata. Traditsioonilised veebipõhised infosüsteemid serverisid terve veebilehe sisu iga päringu ja andmete muutmise korral. See tähendab, et pärast ühte päringut võis muutuda kogu lehel ainult üks väike osa, kuid selle peale värskendati ja serveriti kogu leht kasutajale uuesti [22]. Dünaamilised ühelehelised veebirakendused on võrreldes vanade traditsiooniliste veebirakendustega märkimisväärselt kiiremad ja paindlikumad.

Üheleheliste veebipõhiste infosüsteemide loomiseks leidub ka muid võimalusi. VueJS ühed suuremad ja tuntumad alternatiivid on React või AngularJS. AngularJS on loodud Google poolt ja on eelnimetatud raamistikust kõige vanem. Võrreldes AngularJS-iga on VueJS uuem ja palju lihtsam ning paindlikum, andes arendajale vabamad käed struktureerida oma kood sedasi nagu soov on. AngularJS oli VueJS arendamisel suuresti inspiratsiooniks, mistõttu VueJS parendab mitmeid AngularJS-is leiduvaid kitsaskohti. React on loodud Facebook poolt ja on ligi aasta vanem kui VueJS. React ja VueJS vahel leidub väga palju sarnasusi ning oma jõudluse ja kiiruse poolest on mõlemad raamistikud piisavalt sarnased, et see valikut suurelt ei mõjutanud. React kasutab suuresti JavaScripti

funktsioone, et komponente oma sisuga välja kuvada. Sellisel lähenemisel on omad plussid, näiteks võimaldab see paremini koodi kontrollida ja muutujate tüüpe piirata [23]. Samas lisab selline lähenemine keerukust ja vajab mingil määral rohkemat õppimist võrreldes VueJS mallide (template) lähenemisega, mis sarnaneb rohkem klassikalisele HTML-ile.

#### **4.4.6 Kasutatud tarkvaratehnikad**

Antud veebipõhine infosüsteem arendati MVC (*Model-View-Controller*) arhitektuuri mustrit kasutades. MVC arhitektuuri mustrit kasutades, jaotatakse süsteem kolmeks omavahel ühendatud osaks: *Model*, *View* ja *Controller*.

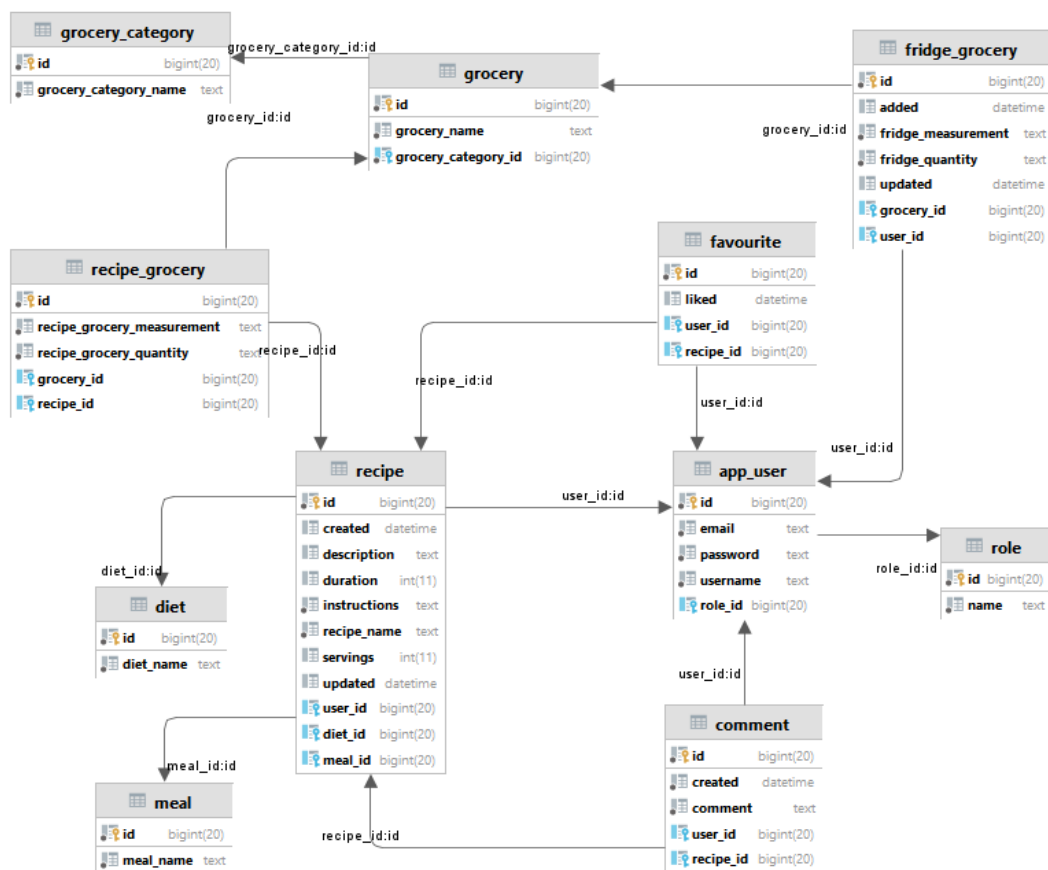
*Model* ehk mudel on süsteemi osa, mis suhtleb andmebaasiga ja tegeleb andmetega. Selles kirjeldatakse domeeni objektid mida hiljem luuakse ja salvestatakse. Üldiselt toimub antud kihis ka salvestamise ajal andmete valideerimine. Igale domeeni objektile luuakse vastavad meetodid, et neid objekte andmebaasi salvestada või neid andmebaasist kätte saada või kustutada.

*View* ehk vaade on süsteemi osa, kus kuvatakse andmeid lõppkasutajale. Antud süsteemi osas luuakse kõik lõppkasutajale kuvatavad vaated ning määratakse nende omavaheline loogika.

*Controller* ehk kontrollor on süsteemi osa, mis tegeleb süsteemi loogikaga. Antud süsteemi kiht tegeleb kasutaja poolt tehtud päringute töötlemisega ja nendele vastamisega. Kasutaja poolt tehtud päringud edastatakse süsteemile vaate kihist ja kontrollor töötleb päringud läbi ja teeb vastavalt oma päringud süsteemi mudeli kihile, et andmeid kätte saada ning siis need vaate kihti edastada [24].

#### **4.5 Andmebaasi mudel**

Käesolevas peatükis kirjeldatakse lähemalt loodud andmebaasi skeemi ja selles olevaid tabeleid ja nende väljasid. Andmebaasi skeemi illustreerib joonis 1 ning tabelite veergude kirjeldused on välja toodud tabelites 2-12.



Joonis 1. Andmebaasi skeem.

Tabel 2. Retsept (Recipe).

Tabeli välja nimi	Välja tähendus
Recipe_Name	Retsepti nimi
Description	Retsepti kirjeldus
Instructions	Retsepti õpetus
Duration	Retsepti kestvus
Servings	Retsepti portsjonid
Created	Retsepti loomise kuupäev

Tabel 3. Toiduaine (Grocery).

<b>Tabeli välja nimi</b>	<b>Välja tähendus</b>
Grocery_Name	Toiduaine nimetus

Tabel 4. Retsepti toiduaine (Recipe\_Grocery).

<b>Tabeli välja nimi</b>	<b>Välja tähendus</b>
Recipe_Grocery_Measurement	Retsepti kuuluva toiduaine mõõteühik
Recipe_Grocery_Quantity	Retsepti kuuluva toiduaine kogus

Tabel 5. Toiduaine kategooria (Grocery\_Category).

<b>Tabeli välja nimi</b>	<b>Välja tähendus</b>
Grocery_Category_Name	Toiduaine kategooria nimetus

Tabel 6. Toidukord (Meal).

<b>Tabeli välja nimi</b>	<b>Välja tähendus</b>
Meal_Name	Toidukorra nimetus

Tabel 7. Dieet (Diet).

<b>Tabeli välja nimi</b>	<b>Välja tähendus</b>
Diet_Name	Dieedi nimetus

Tabel 8. Kommentaar (Comment).

<b>Tabeli välja nimi</b>	<b>Välja tähendus</b>
Comment	Kommentaari sisu
Created	Kommentaari loomise kuupäev
Deleted	Kommentaari kustutamise kuupäev

Tabel 9. Lemmikud (Favourite).

<b>Tabeli välja nimi</b>	<b>Välja tähendus</b>
Liked	Retsepti lemmikutesse lisamise kuupäev

Tabel 10. Kasutaja (App\_User).

<b>Tabeli välja nimi</b>	<b>Välja tähendus</b>
Username	Kontoga seotud kasutajanimi
Email	Kontoga seotud email
Password	Konto seotud parool

Tabel 11. Roll (Role).

<b>Tabeli välja nimi</b>	<b>Välja tähendus</b>
Name	Rolli nimi

Tabel 12. Külmkapp (Fridge\_Grocery)

<b>Tabeli välja nimi</b>	<b>Välja tähendus</b>
Fridge_Quantity	Külmkapis oleva toiduaine kogus
Fridge_Measurement	Külmkapis oleva toiduaine mõõteühik
Added	Külmkappi lisatud toiduaine lisamise kuupäev
Updated	Külmkappi lisatud toiduaine uuendamise kuupäev

## **5 Lahendus**

Käesolev peatükk kirjeldab loodud süsteemi kasutajalugusid ja esmast infosüsteemi testimise protsessi.

### **5.1 Kasutajalood**

Antud peatükis käsitletakse süsteemi kasutajalugusid. Kasutajalood kirjeldavad, mis samme peab süsteemi kasutaja tegema, et mingit kindlast tegevust teha ja soovitud tulemuseni jõuda. Kasutajalood aitavad paremini selgitada infosüsteemi kasutusvoogu ja tihti peale ka selle kaudu avastada uusi kriteeriume infosüsteemi parendamiseks. Välja tuuakse unikaalseid kasutajalugusid, seega korduvaid tegevuskäike välditakse.

#### **5.1.1 Toiduainete valimine**

Antud kasutajalugu kirjeldab täpsemalt (olemasolevate) toiduainete valimise protsessi. Toiduainete valimiseks vajutab kasutaja külghüppel olevale toiduainele, mida soovib lisada valitud toiduainete listi, mistõttu muutub toiduaine „aktiivseks“. Külghüppel on toiduained kategooria järgi eraldatud. Toiduaine eemaldamiseks listist, vajutab kasutaja „aktiivsele“ toiduainele, mistõttu see eemaldatakse listist.

Samas on kasutajal võimalik üleval navigatsiooni reast vajutada „Toiduained“ nupu, peale mille tagajärjel navigeeritakse ta toiduainete vaatesse. Toiduainete vaates saab kasutaja vajutada toiduainele, mida soovib valitud toiduainete listi lisada. Toiduained on jaotatud kategooriate kaupa. Valitud toiduainete listis olevad toiduained on samas ka toiduainete vaates näha ja on eraldatud teistest toiduainetest.

#### **5.1.2 Retseptide otsimine ja filtreerimine**

Antud kasutajalugu kirjeldab täpsemalt retseptide otsimise ja nende filtreerimise protsessi. Retseptide otsimiseks ja filtreerimiseks, navigeerib kasutaja retseptide vaatesse. Antud vaates on kasutajal võimalik otsingusse sisestada retsepti nimi ja valida rippmenüüst toidukord või dieet millele retsept peab vastama. Seejärel „Otsi“ nupule vajutades, kuvatakse talle otsingule vastavad retseptid.



Kui kasutaja on valinud toiduaineid, siis retseptid visualiseerivad, mis toiduained kattuvad valitud toiduainetega ja millised toiduained on retseptis vajalikud aga ei esine valitud toiduainete listis.

### **5.1.3 Retseptide lisamine planeerijasse**

Antud kasutajalugu kirjeldab täpsemalt kuidas kasutaja saab retsepti lisada planeerijasse. Olles retseptide vaates, saab kasutaja lisada retsepti planeerimise listi, vajutades retsepti üleval paremal olevale „+“ märgiga nupule. Vajutades antud nupule, muutub nupp ja kuvab „-“ märki, indikeerimaks, et retsept on listi lisatud ja uuesti nupule vajutades eemaldatakse see listist. Samas on võimalik kasutajal eraldi retsepti vaates lisada antud retsept planeerimise listi, vajutades nupule „Lisa plaani“.

### **5.1.4 Konto loomine**

Antud kasutajalugu kirjeldab täpsemalt kasutaja konto loomise protsessi. Süsteemi kasutaja saab avada üleval paremal oleva konto rippmenüü ning seejärel seal vajutada „Registreeri“ nupule. Nupu vajutuse tagajärjel muutub rippmenüü sisu ja kasutaja saab täita konto loomiseks vastava vormi.

### **5.1.5 Retsepti lisamine**

Antud kasutajalugu kirjeldab täpsemalt kontoga kasutaja retsepti loomise protsessi. Kasutajatunnuse olemasolul saab kasutaja uue retsepti lisada retseptide vaates, vajutades navigatsiooni ribal „Lisa retsept“ peale. Avaneb vaade, kus kasutaja saab sisestada retsepti nime, lisainfo, õpetuse ja määrata kestvuse ning portsjonite arvu. Samas saab kasutaja lisada retsepti jaoks vajalike toiduaineid ja määrata neile koguse koos mõõteühikuga.

### **5.1.6 Toiduainete külmkappi lisamine**

Antud kasutajalugu kirjeldab täpsemalt kuidas kasutajatunnusega kasutaja saab toiduaineid oma külmkappi lisada. Kasutajatunnuse olemasolul saab kasutaja salvestada oma olemasolevaid toiduaineid, vajutades navigatsiooni ribal „Külmkapp“ valiku peale. Avaneb vaade, kus kasutaja saab sisestada oma olemasolevad toiduained koos nende kogustega või juba sisestatud toiduaineid muuta või eemaldada.

## 5.2 Esmane testimine ja selle tulemused

Antud peatükis kirjeldatakse lähemalt loodava veebipõhise infosüsteemi minimaalse töötava produkti testimise faasi ja selle tulemusi. Testimine viidi läbi kontrollitult nelja kasutaja vahel.

Kasutajatel paluti testida ja arvamust avaldada järgnevate punktide kohta:

- toiduainete lisamine valitud toiduainete listi ja toiduainete otsimine;
- retsepti loomine kasutajaga;
- retseptide otsimine filtrite ja toiduainete abil;
- retseptide planeerimine;
- süsteemi üldine kasutatavus;
- mobiilne kasutatavus.

### 5.2.1 Testimise tulemused

Süsteemi testimise käigus paluti kasutajatel lisada toiduaineid valitud toiduainete listi. Valitud toiduainete listi prooviti toiduaineid lisada vasakult leiduvast külgnenüüst, kui ka eraldi toiduainete vaatest. Kasutajate jaoks oli soovitud toiduainete listi lisamine lihtne ja arusaadav. Samuti oli võimalik toiduaineid otsida ja sedasi lisada neid oma listi. Ebamugav oli kasutajate jaoks ainult mitme või isegi kõikide toiduainete eemaldamine listist.

Süsteemi testimise käigus paluti kasutajatel luua retsept. Retsepti loomine ja retsepti väljad olid kasutajate poolest üldiselt arusaadavad. Probleemseks jäi toiduainete lisamine retsepti ja samas arusaamine, kui vähe või palju teksti oli lahtritesse võimalik kirjutada või kui väikese või suure numbriga võis lahtrisse sisestada. Peale retsepti loomist suunati kasutaja loodud retsepti vaatesse ning kasutajad leidsid hiljem ka retsepti „Minu retseptid“ vaatest üles.

Testimise käigus paluti kasutajatel otsida ja filtreerida retsepte. Retsepte paluti kasutajatel filtreerida toidukordade ja ka dieetide järgi. Leiti, et otsimise ja retseptide filtreerimise protsess oli arusaadav. Kasutajad avaldasid arvamust, et süsteemis oli hea valitud

toiduainete kuvamine retseptide vaates. Retseptis olevad toiduained, mis vastasid kasutaja valitud toiduainete listis olevate toiduainetega, värviti rohekaks ja toiduained, mis listis ei olnud värviti punaseks, teavitades kasutajale, et antud toiduainet oleks vaja hankida.

Testimise käigus paluti kasutajatel valida paar retsepti ning lisada need toidu planeerijasse. Retsepti lisamine planeerijasse oli kasutajate jaoks kerge. Kõikide retseptide vaates, kui ka eraldi retsepti vaates on planeerijasse lisamiseks nupp olemas. Enamus testijaid mainisid, et alguses ei olnud kõikide retseptide vaates iga retsepti juures olev pluss nupp piisavalt selge, kuid kui saadi funktsionaalsusest aru, siis oli nupu olemasolu väga mugav. Planeeriija vaatesse said kasutajad minna kasutades külghenüüd. Planeerijas olev info oli kasutajate jaoks mõistetav.

### **5.2.2 Testimisest tulenevad muudatused**

Valitud toiduvarude suures koguses eemaldamiseks sai lisatud külghenüüsse ja eraldi toiduainete vaatesse nupp, mis võimaldab kogu valitud toiduvarude listi tühjendada. Antud nupp tuleb alles siis nähtavale, kui valitud toiduainete listi on vähemalt üks toiduaine lisatud.

Retsepti loomisel oli probleemseks vormi väljade valideerimine ja info edastamine kasutajale. Tavakasutajal on ebamugav näha veateateid või veelgi hullemal juhul teavitusi mitte saada, mistõttu oleks vajalik kasutajale märku anda, mis väljad pole õigesti täidetud, et süsteemi kasutaja teaks kuhu ja milliseid parandusi tuleb teha. Samas koormab süsteemi vigaste päringute tegemine ja sellest tulenevalt on hea teha päring alles korrektselt täidetud väljade puhul.

Samuti tehti süsteemi välimuse osas muudatusi, et kasutajate kogemus oleks mugavam ja süsteem arusaadavam. Muudatusi tehti ka mobiilse vaate osas.

## 6 Tulemused

Antud peatükis on välja toodud tehtud töö analüüs ja loodud veebipõhise infosüsteemi võimalikud edasiarendused.

### 6.1 Tulemuste analüüs

Bakalaureusetöö eesmärgiks oli luua veebipõhine infosüsteem, mis võimaldaks kasutajatel luua retsepte, otsida ja filtreerida retsepte, arvestades ka kasutaja poolt valitud toiduaineid, jälgida retsepte ja anda retseptidele hinnangut.

Bakalaureusetöö eesmärk sai täidetud. See tähendab, et loodi veebipõhine infosüsteem, mis võimaldab süsteemi kasutajatel leida oma olemasolevatele toiduainetele kasutusviise toidu valmistamisel. Täidetud sai ka alaeesmärk, mis tähendab, et loodud süsteem aitab kaasa toiduainete riknemise vähendamisele kodudes. Vastavalt soovitud loodi veebipõhine infosüsteem, mis võimaldab retseptide filtreerimist kategooriate järgi, retseptide otsimist, retseptide lisamist planeerijasse, retseptide loomist, retseptide jälgimist ja nende hindamist.

### 6.2 Loodud süsteem

Antud peatükis kirjeldatakse loodud süsteemi struktuuri ja andmete pärimise ning liikumise loogikat. Loodud süsteemi lähtekood asub Bitbucket keskkonnas [25].

Antud veebipõhine infosüsteem koosneb tagarakendusest ja eesrakendusest, mis suhtlevad omavahel REST (ingl. *REpresentational Sstate Transfer*) arhitektuuri abil. *REST* on arhitektuuri stiil veebipõhiste infosüsteemide loomiseks, mis koosneb kindlatest piirangutest, mille täitmise puhul saab nimetada süsteemi *RESTful* süsteemiks [26]. Veebipõhise infosüsteemi loomisel püüti lähtuda hea tava järgi, hoides süsteemi andmed, äriloogika ja vaated teineteisest lahus. Antud tavast peeti kinni nii tagarakenduse kui ka eesrakenduse loomisel. Hoides süsteemi osad teineteisest eraldi on tulevikus kergem süsteemile juurde ehitada ja seda laiendada.

Loodud veebipõhise infosüsteemi tagarakendus on kirjutatud kasutades Java programmeerimiskeelt koos Gradle automatiseerimise tööriista ja Spring Boot

raamistikuga. Süsteemi tagarakendus vastab *REST* nõuetele ja on seega loomult *RESTful API* (ingl. *Application Programming Interface*). Lisaks kasutati . on jaotatud järgnevateks kihtideks:

- olemite kiht (inglise keeles *Entity Layer*);
- andmete ligipääsukiht (inglise keeles *Data Access Layer*);
- teenusekiht (inglise keeles *Service Layer*);
- esituskiht (inglise keeles *Presentation Layer*).

Loodud veebipõhise infosüsteemi eesrakendus kasutab VueJS raamistikku ja Vuex liidest, et süsteemi olekut hallata. Vuex võimaldab luua globaalse instantsi süsteemi oleku muutmiseks ja meelde jätmiseks, mida on võimalik Vue komponentides hiljem viidata ning kasutada. Tagarakendusega suhtlemiseks kasutatakse *HTTPS* (ingl. *HyperText Transfer Protocol Secure*) protokoll. Loodud eesrakendus on jaotatud järgnevateks kihtideks:

- teenusekiht (inglise keeles *Service Layer*);
- äriloogika kiht (inglise keeles *Business Layer*);
- vaate kiht (inglise keeles *UI Layer*).

Süsteemi jaotamine erinevateks kihtideks aitab süsteemi komponente teineteisest selgelt eristada, tekitades kindlamat struktuuri. Süsteemi kihiline arhitektuur muudab süsteemi loomise protsessi ja hilisema haldamise mugavamaks ja kergemaks.

### **6.2.1 Loodud süsteemi vaated**

Antud peatükis loetletakse süsteemi jaoks loodud erinevad vaated.

- Üldvaade.
- Retseptide, lemmik retseptide ja loodud retseptide vaade.
- Retsepti loomise vaade.
- Retsepti vaade.

- Planeerimise vaade.
- Toiduainete vaade.
- Külmkapi vaade.
- Administraatori haldus vaated.

### **6.3 Võimalikud edasiarendused**

Süsteemi loomise käigus tuli mitmeid mõttekohti, kuidas annaks loodavat süsteemi parendada ja tulevikus veel edasi arendada.

Üks mahukas ja kasutajale potentsiaalselt väga mugav edasiarendus oleks toiduainete tellimise süsteemi juurde lisamine. Sellise mooduli juurde lisamine vajab kindlasti koostööd toidukaubanduskeskustega ja lisaanalüüsi. Antud moodul võimaldaks kasutajal valida kõik retseptis leiduvad toiduained või ainult retseptis puuduolevad toiduained ja need oma ostukorvi lisada. Samas võiks selline moodul töötada ka olemasoleva toidu planeerimise mooduliga. Ostukorvis olevaid toiduaineid võiks saada tellida kasutaja poolt valitud toidupoesst valitud kuupäevaks ja toiduainete kättesaamise viisideks võiksid olla kohale toimetamine või postiautomaatide kasutamine. Samas võiks kasutajal olla võimalus enda konto ühendada antud toidupoodide kontodega, et saada toidukaupade tellimisel vastavaid soodustusi ja punkte. Selline edasiarendus vajab palju lisatööd, et tellimise protsess oleks mugav ja kindlasti ka turvaline.

Süsteemi kasutajatele võib veel plussiks olla toidutegemise plaanide jagamise võimalus foorumi või artiklite kujul. Ideena oleks tegu juba olemasoleva retseptide planeerimise mooduliga, mida oleks võimalik avalikustada koos mõne kirjeldusega. See annaks võimaluse kasutajatel luua näiteks nädalasi toidu menüüsid, mida teised saavad järgi proovida. Samas oleks võimalik inimestel jagada ideid toidu menüüdest, mille koostamiseks ei pea väga palju toidukaupasid ostma. Selline lisalahendus annaks võimaluse nendele, kellel on vähem aega retsepte planeerida, vaadata mida teised on loonud ja neid ideid kasutada.

Üheks edasiarenduseks ja mõttekohaks on ka loodud veebipõhise infosüsteemi kättesaadavaks tegemine teistes riikides ja keeltes (näiteks Soome, Läti, Leedu). See

annaks võimaluse rohkematele inimestele lisada retsepte ja leida oma toiduainetele kasutust. Antud töö eesmärgiks oli küll taolise süsteemi väljaarendamine Eestis kasutuseks. Selline edasiarendus tähendab kindlasti tõlgete lisamist süsteemi ja ka võimalust lasta tõlkida retsepte erinevatesse keeltesse. Võimalus oleks ka lasta teatud kasutajatel selliseid tõlkeid soovi korral luua ja ka kasutajatel loodud tõlgete jaoks parandusi paluda. Oleks tarvis analüüsida kas ja kui mõistlik oleks sellist lahendust luua ja kui kulukaks võib selle loomine muutuda.

Tänapäeval leidub uuematel külmkappidel sisse ehitatud funktsionaalsusi, mis lasevad inimesel toiduaineid süsteemi sisestada või neid sisse skaneerida. Külmkappi sisestatud andmed on seejärel kättesaadavad vastavast mobiilirakendusest, et inimesel oleks mugav poes üle vaadata, mis toidukaup on tal hetkel olemas ja isegi, kui kaua mõni toiduaine on külmkapis olnud. Selliste lahenduste integreerimiseks on vaja teha palju ekstra tööd, et kõik soovitud rakendused saaksid töötada loodud süsteemiga. Sedasi ei pea süsteemi kasutaja pidevalt oma olemasolevaid toiduaineid süsteemis ära märkima ega salvestama, vaid need saadakse kätte teisest rakendusest ja ühtlustatakse loodud süsteemiga, kui kasutaja oma külmkappi vastava toiduaine lisab. Selliste olemasolevate lahenduste ära kasutamine annaks kindlasti loodud süsteemile väärtust juurde ning lihtsustaks kasutaja toiminguid.

## 7 Kokkuvõte

Käesoleva bakalaureusetöö teemaks oli toiduvarude infosüsteemi analüüs ja arendamine. Bakalaureusetöö eesmärgiks oli luua veebipõhine infosüsteem, mis võimaldab süsteemi kasutajal leida oma olemasolevatele toiduainetele kasutusviise toidu valmistamisel ja aidata kasutajal paremini oma toiduvalmistamist ja toidukaupade ostmist planeerida. Samas oli eesmärgiks loodud infosüsteemi abil vähendada toiduvarude riknemist Eesti kodudes.

Töö jaoks koostati küsimustik, millele andsid vastuseid kokku 52 inimest. Küsimustiku valim oli seega liiga väike, et suuremaid järeldusi teha, kuid küsitluse vastustest oli näha, et märkimisväärsel osal vastajatest jääb toiduaineid sageli üle. Töö käigus loodi veebipõhine infosüsteem, mis vastab seatud eesmärkidele:

- süsteem on mõeldud eraisikutele toiduvalmistamise ja toiduvarude planeerimiseks;
- süsteem võimaldab süsteemi kasutajal valida oma olemasolevaid toiduaineid ja otsida ning filtreerida retsepte, võttes ka arvesse olemasolevaid toiduvarusid;
- süsteem võimaldab süsteemi kasutajal luua retsepte;
- süsteem võimaldab kontoga kasutajal valitud toiduaineid meelde jätta;
- süsteem on kasutatav nii arvutis kui ka mobiilis läbi veebi.

Maailmas, sealhulgas ka Eestis, on loodud mitmeid sarnaseid toiduga ja toiduvarudega seotud infosüsteeme, kuid erinevate infosüsteemide võrdlemisel selgus, et võrreldavad süsteemid ei sarnane täpselt käesoleva töö raames loodud infosüsteemiga. Tähtsamateks tunnusteks on loodava süsteemi puhul toiduvarude järgi retseptide otsimine, kasutaja olemasolevate toiduainete salvestamine ja muutmine, retseptide lisamine ja nende hindamine ja samas on loodud süsteem tasuta kättesaadav.

Kokkuvõtteks sai töö alguses püstitatud eesmärk töö käigus täidetud.



Infosüsteemi loomise käigus tuli ka mitmeid mõttekohti, kuidas annaks loodavat süsteemi tulevikus täiustada ja edasi arendada. Ühtedeks mahukamateks ja kasutajale potentsiaalselt väga mugavateks edasiarendusteks oleks toiduainete tellimise süsteemi juurde lisamine ja integratsioon tarkade külmkappide ja teiste lahendustega, et kasutaja olemasolevate toiduainete sisestamist mugavamaks muuta.

## Kasutatud kirjandus

- [1] „People who died from hunger“, The World Counts, [Võrgumaterjal]. Loetud aadressil <https://www.theworldcounts.com/challenges/people-and-poverty/hunger-and-obesity/how-many-people-die-from-hunger-each-year/story>. [Kasutatud 10 11 2021].
- [2] „Fight climate change by preventing food waste“, World Wild Life, [Võrgumaterjal]. Loetud aadressil <https://www.worldwildlife.org/stories/fight-climate-change-by-preventing-food-waste>. [Kasutatud 09 11 2021].
- [3] „Toidujäätmete tekke ennetamine ja toidu päästmine“, Sotsiaalministeerium, [Võrgumaterjal]. Loetud aadressil <https://www.sm.ee/et/toidujaatmete-annetamine>. [Kasutatud 10 11 2021].
- [4] „Methane: A crucial opportunity in the climate fight“, Environmental Defense Fund, [Võrgumaterjal]. Loetud aadressil <https://www.edf.org/climate/methane-crucial-opportunity-climate-fight>. [Kasutatud 09 11 2021].
- [5] „The Environmental Impact of Food Waste“, Move For Hunger, [Võrgumaterjal]. Loetud aadressil <https://moveforhunger.org/the-environmental-impact-of-food-waste>. [Kasutatud 09 11 2021].
- [6] „Toidujäätmed“, Rimi, 2021. [Võrgumaterjal]. Loetud aadressil <https://www.rimi.ee/uudised/toidujaatmed>. [Kasutatud 08 11 2021].
- [7] J. Voltri, „Eestis visatakse igal aastal ära ligi 164 miljoni euro eest toitu“, Eesti Rahvursinghääling, 2021. [Võrgumaterjal]. Loetud aadressil <https://www.err.ee/1608229263/eestis-visatakse-igal-aastal-ara-ligi-164-miljoni-euro-eest-toitu>. [Kasutatud 12 11 2021].
- [8] M. Mihkels, „Kohutav raiskamine: kolmandik Eesti inimestest viskab iga nädal ära kuni viis kilo toitu“, Rahageenius, [Võrgumaterjal]. Loetud aadressil <https://raha.geenius.ee/rubriik/uudis/kohutav-raiskamine-kolmandik-eesti-inimestest-viskab-iga-nadal-ara-kuni-viis-kilo-toitu/>. [Kasutatud 12 11 2021].
- [9] O. Melnic, „Java Vs. PHP: What To Choose In 2021“, Hackernoon, [Võrgumaterjal]. Loetud aadressil <https://hackernoon.com/java-vs-php-what-to-choose-in-2021-6t1133px>. [Kasutatud 26 11 2021].
- [10] M. Nehra, „Java vs .NET for Web Application Development“, Decipherzone, [Võrgumaterjal]. Loetud aadressil <https://www.decipherzone.com/blog-detail/Java-vs-.NET-for-Web-Application-Development>. [Kasutatud 2021 11 26].
- [11] T. B. Pham, „Java or .NET For Web Application Development“, Saigon Technology, [Võrgumaterjal]. Loetud aadressil <https://saigontechnology.com/blog/java-or-net-for-web-application-development>. [Kasutatud 26 11 2021].
- [12] „What is Gradle?“, Gradle, [Võrgumaterjal]. Loetud aadressil [https://docs.gradle.org/current/userguide/what\\_is\\_gradle.html](https://docs.gradle.org/current/userguide/what_is_gradle.html). [Kasutatud 16 11 2021].

- [13] R. Bhadauria, „ANT vs Maven vs Gradle“, Medium, [Võrgumaterjal]. Loetud aadressil <https://medium.com/@257ramanrb/ant-vs-maven-vs-gradle-cd8ab4c2735f>. [Kasutatud 16 11 2021].
- [14] B. Karia, „A quick intro to Dependency Injection: what it is, and when to use it“, FreeCodeCamp, [Võrgumaterjal]. Loetud aadressil <https://www.freecodecamp.org/news/a-quick-intro-to-dependency-injection-what-it-is-and-when-to-use-it-7578c84fa88f/>. [Kasutatud 26 11 2021].
- [15] „Spring vs. Spring Boot vs. Spring MVC“, JavaTpoint, [Võrgumaterjal]. Loetud aadressil <https://www.javatpoint.com/spring-vs-spring-boot-vs-spring-mvc>. [Kasutatud 26 11 2021].
- [16] „Java Spring Boot“, International Business Machines Corporation, 2020. [Võrgumaterjal]. Loetud aadressil <https://www.ibm.com/cloud/learn/java-spring-boot>. [Kasutatud 26 11 2021].
- [17] R. Saive, „What is MariaDB? How Does MariaDB Work?“, Tecmint, [Võrgumaterjal]. Loetud aadressil <https://www.tecmint.com/what-is-mariadb-how-does-mariadb-work/>. [Kasutatud 24 11 2021].
- [18] R. Peterson, „MariaDB vs MySQL: What is the Difference Between MariaDB and MySQL“, Guru99, [Võrgumaterjal]. Loetud aadressil <https://www.guru99.com/mariadb-vs-mysql.html>. [Kasutatud 24 11 2021].
- [19] „3 Benefits of Using MariaDB Over MySQL“, A2 Hosting, 2020. [Võrgumaterjal]. Loetud aadressil <https://www.a2hosting.com/blog/mariadb-vs-mysql/>. [Kasutatud 24 11 2021].
- [20] „Difference Between Object Oriented Database and Object Relational Database“, Pediaa, 2019. [Võrgumaterjal]. Loetud aadressil <https://pediaa.com/difference-between-object-oriented-database-and-object-relational-database/>. [Kasutatud 2021 11 25].
- [21] „MariaDB vs PostgreSQL: Know the difference between the databases“, OptimizDBA, [Võrgumaterjal]. Loetud aadressil <https://optimizdba.com/mariadb-vs-postgresql-know-the-difference-between-the-databases%EF%BB%BF/#:~:text=PostgreSQL%20supports%20JSON%20and%20JSONB,alias%20for%20JSON%2C%20unlike%20PostgreSQL.&text=MariaDB%20has%20a%20considerably%20smaller,is%20short%20o>. [Kasutatud 24 11 2021].
- [22] K. Lawson, „What Is a Single Page Application and Why Do People Like Them so Much?“, Bloomreach, [Võrgumaterjal]. Loetud aadressil <https://www.bloomreach.com/en/blog/2018/07/what-is-a-single-page-application.html>. [Kasutatud 31 10 2021].
- [23] „Comparison with Ohter Frameworks“, VueJS, [Võrgumaterjal]. Loetud aadressil <https://vuejs.org/v2/guide/comparison.html>. [Kasutatud 31 10 2021].
- [24] E. Mwangi, „What is MVC Architecture and how does it work?“, Section, [Võrgumaterjal]. Loetud aadressil <https://www.section.io/engineering-education/what-is-mvc-and-how-does-it-work/>. [Kasutatud 26 11 2021].
- [25] H. M. Mölder, Bitbucket, [Võrgumaterjal]. Loetud aadressil <https://bitbucket.org/hmolde/food-planning-system/src/master/>.
- [26] L. Gupta, „What is REST“, REST API Tutorial, [Võrgumaterjal]. Loetud aadressil <https://restfulapi.net/>. [Kasutatud 27 12 2021].

## **Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks<sup>1</sup>**

Mina, Harri Martin Mölder

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „Toiduvarude planeerimise infosüsteemi analüüs ja arendamine“, mille juhendaja on Toomas Lepikult
  - 1.1. reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
  - 1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

31.12.2021

---

<sup>1</sup> Lihtlitsents ei kehti juurdepääsupiirangu kehtivuse ajal vastavalt üliõpilase taotlusele lõputööle juurdepääsupiirangu kehtestamiseks, mis on allkirjastatud teaduskonna dekaani poolt, välja arvatud ülikooli õigus lõputööd reprodutseerida üksnes säilitamise eesmärgil. Kui lõputöö on loonud kaks või enam isikut oma ühise loomingu tegevusega ning lõputöö kaas- või ühisautor(id) ei ole andnud lõputööd kaitsvale üliõpilasele kindlaksmääratud tähtajaks nõusolekut lõputöö reprodutseerimiseks ja avalikustamiseks vastavalt lihtlitsentsi punktile 1.1. ja 1.2, siis lihtlitsents nimetatud tähtaja jooksul ei kehti.

## Lisa 2 – Küsimustik ja vastuse variandid

1. Kui sageli Te käite toidukaupa ostmas toidulistiga? \*

Mitte kunagi      1      2      3      4      5      Alati

2. Kas ja kui pikalt Te oma toidu tegemist ette planeerite? \*

- Ei planeeri oma toidu tegemist ette
- Ainult tänase päeva toitumist
- Paari järgneva päeva toitumist
- Käesoleva nädala toitumist
- Tulevast või tulevate nädalate toitumist
- Planeerin rohkem ette

3. Kui sageli toiduainete allahindlused ja kampaaniad mõjutavad Teie toiduainete ostmist? \*

Mitte kunagi      1      2      3      4      5      Väga sageli

4. Kui mitmele inimesele Te tavaliselt toitu ostate? \*

- Ühele inimesele
- Kahele inimesele
- Kolmele inimesele
- Neljale või enamale inimesele

5. Kui sageli jääb Teil ostetud toiduaineid üle, mida Te ei oska ära kasutada ja mis seetõttu riknevad? \*

Ei jää kunagi üle      1      2      3      4      5      Pidevalt või alati jääb üle

6. Teie sugu \*

- N
- M
- Muu

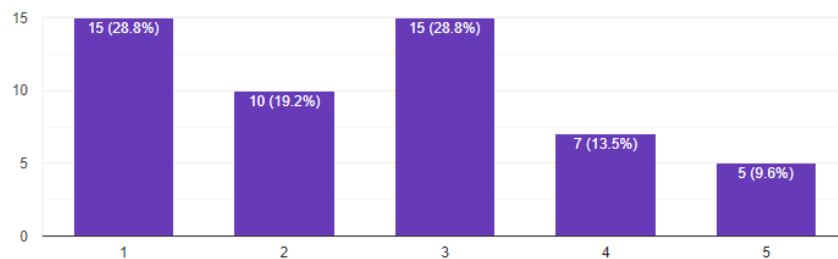
7. Teie vanusegrupp \*

Valige ▼

## Lisa 3 Küsimustiku tulemused

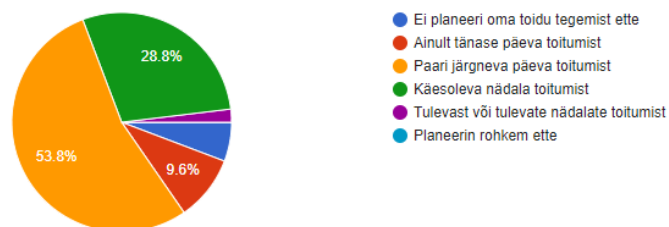
1. Kui sageli Te käite toidukaupa ostmas toidulistiga?

52 responses



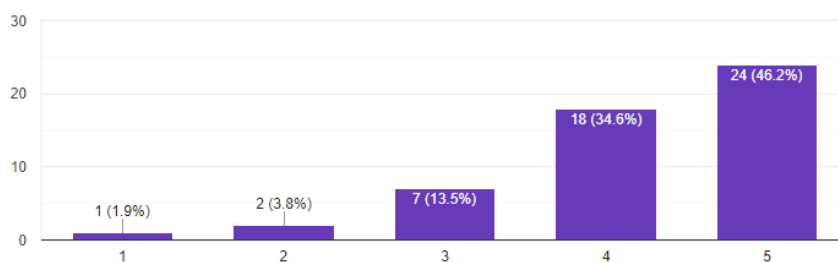
2. Kas ja kui pikalt Te oma toidu tegemist ette planeerite

52 responses



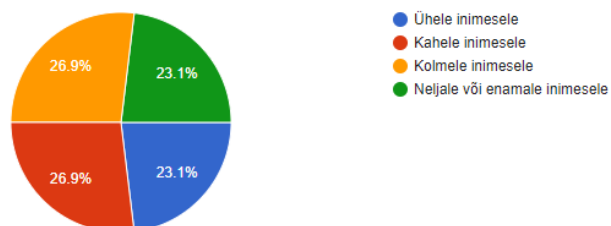
3. Kui sageli toiduainete allahindlused ja kampaaniad mõjutavad Teie toiduainete ostmist?

52 responses



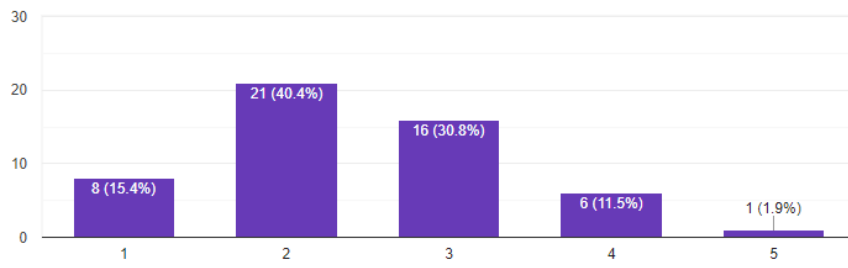
4. Kui mitmele inimesele Te tavaliselt toitu ostate?

52 responses



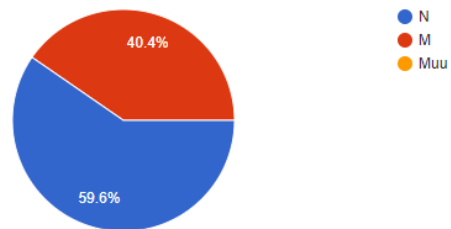
5. Kui sageli jääb Teil ostenud toiduaineid üle, mida Te ei oska ära kasutada ja mis seetõttu riknevad?

52 responses



6. Teie sugu

52 responses



7. Teie vanusegrupp

52 responses

