

TALLINNA TEHNIKAÜLIKOOL  
Infotehnoloogia teaduskond  
Tarkvarateaduse instituut

Sven Kõiv 142799IAPB

# **VEDELKÜTUSE HANGETE KORRALDAMISE VEEBIRAKENDUS**

Bakalaureusetöö

Juhendaja: Jaagup Irve

Tehnikateaduste  
magister

Tallinn 2018

## **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: Sven Kõiv

15.05.2018

## **Annotatsioon**

Töö eesmärk on teha veebirakendus, kus saavad era- ja juriidilised isikud korraldada vedelkütuse hankeid.

Töö tulemusena valmis veebirakendus, kus on realiseeritud kasutajaks registreerimine mobiil-ID ja smart-ID abil. On loodud kaks rolli: pakkuja ja hankija. Hankija saab korraldada vedelkütuse hanke, kas eraisiku või ettevõttena ning vedelkütuse müüja ehk pakkuja saab sellel hankel osaleda. Hankija saab valida hanke võitja ja seejärel suhelda temaga reaalsajas läbi *WebSocket*'i, et vedelkütuse tarnimise protsess läheks võimalikult sujuvalt.

Veebirakendus on kättesaadav järgneval aadressil:

<https://turgoil.ee>

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 42 leheküljel, 5 peatükki, 33 joonist, 0 tabelit.

## **Abstract**

### **Web Applicaton For Buying Liquid Fuels**

The goal of this thesis is to make a web application where people and companies are able to make requests to buy liquid fuels in large quantities.

The product of this thesis consists of registration with smart-ID and mobile-ID which are very popular authentication methods in Estonia. In the application there are two roles the seller and the buyer. Request to buy liquid fuel can be made as a company or as a person. Liquid fuel sellers are able to make bids to the request. The buyer can choose the winner who is going to provide her with fuel. During the deal they can chat real-time through WebSockets in order to make the deal go smoother.

The application web address is following:

<https://turgoil.ee>

The thesis is in Estonian and contains 42 pages of text, 5 chapters, 33 figures, 0 tables.

## Lühendite ja mõistete sõnastik

<i>Express</i>	Veebirakenduse raamistik Javascriptis serveri arendamiseks. [1]
Hange	Hankija poolt korraldatud hange, millele saavad pakkujad teha pakkumisi.
Hankija	Eraisik või juriidiline isik, kes soovib korraldada hanke.
<i>JWT</i>	See on andmete edastusviis kahe osapoole vahel, kus tavaliselt üks osapool allkirjastab need ning teine kasutab neid. Need andmed kehtivad ainult määratud ajaks. [2]
<i>Node</i>	Arenduskeskkond, mis kasutab Javascripti ning milles programmeerimine toimub asünkroonselt. [3]
<i>NPM</i>	Javascripti moodulite haldamisevahend, mis võimaldab alla laadida jagatud mooduleid. [4]
Pakkuja	Juriidiline isik, kes osaleb hangetel ning teeb omapoolseid pakkumisi.
<i>Port forwarding</i>	Pordisiire ehk võrgupordi siirdamine ühelt masinalt teisele. [5]
<i>React</i>	Veebirakenduse raamistik Javascriptis kasutajaliideste arendamiseks. [6]
<i>SPA</i>	Üksleht rakendus.
<i>SSL (Secure Sockets Layer)</i>	„Infoturbe protokoll üle Interneti edastatavate andmete turvalisuse tagamiseks.“ [5]
<i>TCP</i>	„TCP lisab internetiprotokollile töökindla sideühenduse ja andmevoo reguleerimise ning võimaldab täisdupleksühendusi.“ [5]
<i>URL</i>	Internetiaadress. [5]
Vedelkütus	Bensiin või diislikütus
<i>WebSocket</i>	Arvuti kommunikatsiooni protokoll, mis ühendab kaks osapoolt üle TCP, mille tulemusena mõlemad saavad samal ajal rääkida . [7]

## Sisukord

1 Sissejuhatus .....	10
2 Taust .....	11
2.1 Mis on vedelkütus? .....	11
2.2 Vedelkütuse ostmise protsess hetkel erasektoris .....	11
2.3 Mis on vedelkütuse hange? .....	11
3 Rakenduse kavandamine .....	12
3.1 Probleemi püstitus ja töö eesmärk .....	12
3.2 Funktsionaalsed nõuded .....	13
3.3 Alternatiivsed lahendused .....	15
4 Veebirakendus .....	17
4.1 Kasutatud tehnoloogiad .....	17
4.2 Andmebaasi arhitektuur .....	17
4.2.1 Isik ja seosed .....	17
4.2.2 Autentimine .....	18
4.2.3 Ettevõtte ja seosed .....	19
4.2.4 Hange ja seosed .....	21
4.2.5 Pakkumine ja seosed .....	22
4.2.6 Sõnum ja seosed .....	23
4.3 Rakenduse arhitektuur .....	24
4.3.1 Hanke seisundi automaatne uuendamine .....	24
4.3.2 WebSocket .....	25
4.4 Turvalisus .....	25
4.5 Kasutajaliides .....	28
4.5.1 Isiku tuvastamine mobiil-ID abil .....	29
4.5.2 Külastaja tuvastas end mobiil-ID abil .....	30
4.5.3 Kasutaja registreerimine õnnestus .....	30
4.5.4 Kasutaja on sisse loginud .....	31
4.5.5 Kõik hanked - hankija .....	31
4.5.6 Seaded - konto .....	32

4.5.7 Seaded - muuda parooli .....	33
4.5.8 Seaded - ettevõtte.....	33
4.5.9 Seaded – ettevõtte pildi lisamine .....	34
4.5.10 Lisa hange.....	35
4.5.11 Minu hanked .....	35
4.5.12 Hanke andmed – Pakkuja .....	36
4.5.13 Hanke andmed – Võitja valimine hankija .....	37
4.5.14 Hanke andmed – Teostamisel.....	37
4.5.15 Sõnum.....	38
4.5.16 Sõnum - Hankija.....	38
4.5.17 Sõnum - Pakkuja.....	39
4.6 Tagasiside .....	40
5 Kokkuvõte .....	41
Kasutatud kirjandus .....	42

## Jooniste loetelu

Joonis 1. Vedelkütuse ostmise protsess hetkel erasektoris (kliendi vaates) .....	11
Joonis 2. Vedelkütuse hanke korraldamise lahendus (hankija vaates) .....	13
Joonis 3. Kasutusjuhtude diagramm .....	15
Joonis 4. Isiku tabelid ja seosed. ....	18
Joonis 5. Autentimise tabelid. ....	19
Joonis 6. Ettevõtte tabelid ja seosed. ....	20
Joonis 7. Hanke tabelid ja seosed. ....	21
Joonis 8. Pakkumise tabelid ja seosed. ....	22
Joonis 9. Sõnumite tabelid ja seosed. ....	23
Joonis 10. Hanke seisundi automaatne uuendamine .....	24
Joonis 11. JWT allkirjastamine. ....	26
Joonis 12. JWT verifitseerimine. ....	26
Joonis 13. WebSocket'i ühenduse kaitsmine. ....	27
Joonis 14. Kasutajaliidese moodulid .....	28
Joonis 15. Kasutaja loomine mobiil-ID abil .....	29
Joonis 16. Mobiil-ID abil isiku tuvastamine .....	30
Joonis 17. Registreerimine õnnestus .....	30
Joonis 18. Sõnum konto aktiveerimine .....	31
Joonis 19. Sisse loginud kasutaja .....	31
Joonis 20. Kõik hanked - Hankija .....	32
Joonis 21. Seaded - Konto .....	32
Joonis 22 Seaded – Muuda parooli .....	33
Joonis 23. Seaded - Ettevõtte .....	34
Joonis 24. Seaded – Ettevõtte pildi lisamine .....	34
Joonis 25. Lisa hange .....	35
Joonis 26. Minu hanked .....	36
Joonis 27. Hanke andmed – Pakkuja .....	36
Joonis 28. Hanke andmed – Võitja valimine .....	37
Joonis 29. Hanke andmed – Hange on teostamisel .....	38



Joonis 30. Sõnum pärast seda kui hange läks olekusse teostamisel .....	38
Joonis 31. Sõnum - hankija vaade .....	39
Joonis 32. Sõnum – Pakkuja vaade .....	39
Joonis 33. Sõnum - Pakkuja vaade – Lõpeta hange .....	40

## 1 Sissejuhatus

Hetkel Eestis, infotehnoloogia ajastul, ei ole võimalik korraldada vedelkütuse hanget erasektoris, mille protsess oleks kiire ning tagaks ka turu parima hinna. Et osta vedelkütust, peavad nii era- kui ka juriidilised isikud võtma eraldi ühendust kõikide pakkujatega kas telefoni või e-posti teel. Eestis on kütusefirmasid ligikaudu 120 [8]. Kui klient soovib saada turu parimat hinda, siis oleks väga ajakulukas kõik pakkujad läbi helistada.

Käesoleva bakalaureusetöö eesmärk on luua veebirakendus, mis koondab kõik vedelkütuse hanked ühte keskkonda. Samuti vähendab see kütuseostjate ajakulu ning suurendab turu konkurentsi.

Lõputöö idee pärineb inimeselt, kes tegeleb hulgikütuse müügiga Eestis ning kellega sai arenduse käigus aktiivselt suheldud. Rakenduse arhitektuur ja tarkvara on disainitud autori poolt ning selle arendamiseks on kasutatud *NPM*'i mooduleid.

Töö teises peatükis kirjutatakse lahti, mis on hange ning selgitatakse vedelkütuse hanke korraldamise protsessi lähemalt. Kolmandas peatükis tõstatatakse probleem, kirjutatakse lahti veebirakenduse funktsionaalsed nõuded ning tuuakse ülevaade alternatiivsetest lahendustest. Neljandas peatükis tuuakse ülevaade valminud veebirakendusest, selle arhitektuurist ja kasutatud tehnoloogiast. Samuti on välja toodud vedelkütuse hulгимüüja tagasiside valminud rakenduse kohta.

## 2 Taust

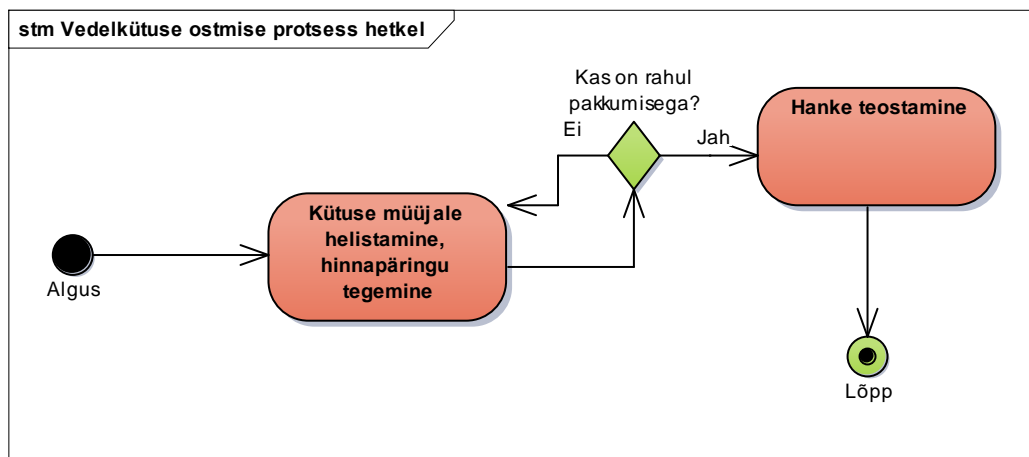
Käesolevas peatükis seletatakse, mis on vedelkütus ning kirjeldatakse vedelkütuse ostmise protsessi hetkel. Sõnastatakse vedelkütuse hanke mõiste.

### 2.1 Mis on vedelkütus?

Vedelkütus on vedel põlevaine, mida kasutatakse energiaallikana soojusjõumasinates, näiteks sisepõlemismootoris sõiduautes ja veoautodes [9], [10]. Vedelkütuste hulka kuuluvad näiteks diislikütus, bensiin [9].

### 2.2 Vedelkütuse ostmise protsess hetkel erasektoris

Selleks, et saada parima hinnaga kütust erasektoris, peab klient teadma ettevõtete kontakte, kes tegelevad kütuse müügiga. Kuna ettevõtteid on palju, siis võtab kõikide kütusepakujate läbihelistamine palju aega (Joonis 1).



Joonis 1. Vedelkütuse ostmise protsess hetkel erasektoris (kliendi vaates)

### 2.3 Mis on vedelkütuse hange?

Raamatus „Riigihanked“ on defineeritud mõiste riigihange kui kaupade ostmine, teenuste või ehitustööde tellimine hankija poolt [11]. Vedelkütuse hanke kontekstis tähendab see seda, et ostja ehk hanke korraldaja ehk hankija soovib osta vedelkütust.

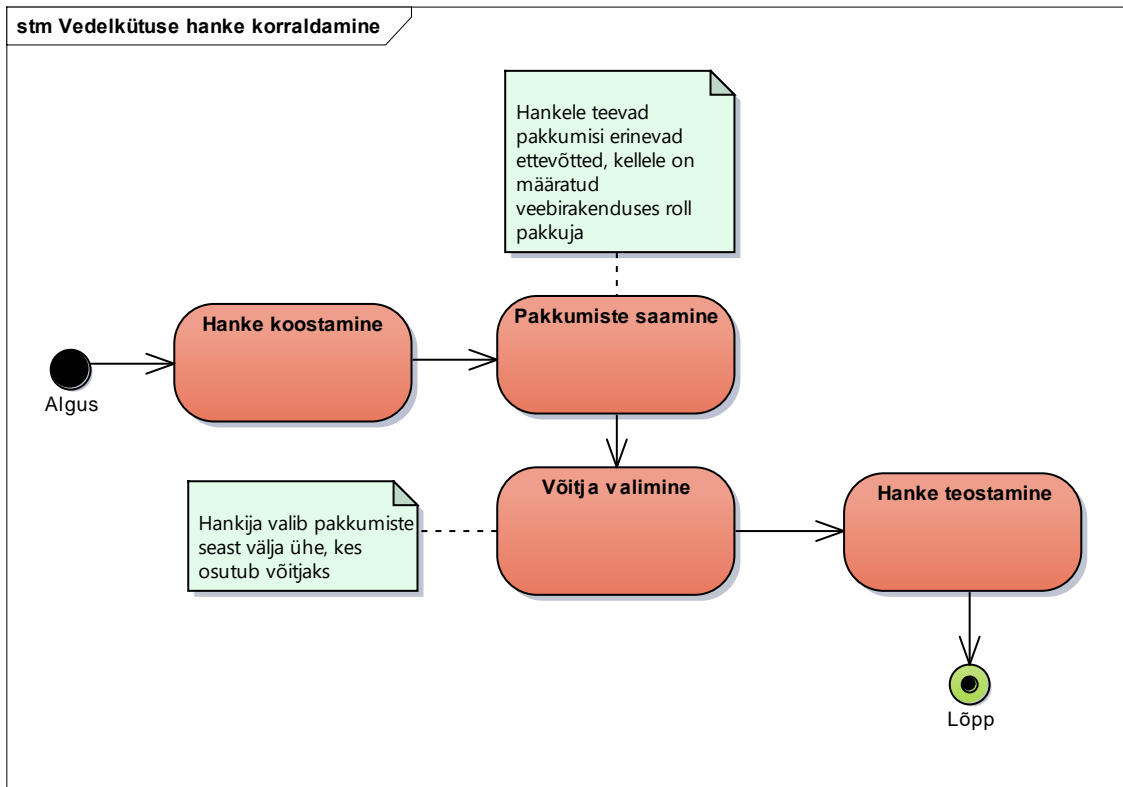
## **3 Rakenduse kavandamine**

Käesolevas peatükis sõnastatakse probleem, millest lähtuvalt püstitatakse töö eesmärk. Samuti kirjutatakse lahti loodava rakenduse funktsionaalsed nõuded ning antakse ülevaade alternatiivsetest lahendustest.

### **3.1 Probleemi püstitus ja töö eesmärk**

Tänapäeval on vedelkütuse hanke korraldamine aeganõudev tegevus. Kui hanke korraldaja ei võta ühendust kõikide hulgikütuse müüjatega, siis ei ole garanteeritud ka turu parim hind. Hetkel ei ole Eestis võimalik korraldada vedelkütuse hanget erasektoris, mis tagaks turul parima hinna ning vähendaks klientide helistamise ajakulu.

Töö eesmärk on teha veebirakendus, mis koondaks kõik vedelkütuse hanked ja pakkumised ühte keskkonda. Seejärel saab klient valida pakkuja, kes teeb talle parima pakkumise (Joonis 2). Selline lahendus säästab aega ning aitab tagada ka parimat turuhinda ning suurendab konkurentsi.



Joonis 2. Vedelkütuse hanke korraldamise lahendus (hankija vaates)

### 3.2 Funktsionaalsed nõuded

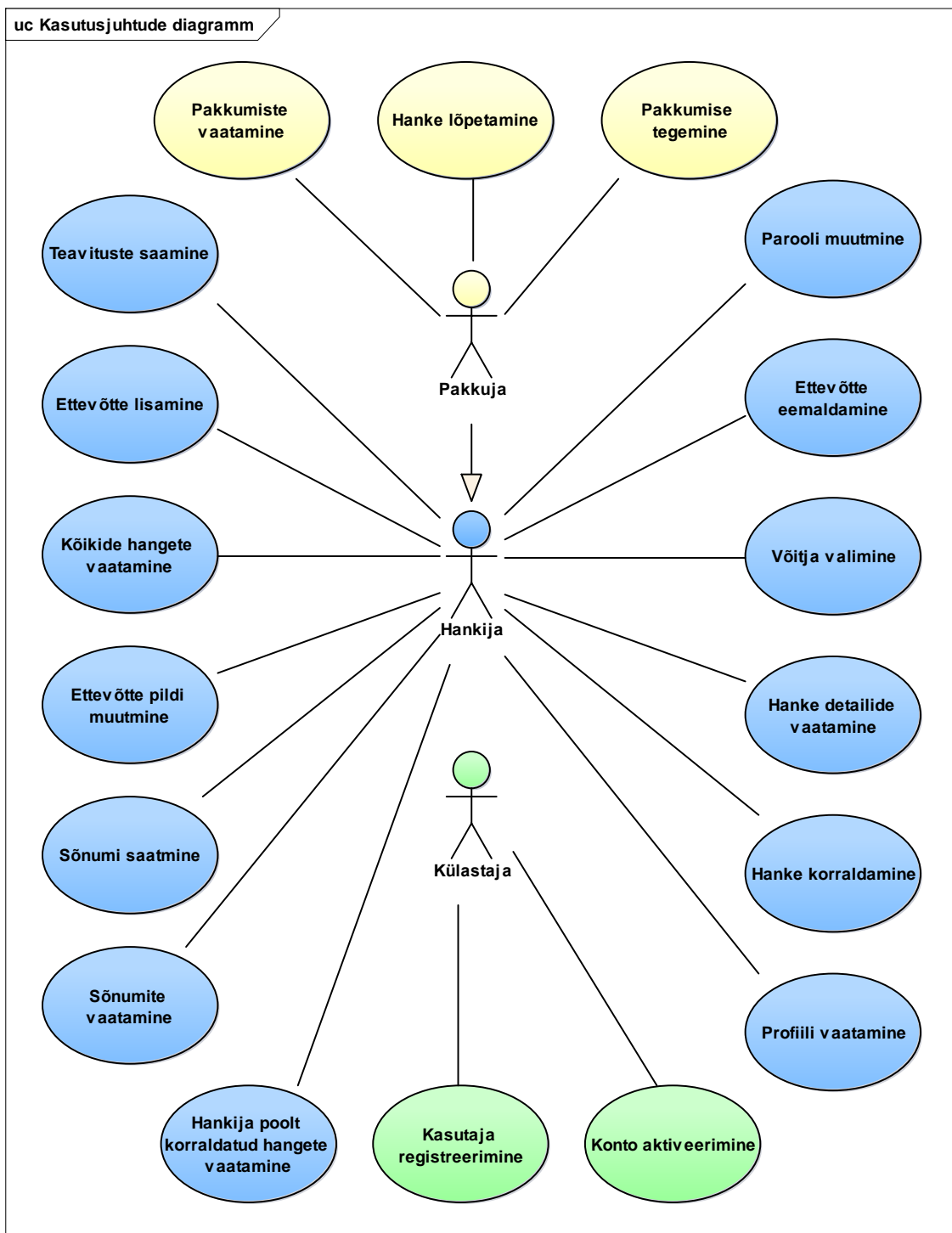
- Rakendus peab töötama Mozilla Firefox 59.0.3 ja Chrome 66.0.3359.139 veebilehitsejates.
- Kasutaja saab ennast registreerida mobiil – ID või smart – ID abil.
- Kasutaja saab pärast registreerimist aktiveerida end vajutades lingile, mis on saadetud tema e-posti aadressile.
- Kasutaja saab end autentida e-posti ja parooli abil.

**Järgnevatel funktsionaalsetel nõuetel eeldatakse, et kasutaja on sisse loginud.**

- Kasutaja näeb enda profiili.
- Kasutaja, kes on hankija, saab korraldada hanke.
- Kasutaja näeb aktiivseid ja lõppenud hankeid.

- Kasutaja, kes on pakkuja, näeb hankija andmeid, mis aitavad tal pakkumist teha.
- Kasutaja näeb hanke detailset vaadet.
- Kui hange on aktiivne, siis hanke korraldaja ei näe pakkumise summat.
- Hankijale on pakkumised varjatud. Kui valib võitja, siis ta näeb kes sellise pakkumise tegi.
- Hanke korraldaja saab valida võitja.
- Hanke korraldajal on ülevaade tema korraldatud hangetest.
- Pakkujal on ülevaade tema korraldatud pakkumistest.
- Sõnumite saatmine hanke võidu puhul reaalses ajas hankija ja pakkuja vahel.
- Kui pakkuja on kindel, et hange on edukalt lõppenud (klient on kütuse kätte saanud ja pakkuja on raha kätte saanud), siis pakkuja lõpetab hanke.
- Hanget saab korraldada eraisikuna või ettevõttena.
- Ettevõtte lisamine.
- Ettevõtte logo lisamine.
- Parooli vahetamine.
- Ettevõtte eemaldamine.

(Joonis 3)



Joonis 3. Kasutusjuhtude diagramm

### 3.3 Alternatiivsed lahendused

Sellist veebirakendust, mis keskendub vedelkütuse hulгимүүгиле ei ole Eestis tehtud. Sarnasel põhimõttel töötavad lahendused on näiteks <https://riigihanked.riik.ee/register/>, <http://hange.ee/>, <https://www.timber.ee/>. Timber on samuti hangete keskkond, kuid see

on spetsialiseerunud raieõiguste enampakkumistele. Hange.ee keskendub eelkõige ehitushangete korraldamisele, kus klient saab valida sobivaima ehitusettevõtte, kes teostab hanke. Keskkond „Riigihanked“ keskendub avaliku sektori probleemide lahendamisele.



## 4 Veebirakendus

Käesolevas peatükis räägitakse veebirakenduses kasutatud tehnoloogiast. Antakse ülevaade andmebaasi ja rakenduse arhitektuurist, rakenduse turvalisusest ning kasutajaliidesest. Lisaks on välja toodud ka tagasiside rakendusele vedelkütuse hulgimüüja poolt.

### 4.1 Kasutatud tehnoloogiad

Veebirakenduse loomiseks on kasutatud järgnevaid programmeerimiskeeli: Javascript, MySQL, HTML, LESS / CSS.

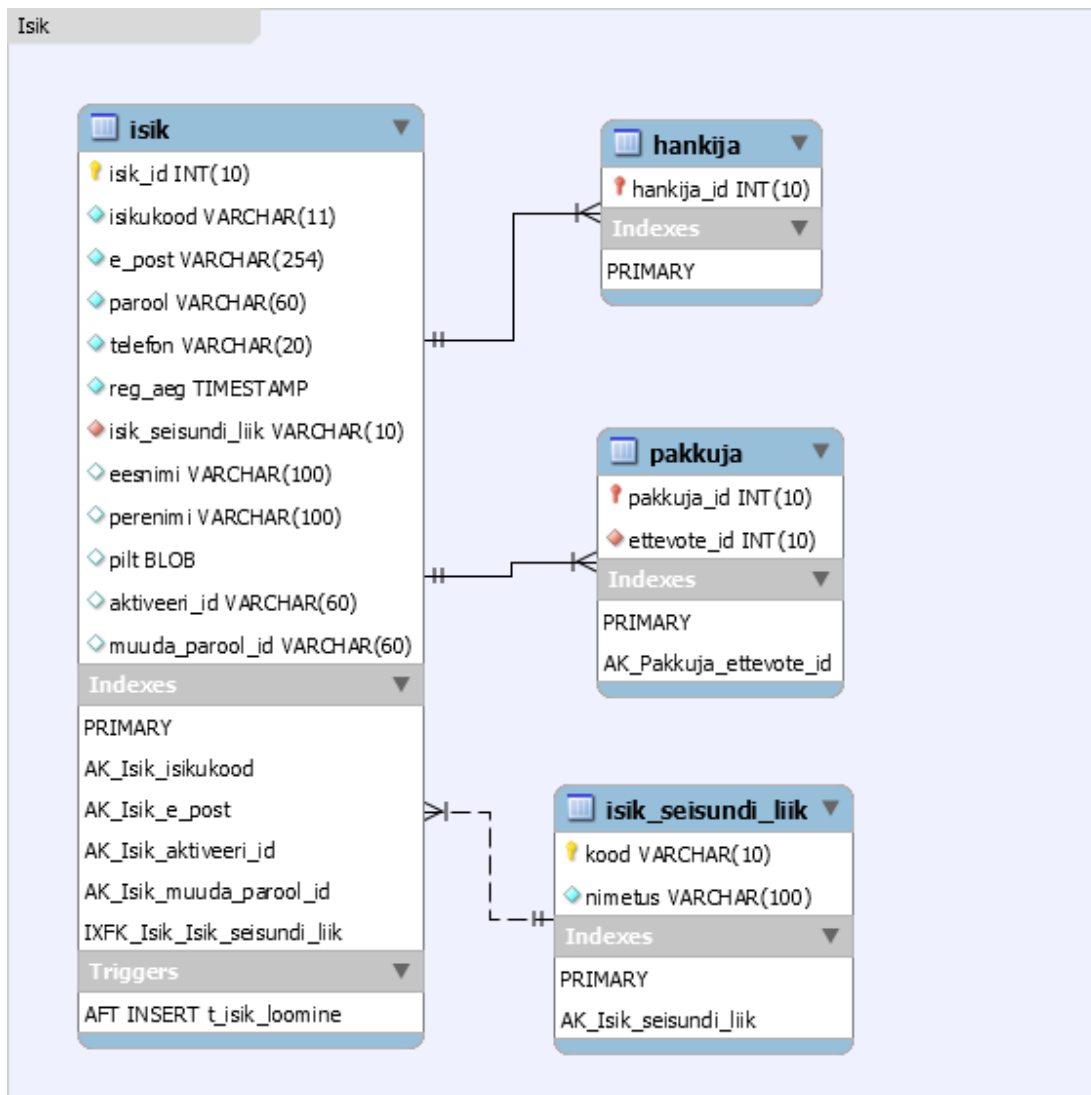
Node rakenduse serveri raamistik on *Express* ning kasutajaliidesel on see *React*. Tehnoloogiad said valitud sellised, sest autor on nendega kõige rohkem kokku puutunud ning usub, et nende tööriistade abil saab paindliku rakenduse valmis, mis võimaldab ka reaajas suhtlust läbi *WebSocket*'i.

### 4.2 Andmebaasi arhitektuur

Järgnevalt kirjeldatakse andmebaasi arhitektuuri ning seletatakse lahti andmetüübid, mis vajavad täiendavat kommenteerimist. Andmetüüpide suurused said valitud lähtudes eelkõige Eesti riigist. Andmebaasiks on MariaDB 10.2.

#### 4.2.1 Isik ja seosed

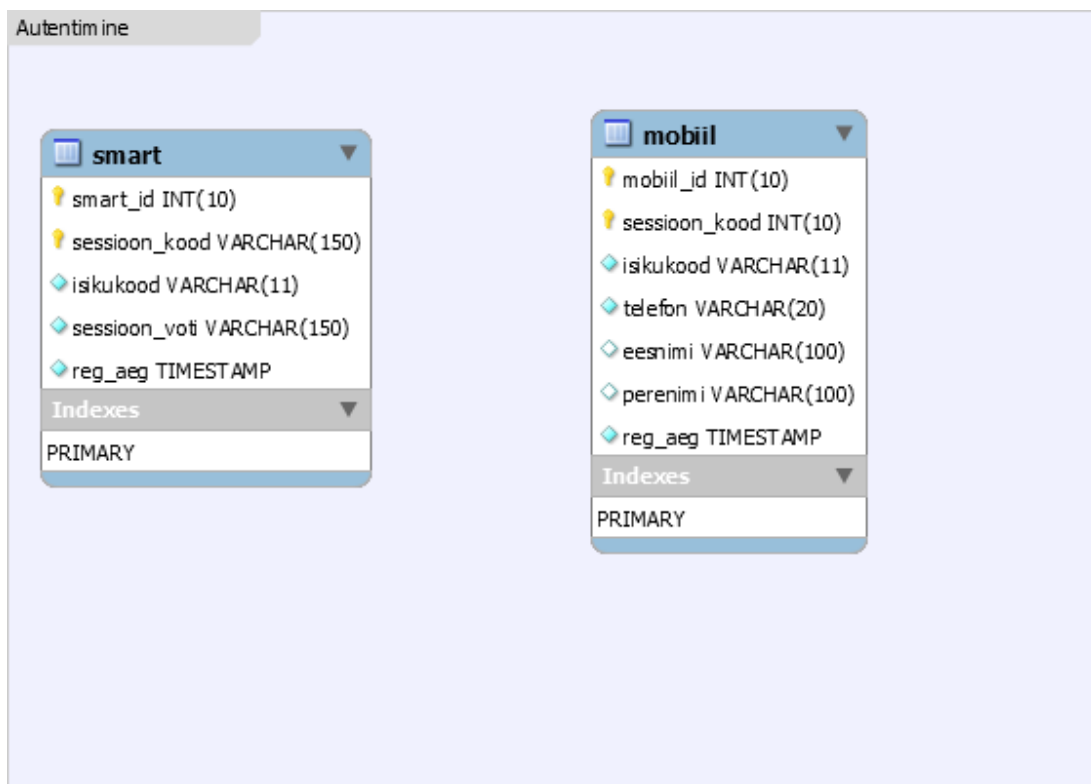
Tabelis Isik on parool räsitud. Isikule on määratud seisundi liik, mis on kas aktiivne või mitteaktiivne. Kui registreeritakse isik, siis see on automaatselt seisundis mitteaktiivne. Selleks, et kasutaja aktiveerida, peab minema leheküljele, kus on *URL*'i sees unikaalne aktiveerimise identifikaator. Unustatud parooli muutmise (kasutaja pole sisse loginud) toimub analoogselt, kuid kuna töömaht on suur, siis hetkel seda ei ole implementeeritud. Kui luuakse uus isik, siis käivitub protsess, mis lisab isiku ka tabelisse Hanke\_sonum\_vaatamine, selleks, et oleks teada, millal oli viimane kord kui isik oma sõnumeid vaatas. (Joonis 4)



Joonis 4. Isiku tabelid ja seosed.

#### 4.2.2 Autentimine

Mõlemas tabelis on sessioon\_kood, mis on tagastatud kolmanda osapoole SK ID Solutions poolt, mille abil tehakse järjekordne päring, et saada teada, kas kasutaja on ennast edukalt autentitud. Selleks, et olla kindel, et need identifikaatorid on unikaalsed, siis on loodud eraldi identifikaatorid mõlemale tabelile. Smart-ID puhul on vaja hoiustada ka sessiooni võtit, mis on juhuslikult genereeritud räsi väärtus, mille abil hiljem kontrollitakse sertifikaadi valiidsust. [12] (Joonis 5)



Joonis 5. Autentimise tabelid.

#### 4.2.3 Ettevõtte ja seosed

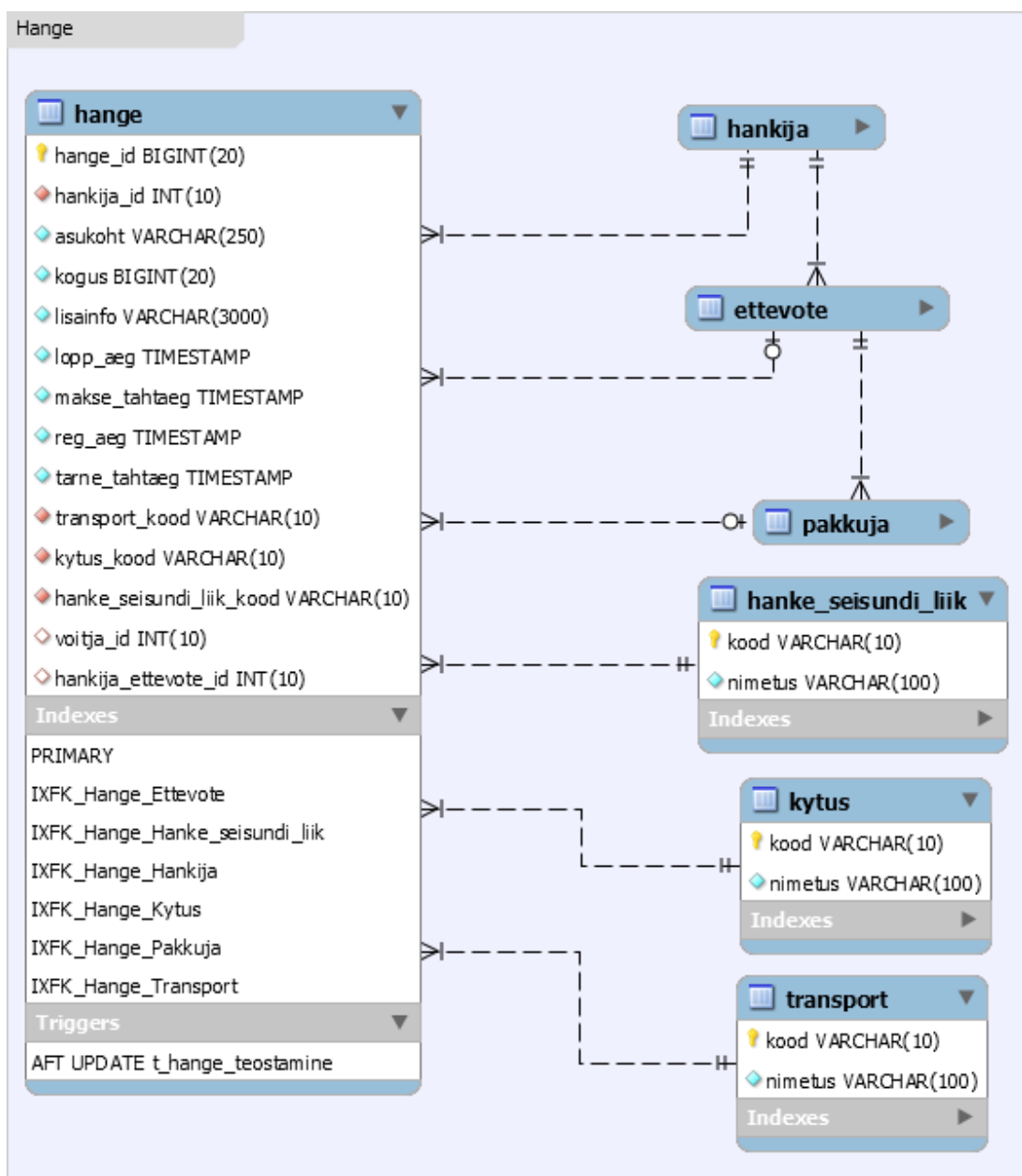
Tabeli Ari\_register andmed on võetud Registrite ja Infosüsteemide Keskusest [13]. Selles tabelis ariregistri\_kood on indekseeritud, sest see teeb ettevõtte otsimise andmebaasist kiiremaks. Äriregistri andmeid uuendab Registrite ja Infosüsteemide Keskus kord nädalas. Ettevõtte pildi andmetüüp on *TEXT*, kuna seda on mugavam edastada kliendile võrreldes *BLOB*'iga. Kui kasutaja eemaldab ettevõtte, siis ettevõtet andmebaasist realselt ei eemaldata, vaid see läheb mitteaktiivsesse olekusse. (Joonis 6)



Joonis 6. Ettevõtte tabelid ja seosed.

#### 4.2.4 Hange ja seosed

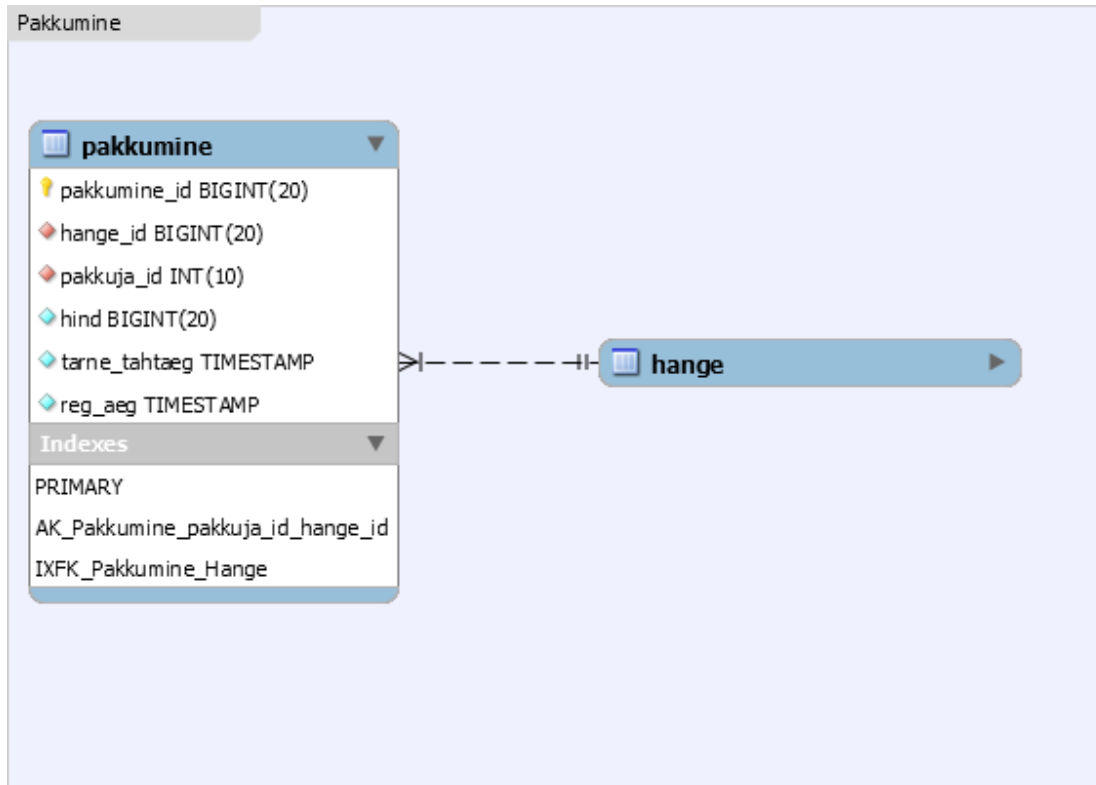
Tabelis Hange on alati hankija\_id, et oleks teada, kes selle hanke korraldas. Kui kasutaja korraldas hanke ettevõtteks, siis lisatakse ka hankija\_ettevotte\_id, kui see puudub, siis hange on korraldatud eraisikuna. Kui hankija on võitja valinud, siis hange läheb seisundisse teostamisel, seejärel lisatakse tabelisse Hange võitja\_id, see on sama mis pakkuja\_id. Pärast võitja valimist käivitub protsess, mis saadab hanke võitjale ja hankijale automaatselt sõnumi, et kasutajaliideses oleksid mõlemad teadlikud sellest, et saavad nüüd omavahel suhelda. (Joonis 7)



Joonis 7. Hanke tabelid ja seosed.

#### 4.2.5 Pakkumine ja seosed

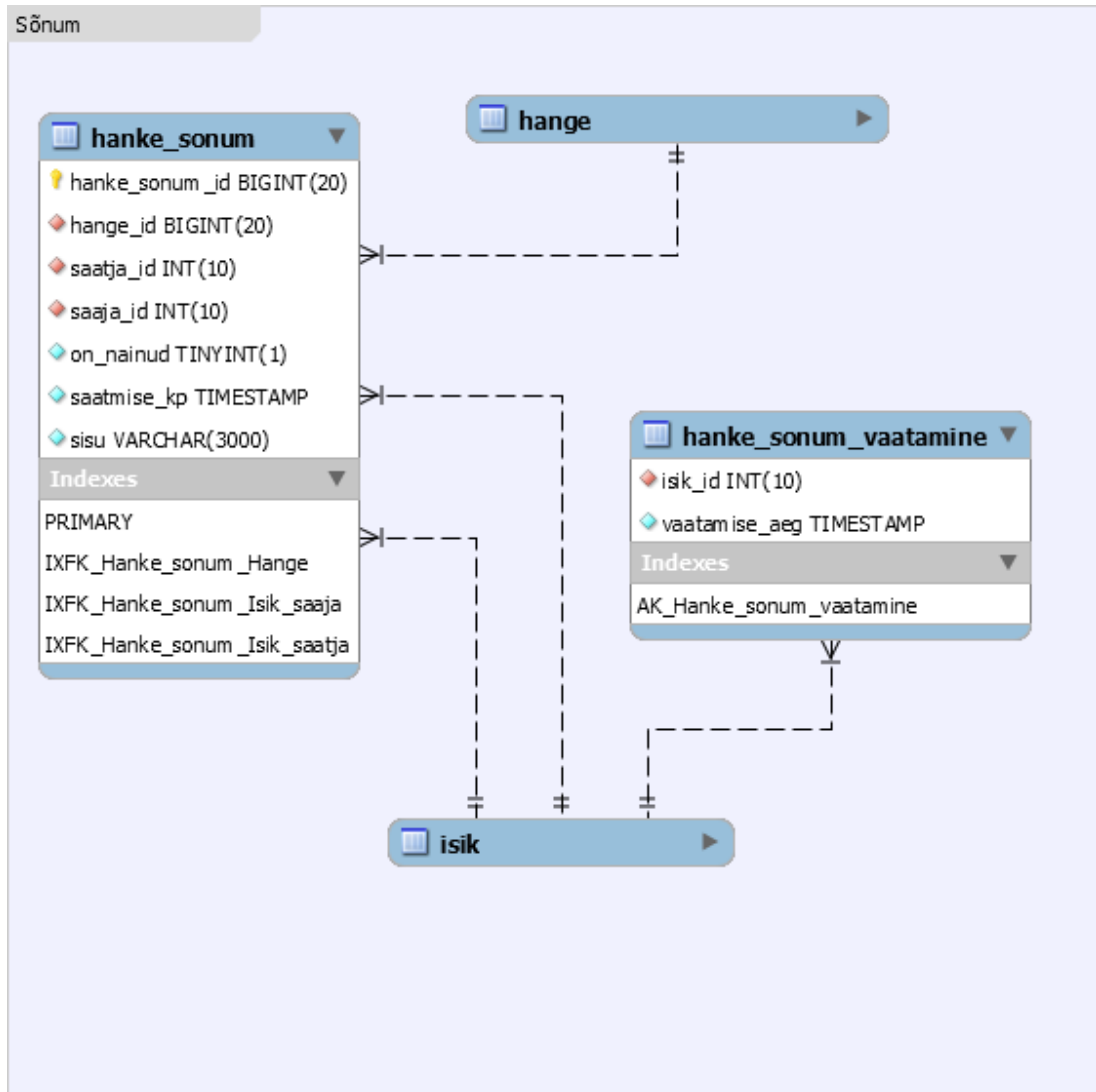
Tabelis Pakkumine saab teha pakkuja ainult ühe pakkumise hankele, selle jaoks on loodud unikaalsuse kitsendus hange\_id ja pakkuja\_id vahel. (Joonis 8)



Joonis 8. Pakkumise tabelid ja seosed.

#### 4.2.6 Sõnum ja seosed

Igale hankele luuakse eraldi oma jututuba. Ning kui saadetakse sõnum, siis on vaja teada, kes saadab kellele ja mis hanke asjus. Selleks, et saada teavitusi, kui uus sõnum on saadetud, siis on ka selline tabel nagu hanke\_sonum\_vaatamine. Selles tabelis on kirjas, millal antud isik sõnumit vaatas. Kui kasutaja vaatas sõnumeid hiljem, kui oli saadetud viimane sõnum, siis ei näidata teavitust, et uus sõnum on saadetud. (Joonis 9)



Joonis 9. Sõnumite tabelid ja seosed.

## 4.3 Rakenduse arhitektuur

Apache on konfigureeritud nii, et päringud, mis tulevad pordile 80 ja 443 on suunatud proksi abil *Node* rakendusele, mis kuulab pordil 3000 sisse tulevaid päringuid. *WebSocket*'i realiseerimiseks on kasutatud *port forwarding* meetodit, mille tulemusel suunatakse sissetulev päring otse pordile 3433. Selleks, et liiklus läbi pordi 3433 oleks turvaline on hangitud *SSL* sertifikaat, mille abil krüpteeriti liiklus.

Selleks, et saaks kasutaja registreerimisel isiku tuvastada, siis saadab server mobiil-ID või smart-ID päringu SK ID Solutions demo keskkonnale. Nemad saavad vastuse autentimise staatusest ning vastavalt sellele server tagastab veateate kliendile või töötleb seda päringut edasi.

### 4.3.1 Hanke seisundi automaatne uuendamine

Rakenduses on hankel mitu erinevat seisundit nagu näiteks aktiivne, võitja valimine, teostamisel ja lõppenud. Kuna osade seisundite puhul on määratud ajaline piirang, kui kaua see tohib sellises seisundis olla, siis on vaja ka lahendust, mis suudaks neid ühest olekust teise viia. Seda teeb järgnev kood, mis küsib iga viie minuti tagant antud seisundis olevad hanked, mis peaksid muutma oma seisundit järgneva kümne minuti jooksul. Igale hankele määratakse eraldi mitme millisekundi pärast see oma seisundit muudab. Kui on vaja seisundit muuta, siis rakendus teeb andmebaasis vastava muudatuse ning teavitab sellest kõiki, kes on *WebSocket* ühenduses. (Joonis 10)

```
12
13  /**
14   * Start polling the changes.
15   */
16  (() => {
17    pollActiveDemandsEnding();
18    pollChoosingWinnerDemandsEnding();
19    setInterval(pollChoosingWinnerDemandsEnding, FIVE_MINUTES);
20    setInterval(pollActiveDemandsEnding, FIVE_MINUTES);
21  })();
22
```

Joonis 10. Hanke seisundi automaatne uuendamine



### 4.3.2 WebSocket

Kui klient soovib ennast ühendada *WebSocket*'i abil serveriga, siis selleks ta saadab päringu, milles on tema *JWT*. Kui server aktsepteerib selle päringu, siis toimub ühenduse loomine ning ühendusele tekib unikaalne identifikaator. Kuna on vaja saata personaalseid sõnumeid kahe arvuti vahel, mis on ühenduses serveriga, siis server seob *WebSocket*'i unikaalse identifikaatori *JWT* sees oleva kasutaja identifikaatoriga. Seega, kui keegi soovib saata personaalset sõnumit kasutajale, kelle identifikaator on näiteks 5, siis server vaatab, kas kasutaja identifikaatoriga 5 on ühenduses serveriga. Kuna kasutaja identifikaator oli eelnevalt seotud *WebSocket* identifikaatoriga, siis on serverile teada ka see ühendus, kus kasutaja kuuluab ning edastab selle sõnumi sinna.

## 4.4 Turvalisus

Kasutajate registreerimine toimub mobiil-ID ja smart-ID abil, et teha kindlaks, kas see isik on ikka see, kes ta väidab end olevat. Juhul kui isik on end tuvastanud mobiil-ID või smart-ID abil, väljastatakse kasutajatele allkirjastatud andmed *JWT* kujul, mis kehtib tund aega (Joonis 11). Seejärel, kui kasutaja saadab teinekord oma andmed, siis kontrollitakse, kas tema saadetud *JWT* on valiidne (Joonis 12). Juhul kui on, siis ta saab end registreerida kasutajaks.

```

28
29  /**
30   * Sign the token.
31   *
32   * @payload {object} The data to be signed.
33   * @expires {string} The time how long token is valid
34   * @returns {Promise<string>} The signed JWT
35   */
36  function signToken(payload, expires) {
37    const opts = {
38      issuer: rc.jwt.issuer,
39      audience: rc.jwt.audience,
40      expiresIn: expires,
41    };
42    return new Promise((resolve, reject) => {
43      jwt.sign(payload, rc.jwt.secret, opts, (err, token) => {
44        if (err) {
45          log.error(err);
46          reject(err);
47        } else {
48          resolve(token);
49        }
50      });
51    });
52  }
53

```

Joonis 11. JWT allkirjastamine.

```

53
54  /**
55   * Verify the token
56   *
57   * @params {string} The data to decoded
58   * @returns {Promise<string>} The decoded data
59   */
60  function verifyToken(token) {
61    return new Promise((resolve, reject) => {
62      jwt.verify(token, rc.jwt.secret,
63        {audience: rc.jwt.audience, issuer: rc.jwt.issuer}, (err, decoded) => {
64        if (err) return reject(err);
65        resolve(decoded);
66      });
67    });
68  }
69

```

Joonis 12. JWT verifitseerimine.

Rakenduses on määratud, mis päringuid kasutaja teha saab. Kui kasutaja *JWT* on aegunud või on võltsitud, siis tagastatakse veakood 401, mis tähendab seda, et antud isikul ei ole lubatud päringuid teha. Kui näiteks kasutaja, kellel on roll hankija, üritab teha pakkuja alt päringut, siis tagastatakse veakood 500, mis tähendab seda, et server ei suutnud päringut täita.

Juhul kui kasutaja on end autentunud ning kõik on korras, siis kasutaja näeb ainult neid andmeid, mis on talle lubatud. Näiteks kui hange on olekus „aktiivne“, siis ei ole näha hanke korraldajale pakkujate tehtud summasisid. Kui hange on olekus „võitja valimine“, siis ei ole näha pakkuja andmeid, vaid summa ja tarnetähtaeg, mis on hankele pakutud. Ainult pakkujad näevad hanke korraldaja andmeid, sest see aitab neil pakkumisi paremini teha.

Lisaks on implementeeritud sisselogitud kasutajatele *WebSocket* ühendus. Vältimaks olukorda, kus kõik saavad end serveriga ühendada on jällegi kasutusel *JWT*. Server kontrollib oma salajase võtme abil, kas see on ikka selle serveri poolt väljastatud *JWT*. Kui on, siis luuakse ühendus serveri ja kliendi vahel (Joonis 13).

```
29
30  /**
31   * Socket middleware verifies that account has access to make socket connection
32   */
33  io.use((socket, next) => {
34    const token = socket.handshake.query.token;
35    jwt.verify(token, rc.jwt.secret, (err, decoded) => {
36      if (err) {
37        return next(new Error('Socket NO Authorization'));
38      }
39      socket.user = decoded;
40      next();
41    });
42  });
43
44  io.on('connection', socket => {...});
128
```

Joonis 13. *WebSocket*'i ühenduse kaitsmine.

## 4.5 Kasutajaliides

Kasutajaliideses leiduvad ikoonid on võetud Font Awesome [14] ja Flaticon [15] lehekülgedelt. Suuremate piltide puhul on kasutatud Freepik'i [16]. Autoril on piltide kasutamiseks litsents. Kasutajaliideses olevad värvid on saanud inspiratsiooni Material UI värvi valikust. [17]

Järgnevalt on väljatoodud moodulid, mida on kasutatud kasutajaliidese ehitamisel. Lisaks seletatakse lahti moodulid, mida autor peab oluliseks. (Joonis 14)

```
5   "dependencies": {
6     "bootstrap": "^4.0.0-beta.3",
7     "font-awesome": "^4.7.0",
8     "history": "^4.7.2",
9     "jsonwebtoken": "^8.2.0",
10    "moment": "^2.22.0",
11    "prop-types": "^15.6.0",
12    "react": "^16.2.0",
13    "react-avatar-editor": "^11.0.2",
14    "react-custom-scrollbars": "^4.2.1",
15    "react-dates": "^16.3.6",
16    "react-datetime": "^2.14.0",
17    "react-dom": "^16.2.0",
18    "react-dropzone": "^4.2.9",
19    "react-redux": "^5.0.6",
20    "react-router-dom": "^4.2.2",
21    "react-router-redux": "^5.0.0-alpha.9",
22    "react-scripts": "1.1.0",
23    "react-select": "^1.2.1",
24    "reactstrap": "^5.0.0-beta.3",
25    "redux": "^3.7.2",
26    "redux-logger": "^3.0.6",
27    "redux-thunk": "^2.2.0",
28    "socket.io-client": "^2.1.0"
29  },
```

Joonis 14. Kasutajaliidese moodulid

Moodul „react-router-dom“ abil on võimalik ehitada võimas navigatsioon SPA rakenduses. See võimaldab kasutajat suunata kui tal puudub õigus mingisugust lehte vaadata. Kui URL muutub, siis seda kasutades ei pea tervet lehte uuesti laadima, vaid

määratud komponendi, mis selle muutust kontrollib. Moodul „history“ on veebilehitseja ajalugu.

Moodul „redux“ loob hoidla (*store*), kus tegeletakse rakenduse andmete talletamisega. Kui mõni komponent tahab sellest hoidlast andmeid küsida, siis selleks on vaja moodulit „react-redux“. Selle mooduli abil saab panna komponendi hoidla muudatusi jälgima. Igakord kui muudetakse andmeid selles hoidlas, mida antud komponent jälgib, saab ka see komponent muudetud andmed. Selleks, et teha hoidlas muudatusi tuleb saata päring (*action*) hoidlasse. Tüüpiline päring näeb välja järgmine, kus *type* ütleb, mis tüüpi päringuga on tegemist ning *payload* sisaldab andmeid, mis lähevad hoidlasse. Moodul „redux-thunk“ teeb päringute (*action*) saatmise mugavamaks. Lisaks selline moodul nagu „redux-logger“ näitab, mis päring (*action*) käivitati ning, milline on hoidla järgmine olek.

Selleks et kasutajaliidese poole peal oleks teada, sisseloginud kasutaja nimi, roll, identifikaator, siis on vaja *JWT* dekodeerida ning selle jaoks on moodul „jsonwebtoken“.

#### 4.5.1 Isiku tuvastamine mobiil-ID abil

Kui külastaja on valinud registreerimiseks mobiil-ID, siis ta peab sisestama telefoni numbri ja isikukoodi. Kui sellise isikukoodiga konto on juba olemas, siis ta ei saa end uuesti tuvastada. (Joonis 15)

The screenshot shows a web form titled "Kasutaja loomine" (User creation). Below the title is a link "Logi sisse" (Log in) for existing users. There are three radio button options for authentication: "Mobiil-ID" (selected), "Smart-ID", and "ID-Kaart". Below these are two input fields: "Sisesta oma telefoni number" (Enter your phone number) with the value "+37200000766" and "Sisesta oma isikukood" (Enter your personal code) with the value "60001019906". At the bottom is a blue button with a lock icon and the text "Edasi" (Next).

Joonis 15. Kasutaja loomine mobiil-ID abil

#### 4.5.2 Külastaja tuvastas end mobiil-ID abil

Kui mobiil-ID abil isiku tuvastamine oli edukas, siis saab ta sisestada e-posti aadressi ning parooli. (Joonis 16)

### Kasutaja loomine

Kas sul on juba kasutaja? [Logi sisse](#)

E-post	Nimi
<input type="text" value="ssven1995@gmail.com"/>	<input type="text" value="MARY ÄNN O'CONNÉŽ-ŠUSLIK TES"/>
Parool	Isikukood
<input type="password" value="....."/>	<input type="text" value="60001019906"/>
Parool uuesti	Telefon
<input type="password" value="....."/>	<input type="text" value="+37200000766"/>

 Loo kasutaja

Joonis 16. Mobiil-ID abil isiku tuvastamine

#### 4.5.3 Kasutaja registreerimine õnnestus

Kui külastajal õnnestus kasutaja registreerida, siis saadetakse konto aktiveerimiseks juhend e-posti aadressile. (Joonis 17, 18)

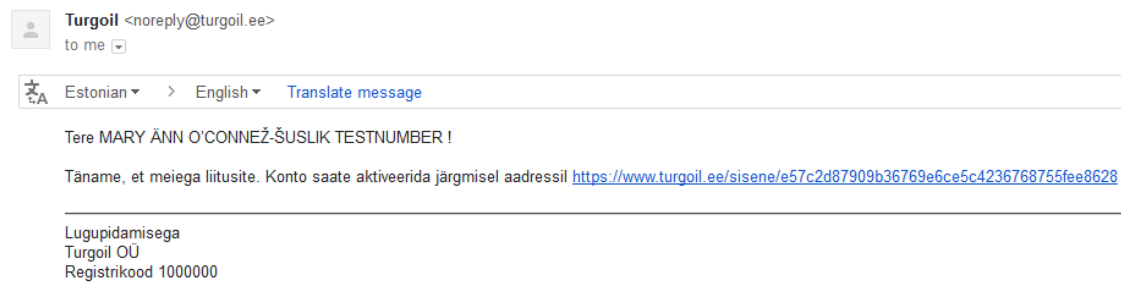
### Kasutaja loomine

Kas sul on juba kasutaja? [Logi sisse](#)



Kasutaja on registreeritud! Selleks, et see aktiveerida järgi juhiseid, mis on saadetud sinu e-posti aadressile.

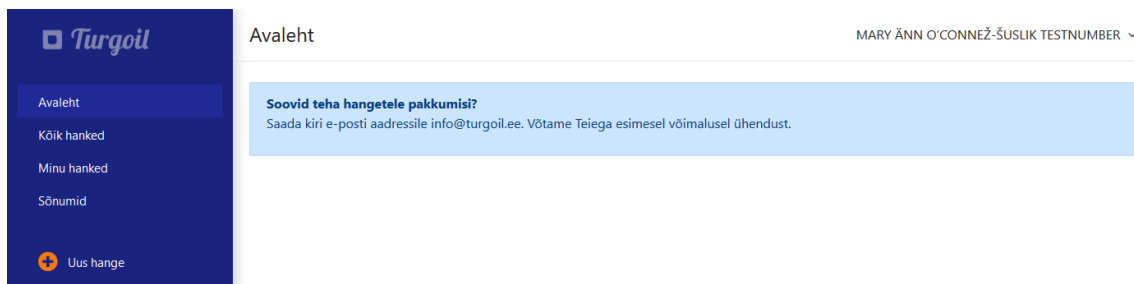
Joonis 17. Registreerimine õnnestus



Joonis 18. Sõnum konto aktiveerimine

#### 4.5.4 Kasutaja on sisse loginud

Kui kasutaja on sisse loginud, siis on tal võimalik korraldada hankeid eraisikuna. Selleks, et korraldada hankeid juriidilise isikuna, peab ta muutma oma seadeid, lisades sinna ettevõtte. Kui külastaja soovib teha vedelkütuse hangetele pakkumisi, siis ta peab avaldama soovi, saates kirja aadressile [info@turgoil.ee](mailto:info@turgoil.ee), mille tulemusel lisatakse hankija manuaalselt pakkujaks. (Joonis 19)



Joonis 19. Sisse loginud kasutaja

#### 4.5.5 Kõik hanked - hankija

Hankijal on võimalus vaadata kõiki hankeid, mis on aktiivsed või lõppenud, tasub tähele panna, et hankija ei näe teiste hankijate isiku andmeid. Hanke detailides on kütuse liik, kogus, maksetähtaeg, transport, aega lõpuni ning lisamise kuupäev. (Joonis 20)

**Turgoil** Kõik hanked MARY ÄNN O'CONNÉŽ-ŠUSLIK TESTNUMBER

Avaleht  
 Kõik hanked  
 Minu hanked  
 Sõnumid  
 + Uus hange

Aktiivsed hanked Löppenud hanked

Hankeid leitud 3 < 1 1 > Järjesta


ID 0244 - Eraisik	Kütus: bensiin98 Kogus: 222 222 L Makse tähtaeg: 17. mai 00:00 Tarne tähtaeg: 17. mai 00:00 Transport: Hankija poolt	20 h	Lisatud: 14. mai 03:02	Vaata lähemalt
ID 0245 - Eraisik	Kütus: diisel Kogus: 90 000 L Makse tähtaeg: 24. mai 00:00 Tarne tähtaeg: 19. mai 00:00 Transport: Hankija poolt	56 h	Lisatud: 14. mai 11:12	Vaata lähemalt
ID 0246 - Eraisik	Kütus: diisel Kogus: 15 000 L Makse tähtaeg: 24. mai 00:00 Tarne tähtaeg: 17. mai 00:00 Transport: Võitja poolt	10 h	Lisatud: 14. mai 19:02	Vaata lähemalt

< 1 1 >

Joonis 20. Kõik hanked - Hankija

#### 4.5.6 Seaded - konto


Kasutajal on võimalik vaadata oma konto andmeid, kus on väljatoodud: eesnimi, perenimi, isikukood, e-post, telefoni number ja kasutajaks registreerimise kuupäev. (Joonis 21)

 **Konto**


---

Eesnimi:	MARY ÄNN
Perenimi:	O'CONNÉŽ-ŠUSLIK TESTNUMBER
Isikukood:	60001019906
E-post:	ssven1995@gmail.com
Telefoni number:	+37200000766
Registreerimise kuupäev:	15. mai 03:00

---

 **Muuda parooli**

---

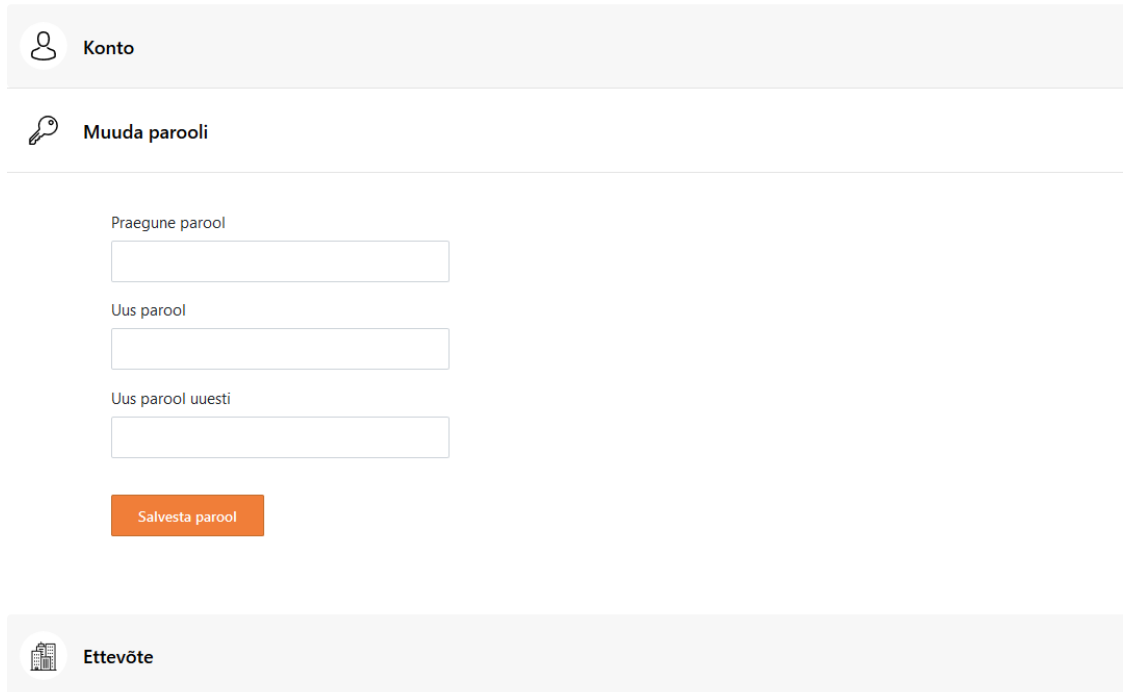
 **Ettevõte**

Joonis 21. Seaded - Konto



#### 4.5.7 Seaded - muuda parooli

Kasutajal on võimalik muuta oma konto parooli, sisestades selleks kehtiv parool ning kaks korda uus. Parooli pikkus peab olema 7 – 128 tähemärki. (Joonis 22)





The screenshot shows a user account settings interface. At the top, there is a navigation bar with two items: 'Konto' (Account) with a person icon and 'Muuda parooli' (Change Password) with a key icon. Below this, the 'Muuda parooli' section is active. It contains three input fields: 'Praegune parool' (Current password), 'Uus parool' (New password), and 'Uus parool uuesti' (New password again). Below the fields is an orange button labeled 'Salvesta parool' (Save password). At the bottom of the page, there is a footer bar with 'Ettevõtte' (Company) and a building icon.


Joonis 22 Seaded – Muuda parooli

#### 4.5.8 Seaded - ettevõtte

Kasutajal on võimalik lisada juurde ettevõtteid ning ka vajadusel neid eemaldada. Ettevõtte lisamiseks peab sisestama registrikoodi ning seejärel kontrollitakse, kas selline ettevõtte on andmebaasis. Kui sellist ettevõtet ei ole registreeritud, siis on võimalik seda registreerida. (Joonis 23)


 Konto

 Muuda parooli

 Ettevõtte

---

+

	Ettevõtte nimi	Registrikood	KMKR nr	Lisatud
	Sihtasutus Räpina Sadamad ja Puhkealad	90012640	EE101822695	-

Salvesta ettevõtte
Tühista


Joonis 23. Seaded - Ettevõtte





#### 4.5.9 Seaded – ettevõtte pildi lisamine

Kasutajal on võimalik lisada ettevõttele pilt, selleks ta valib pildi oma arvutist või lohistab selle olemasolevasse aknasse. Kui pilt lisatud saab ta suurendada sisse ja välja kasutades selleks hiire rullikut või nuppe “sisse” ja “välja”. Kui pilt ei ole õiget pidi, siis saab ta seda vajadusel ümber suunata, kas paremale või vasakule vajutades nuppe “vasakule” ja “paremale”. (Joonis 24)

Lisa ettevõtte pilt
×

Lisa pilt ▶ bitbucket\_logged\_in.PNG



 SISSE  
 VÄLJA  
 VASAKULE  
 PAREMALE

Salvesta pilt
Tühista

NB! Ettevõtte pilti näevad ainult hanke võitjad

Joonis 24. Seaded – Ettevõtte pildi lisamine

#### 4.5.10 Lisa hange

Hanke lisamisel tuleb valida hanke lõppkuupäev, mis ütleb millal hange lõppeb. Tuleks tähele panna, et hange peab kestma vähemalt 2 – 72 tundi ning peab lõppema tööpäeval. Hetkel demonstreerimiseks on see pandud 2 minutit – 72 tundi. Võitja tuleb valida kahe tunni jooksul pärast hanke lõppu, kui selle ajavahemiku jooksul ei suudeta võitjat valida, siis hange lõppeb ilma võitjata. Korraldaja alt saab valida, kas hange korraldatakse eraisikuna või ettevõttena. Kuna ettevõtte oli eelnevalt registreeritud, siis korraldatakse see selle alt. Tuleb valida veel kütuse liik. Sisestada kogus, mis on vähemalt 1000 L. Maksetähtaeg on kuupäev, millal klient on võimeline tasuma hanke võitjale. Lisainfo, mis aitab kaasa hanke korraldamisele. Transpordi korralduslik info. Tarnetähtaeg, et millal klient soovib, et see kütus talle kohale tuuakse. Lisaks veel asukoht kuhu peab võitja kütuse tooma. (Joonis 25)

Hanke lõppkuupäev	Võitja valimise tähtaeg
15.05.2018 3:21	15. mai 05:21
Korraldaja	Kütus
Sihtasutus Räpina Sadamad ja ...	Bensiin 95
Kogus	Makse tähtaeg
80000	16.05.2018 0:00
Lisainfo	
Transport	Tarne tähtaeg
Võitja poolt	16.05.2018 0:00
Asukoht	
Tartumaa	

Lisa hange

Joonis 25. Lisa hange

#### 4.5.11 Minu hanked

Pärast hanke lisamist on näha, seda hanget minu hangete all. Kuna see hange on selle isiku hange, siis ta näeb korraldaja andmeid ehk enda andmeid. (Joonis 26)

Joonis 26. Minu hanked

#### 4.5.12 Hanke andmed – Pakkuja

Kui pakkuja vaatab aktiivset hanget, millele ta ei ole varem pakkumist teinud, siis tal on võimalik see teha. Sisestades hinna ja ka tarnetähtaja, millal tal on võimalik see kütus kliendini toimetada. Kui teine pakkuja vaatab aktiivset hanget, siis tal ei ole võimalik näha teiste pakkumisi. (Joonis 27)

Joonis 27. Hanke andmed – Pakkuja

#### 4.5.13 Hanke andmed – Võitja valimine

Hanke jooksul ei ole võimalik hankijal näha pakkumiste summasid ega pakkujate andmeid. Kui hange läheb olekusse võitja valimine, siis avaldatakse talle need summad, kus ta siis saab valida hanke võitja, valides sobiva hinna. (Joonis 28)

<b>Võitja valimine</b>	Olek:	Võitja valimine
	Lisatud:	15. mai 03:18
	Lõppeb:	15. mai 03:21

---

### Pakkumised

Pakkumise ID	Summa	Tarne tähtaeg	Lisatud	
0083	9 000 €	17. mai 00:00	15. mai 03:19	
0084	8 000 €	17. mai 00:00	15. mai 03:20	

Joonis 28. Hanke andmed – Võitja valimine

#### 4.5.14 Hanke andmed – Teostamisel

Kui hankija on võitja valinud, siis on kõigile näha, milline ettevõtte hanke võitis, kuid ei ole näha tehtud pakkumise summat, selleks, et vältida konflikte hankijate ja pakkujate vahel. Sest kui teine hankija teeb sarnaste andmetega hanke ning talle tehakse teistsugune hind, siis ta ei pruugi olla rahul. (Joonis 29)

## Teostamisel

Olek: Teostamisel  
Lisatud: 15. mai 03:18  
Lõppeb: 15. mai 03:25

Hanke võitis  
**Sihtasutus Kunstihoone**

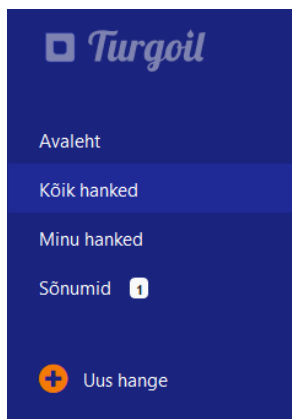
### Pakkumised

Pakkumise ID	Summa	Tarne tähtaeg	Lisatud
0083	9 000 €	17. mai 00:00	15. mai 03:19
0084	8 000 €	17. mai 00:00	15. mai 03:20

Joonis 29. Hanke andmed – Hange on teostamisel

#### 4.5.15 Sõnum

Kui hankija on võitja valinud, siis saadetakse sõnum nii hankijale kui ka pakkujale. Pärast seda nad saavad omavahel sõnumeid saata. (Joonis 30)



Joonis 30. Sõnum pärast seda kui hange läks olekusse teostamisel

#### 4.5.16 Sõnum - Hankija

Hankija näeb sõnumites teostamisel oleva hanke detaile ning vajadusel saab minna ka uuesti detail vaatesse. Lisaks saab ta saata sõnumeid pakkujale, kes hanke võitis. Näiteks leppides kokku, millal kütus kohale jõuab ja millal tasutakse. (Joonis 31)

The screenshot shows a web application interface with a search bar at the top left. The main header is 'Sihtasutus Kunstihoone'. On the left, there is a sidebar with a user profile for 'Tere, Kas Teil ole...' and a timestamp '15. mai 03:32'. The main content area is divided into two columns: 'Hanke kirjeldus' and 'Pakkumine'. The 'Hanke kirjeldus' column lists details: Kütus: bensiin95, Kogus: 80000 L, Makse tähtaeg: 16. mai 00:00, Tarne tähtaeg: 16. mai 00:00, Transport: Võitja poolt, and Lõppes: 15. mai 03:25. The 'Pakkumine' column lists: Võitja: Sihtasutus Kunstihoone, Tarne tähtaeg: 17. mai 00:00, and Hind: 8000 €. A blue button 'Vaata lähemalt' is located below the bid details. At the bottom right of the page, there is an orange button 'Saada'.

Joonis 31. Sõnum - hankija vaade

#### 4.5.17 Sõnum - Pakkuja

Sõnumeid saab ka filtreerida kasutades selleks ettenähtud lahtrit. Jututoad on sorteeritud hiljutiste sõnumite alustel. Sõnumite saatmine toimub läbi *WebSocket* ühenduse. (Joonis 32)

The screenshot shows a web application interface with a search bar at the top left. The main header is 'Sihtasutus Rápina Sadamad ja Puhkealad'. On the left, there is a sidebar with two user profiles: 'Sihtasutus Rápina Sadamad ja Pu...' with a timestamp '15. mai 03:35' and 'Sven Kõiv' with a timestamp '3. mai 21:22'. The main content area features a message from 'Tere, Kas Teil oleks võimalik tuua kütus Tartumaa, Metsa tee 15, eelistavalt tööpäeva lõpus.' Below the message is a blue button that says 'Jah on küll. Auto on seal kusagil 17:30 ajal.'.

Joonis 32. Sõnum – Pakkuja vaade

Kui pakkuja on kliendile kütuse kohale toimetanud ning raha kätte saanud, siis ta võib vajutada lõpeta hange. Selle tulemusena eemaldatakse jututuba ning hange läheb olekust teostamine olekusse lõppenud. (Joonis 33)

👁️

Sihtasutus Rápina Sadamad ja Puhkealad

---

Sihtasutus Rápina Sadamad ja Pu...  
Jah on küll. Aut... 15. mai 03:35

Sven Kõiv  
sssssssss 3. mai 21:22

### Hanke kirjeldus

Kütus:	bensiin95
Kogus:	80000 L
Makse tähtaeg:	16. mai 00:00
Tarne tähtaeg:	16. mai 00:00
Transport:	Võitja poolt
Lõppes:	15. mai 03:25

### Pakkumine

Võitja:	Sihtasutus Kunstihoone
Tarne tähtaeg:	17. mai 00:00
Hind:	8000 €

Lõpeta hange

Vaata lähemalt

Joonis 33. Sõnum - Pakkujate vaade – Lõpeta hange

## 4.6 Tagasiside

Selleks, et saada tagasiside valminud rakenduse kohta, siis pöörduti vedelkütuse hulгимүүгига tegeleva isiku poole. Tagasiside andja analüüsis veebirakendust nii hankija kui ka pakkuja vaatepunktist. Toodi välja, et oleks vaja luua parem hangete filtreerimise süsteem, mis võimaldab filtreerida aktiivseid hankeid, millele on pakkumine juba tehtud. Lisaks võiks olla rohkem reaajas teavitusi, kui näiteks uus hange lisatakse või eemaldatakse. Ning sellisele veebirakendusele võiks olla ka mobiilne versioon. Üldjoontes oli tagasiside andja seisukohal, et rakendus täidab lõputöös püstitatud eesmärgi, kuid kasutajaliides vajaks täiendamist.



## **5 Kokkuvõte**

Käesoleva bakalaureuse tööga loodi vedelkütuse hangete korraldamise veebirakendus, kus potentsiaalsed külastajad saavad registreerida ennast kasutajaks smart-ID ja mobiil-ID abil. Kasutajad saavad hallata enda korraldatud hankeid ja pakkumisi. Hankija saab korraldada hankeid era – ja juriidilise isikuna, lisada ettevõtteid, saata sõnumeid reaajas hanke võitjaga. Lisaks pakkuja saab teha pakkumisi ning lõpetada hankeid.

Töö eesmärk on saavutatud, sest hankija saab teha vedelkütuse hanke, mille tegemine on kiirem kui helistada kõikidele pakkujatele. Lisaks hankija saab valida omale turu parima hinna.

## Kasutatud kirjandus

- [1] „Express,“ [Võrgumaterjal]. Available: <https://expressjs.com/>. [Kasutatud 04 05 2018].
- [2] „Json Web Token,“ [Võrgumaterjal]. Available: <https://tools.ietf.org/html/rfc7519>. [Kasutatud 04 05 2018].
- [3] „Node,“ [Võrgumaterjal]. Available: <https://nodejs.org/en/>. [Kasutatud 04 05 2018].
- [4] „NPM,“ [Võrgumaterjal]. Available: <https://www.npmjs.com/>. [Kasutatud 04 05 2018].
- [5] „E - Teatmik,“ [Võrgumaterjal]. Available: <http://www.vallaste.ee/index.htm>. [Kasutatud 04 05 2018].
- [6] „React,“ [Võrgumaterjal]. Available: <https://reactjs.org/>. [Kasutatud 04 05 2018].
- [7] „WebSocket,“ [Võrgumaterjal]. Available: <https://en.wikipedia.org/wiki/WebSocket>. [Kasutatud 04 05 2018].
- [8] „Logistikauudised.ee,“ [Võrgumaterjal]. Available: <http://www.logistikauudised.ee/uudised/2017/12/08/eesti-vaikeste-kutusemuujate-poordumine>. [Kasutatud 01 05 2018].
- [9] „Vedelkütus,“ [Võrgumaterjal]. Available: <https://et.wikipedia.org/wiki/Vedelkütus>. [Kasutatud 02 05 2018].
- [10] A. Siirde, „Termodünaamiline keha“.
- [11] Lember, P., Muna, E., Kergandberg, E., Pöld, P., Toomingas, C., Lember, K., Jakobsoo, M.V., Riigihanked, Greif, 2017, p. 13.
- [12] „SK ID Solutions,“ [Võrgumaterjal]. Available: <https://github.com/SK-EID/smart-id-documentation#3-general-description>. [Kasutatud 15 05 2018].
- [13] „RIK,“ [Võrgumaterjal]. Available: <http://www.rik.ee/avaandmed>. [Kasutatud 12 05 2018].
- [14] „Font Awesome,“ [Võrgumaterjal]. Available: <https://fontawesome.com/>. [Kasutatud 15 05 2018].
- [15] „Flaticon,“ [Võrgumaterjal]. Available: <https://www.flaticon.com/>. [Kasutatud 05 15 2018].
- [16] „freepik,“ [Võrgumaterjal]. Available: <https://www.freepik.com/>. [Kasutatud 15 05 2018].
- [17] „Material UI,“ [Võrgumaterjal]. Available: <https://www.materialui.co/colors>. [Kasutatud 15 05 2018].
- [18] „Redux,“ [Võrgumaterjal]. Available: <https://redux.js.org/basics/reducers>. [Kasutatud 12 05 2018].