

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Martin Kask 175705IDDR

Veebirakendus maamajade konsultatsioonide haldamiseks

Diplomitöö

Juhendaja: Toomas Lepikult
PhD

Kaasjuhendaja: Elo Lutsepp
MA

Tallinn 2021

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: Martin Kask

26.04.2021

Annotatsioon

Käesoleva lõputöö eesmärgiks on luua veebirakendus Eesti Vabaõhumuuseumi maaarhitektuuri keskuse poolt koordineeritavate maamajade konsultatsioonide haldamiseks. Loodava veebirakenduse põhiülesanneteks on nõustamisprotsessi jälgitavamaks muutmine, konsultatsioonide käigus kogutavate andmete süstematiseerimine ning olemasolevate töövoogude automatiseerimine.

Uuringu käigus leiti, et koodivaba lähenemise või turul saadaolevate lahenduste abil püstitatud eesmärgid saavutada ei ole võimalik. Töö käigus viidi läbi olemasolevate töövoogude analüüs, selgitati välja loodava süsteemi nõuded ning realiseeriti veebirakendus, millele juurdepääs on piiratud registreeritud kasutajatele. Veebirakenduse kasutajaliides realiseeriti React raamistikus kasutades Bootstrap tööriistakomplekti komponente ning tagarakendus realiseeriti Express raamistikus. Töö eesmärkide täitmist hinnatati veebirakenduse kasutajate tagasiside põhjal ning kirjeldati tulevaste edasiarenduste plaane.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 30 leheküljel, 5 peatükki, 8 joonist ja 1 tabelit.

Abstract

Web Application for the Management of Consultations of Country Houses

The goal of this bachelor thesis is to create a web application for managing country house consultations coordinated by the Estonian Open Air Museum's Center of Rural Architecture. The main tasks of the created web application are to make the counselling process more observable, to systematize the data collected during the consultations and to automate the existing workflows.

The study found that the goals set could not be achieved through a code-free approach and failed to find a suitable vendor with ready-built system. In the course of the work, an analysis of the existing workflows was performed, the requirements of the system to be created were determined and a web application to which access is restricted to registered users was implemented. The user interface of the web application was implemented in the React framework using components from the Bootstrap toolkit and the server application in the Express framework. The fulfilment of the objectives of the work was assessed on the basis of the feedback of the users of the web application and the plans for future development were described.

The thesis is in Estonian and contains 30 pages of text, 5 chapters, 8 figures, 1 table.

Lühendite ja mõistete sõnastik

| | |
|-------|--|
| API | <i>Application Program Interface</i> - rakendustarkvara liides |
| CDN | <i>Content Delivery Network</i> - sisuedastusvõrk |
| CSV | <i>Comma Separated Values</i> - toimiku formaat, kus väärtused on eraldatud komadega |
| DOM | Dokumendi objektimudel - standardiseeritud, keelest ja platvormist sõltumatu XML, XHTML ja HTML dokumentidega suhtlemise liides [1] |
| ES11 | ehk EcmaSCRIPT 2020 on üldotstarbeline programmeerimiskeel, mis on standardiks JavaScriptile, tagamaks veebilehtede töötamise erinevates veebisirvijates [2] |
| FTPS | <i>file transfer protocol over implicit TLS/SSL</i> - turvaline toimikuprotokoll toimikutele juurdepääsuks, nende haldamiseks ja edastamiseks üle krüptitud ühenduse |
| HTML | <i>Hyper Text Markup Language</i> - hüpertexti märgenduskeel, kasutatakse veebilehtede struktuuri märgendamiseks |
| HTTP | <i>Hypertext Transfer Protocol</i> - hüpertexti edastusprotokoll |
| HTTPS | <i>Hypertext Transfer Protocol Secure</i> - turvaline hüpertexti edastusprotokoll |
| JSON | <i>JavaScript Object Notation</i> - andmevahetusformaad, mis baseerub JavaScripti programmeerimiskeele alamosadel [3] |
| MAK | SA Eesti Vabaõhumuuseumi Maaarhitektuuri keskus |
| MySQL | vabataarkvaraline SQL baasil loodud relatsioonilise andmebaasi haldamise tarkvara [4] |
| PDF | <i>Portable Document Format</i> - platvormist sõltumatu elektrooniliste dokumentide vorming |
| PM2 | Protsessi haldus tarkvara Node.js rakendustele [5] |
| REST | <i>Representational State Transfer</i> - esindusliku oleku ülekanne on arhitektuuri mudel kliendi ja serveri vaheliseks suhtluseks [6] |
| SPA | <i>single page application</i> - ühe-lehe rakendus |
| SQL | <i>Structured Query Language</i> - struktureeritud päringute keel relatsiooniliste andmebaasisüsteemidega suhtlemiseks |
| URL | <i>Universal Resource Locator</i> - universaalne ressursilokaator |
| XHTML | <i>Extensible HyperText Markup Language</i> - puhtam ja rohkem XML-il põhinev HTML-i alamtüüp [7] |
| XML | <i>Extensible Markup Language</i> - standardne üldotstarbeline märgistuskeel struktureeritud info jagamiseks [8] |

Sisukord

| | |
|--|----|
| 1 Sissejuhatus | 10 |
| 2 Analüüs..... | 12 |
| 2.1 Vajaduste kaardistamine..... | 12 |
| 2.2 Alternatiivsed lahendused..... | 13 |
| 2.2.1 Google teenused | 13 |
| 2.2.2 Trello | 13 |
| 2.2.3 Alternatiivsete lahenduste analüüs | 14 |
| 2.3 Funktsionaalne prototüüp | 14 |
| 2.4 Nõuded..... | 15 |
| 2.4.1 Funktsionaalsed nõuded | 16 |
| 2.4.2 Mitte-funktsionaalsed nõuded | 18 |
| 2.5 Andmemudel | 18 |
| 2.5.1 Olemite semantika | 19 |
| 3 Teostus..... | 20 |
| 3.1 Kasutatud tehnoloogiad | 20 |
| 3.1.1 WebStorm..... | 20 |
| 3.1.2 Git | 20 |
| 3.1.3 Zone veebimajutus..... | 20 |
| 3.1.4 Docker konteinerid | 20 |
| 3.1.5 Npm | 21 |
| 3.1.6 ESLint..... | 21 |
| 3.1.7 Node.js..... | 21 |
| 3.1.8 Express.js..... | 21 |
| 3.1.9 Knex.js..... | 22 |
| 3.1.10 Log4JS-node..... | 22 |
| 3.1.11 Joi | 22 |
| 3.1.12 Sendgrid..... | 23 |
| 3.1.13 Amazon teenused..... | 23 |
| 3.1.14 Maaameti In-ADS API | 23 |

| | |
|--|----|
| 3.1.15 Mocha-sinon-chai-proxyquire | 23 |
| 3.1.16 React | 24 |
| 3.1.17 Babel ja webpack..... | 24 |
| 3.1.18 React Bootstrap ja react-bootstrap-table2..... | 25 |
| 3.1.19 Redux..... | 25 |
| 3.1.20 Jest ja react-testing-library | 26 |
| 3.2 Arhitektuur..... | 26 |
| 3.3 REST API..... | 27 |
| 3.4 Autentimine ja autoriseerimine | 29 |
| 3.5 Vaated..... | 29 |
| 3.5.1 Peavaade - nõustamiste nimekiri | 30 |
| 3.5.2 Konsultantide ja kasutajate nimekirja vaated | 32 |
| 3.5.3 Nõustamise loomise vaade | 33 |
| 3.5.4 Nõustamise detailvaade | 34 |
| 3.6 Kasutajate tagasiside..... | 37 |
| 4 Edasine tegevus | 39 |
| 5 Kokkuvõte | 40 |
| Kasutatud kirjandus | 41 |
| Lisa 1 - Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks | 45 |
| Lisa 2 - Väljavõtte statistikaameti andmetest | 46 |
| Lisa 3 - Töövoogude diagramm..... | 47 |
| Lisa 4 - Aruande ankeet..... | 48 |
| Lisa 5 - Nõustamissoovide ankeet | 49 |
| Lisa 6 - Andmebaasi mudel | 50 |
| Lisa 7 - Kasutajaliidese tühi olekupuud | 51 |
| Lisa 8 - Redux-saga koodinäidis | 52 |
| Lisa 9 - Koodi kataloogipuud struktuur..... | 53 |

Jooniste loetelu

| | |
|--|----|
| Joonis 1. Funktsionaalse prototüübi nõustamiste nimekirjavaate kuvatõmmis..... | 15 |
| Joonis 2. Funktsionaalse prototüübi nõustajate vaate kuvatõmmis..... | 15 |
| Joonis 3. Konsultatsiooni kontrolleri koodinäide | 22 |
| Joonis 4. Veebirakenduse arhitektuur..... | 27 |
| Joonis 5. Peavaade - nõustamiste nimekiri..... | 32 |
| Joonis 6. Konsultantide ja kasutajate nimekirja vaated..... | 33 |
| Joonis 7. Nõustamise loomise vaade | 34 |
| Joonis 8. Nõustamise detailvaade..... | 36 |

Tabelite loetelu

| | |
|--|----|
| Tabel 1. Ülevaade REST API päringutest | 28 |
|--|----|

1 Sissejuhatus

Aastal 2015 loodi Eesti Vabaõhumuuseumi Maaarhitektuuri keskuse (MAK) poolt vanade maamajade renoveerimiste nõustamiste paremaks koordineerimiseks võrgustik, kuhu algselt kuulusid nõustajad MTÜ Vana Maamaja-st ja MTÜ Säästva Renoveerimise Infokeskusest [9]. Aastatega on võrgustikuga liitunud nõustajaid üle-eestiliselt ning praeguseks kuulub sinna neid üle kolmekümne.

Teenust pakutakse kuni 1945. aastani valminud maapiirkondade hoonetele, mida on umbes pooled statistika ameti andmetel 2018. aastal loetelud 128 613 vastavavanuselistest eluruumidest (Lisa 2 - Väljavõte statistikaameti andmetest). Erilist huvi pärandarhitektuuri säilitamise seisukohast pakuvad aga rehielamud, mis on meie piirkonnale ainuomase lahendusena välja kujunenud unikaalne arhitektuuriline nähtus. Neist umbes 60 000-st peamiselt 16. - 19. sajandil ehitatud hoonetest on tänaseks erinevatel andmetel säilinud 5 - 10% [10].

Aastatel 2016-2017, enne käesoleva tööga seotud arenduste algust, pakuti MAK poolt nõustamisteenust ligikaudu 160-le maamaja omanikule [11]. Nõustamiste koordineerimine on ajamahukas ning keerukas, kuna selle käigus peavad MAK töötajad suhtlema nii maamaja omanike kui ka nõustamisteenust pakkuvate ekspertidega. Järjest kasvav nõustajate ring ning potentsiaalselt abi vajavate majaomanike suur arv motiveeris MAK-d otsima võimalusi töö paremaks korraldamiseks. Sellest tulenevalt pöörduiti MAK-st töö autori poole sooviga leida lahendus, mis aitaks nõustamisi efektiivsemalt ning automatiseeritumalt hallata.

Käesoleva diplomitöö peamine eesmärk oli luua veebirakendus, mis muudab nõustamisprotsessi jälgimise MAK töötajatele lihtsamaks, võimaldades neil näha, millises staadiumis iga konkreetne nõustamine antud hetkel on, et vajadusel sekkuda. Teiseks eesmärgiks oli nõustamise käigus loodavate raportite muutmine standardiseeritumaks ning kogutud andmete organiseeritum säilitamine. Selle jaoks loodi nõustajatele võimalus koostada majaomanikele kättesaadavaid raporteid, mis sisaldavad ka pilte hoonetest ja kahjustustest. Kolmandaks eesmärgiks oli automatiseerida

olemasolevaid töövooge ning vähendada MAK töötajate poolt igale konkreetsele objektile sobiva eksperdi leidmiseks kuluvat aega. Töö lisaeesmärgiks oli anda ülevaade rakenduse arendusprotsessist ning edaspidistest parendusvõimalustest.

Töö eesmärkide saavutamiseks viidi esmalt läbi intervjuud MAK töötajatega, analüüsiti olemasolevaid töövooge ning võrreldi turul leiduvaid valmislahendusi. Esmasele analüüsile tuginedes loodi veebirakenduse kasutajaliidese funktsionaalne prototüüp, mille abil kinnistati koostöös MAK töötajatega lõplike nõuete nimekiri. Prototüübi põhjal koostati andmebaasi mudel, loodi serverrakendus koos vajalike lõpp-punktidega ning realiseeriti kasutajaliides. Rakenduse sobivust probleemi lahendamiseks kontrolliti kasutajate tagasisidet analüüsides. Lõpetuseks selgitati edasiste arenduste plaane.

Töös on ingliskeelsete terminite eristamiseks kasutatud kaldkirja. Parema arusaadavuse huvides on mitmete eestikeelsete terminite järel sulgudes ära toodud nende ingliskeelsed vasted.

Käesoleva töö kasutajaliidese esialgse versiooni loomisel aitas olekuhalduse seadistamise ja autoriseerimisvaadete loomisel kaasa töö autori endine kolleeg Pipedrive OÜ-st, Mihkel Kruise.

2 Analüüs

Käesolevas peatükis kirjeldatakse arendustööle eelnenud analüüsi, selle tulemusena loodud funktsionaalseid ja mitte-funktsionaalseid nõudeid ning nõuete põhjal koostatud andmemudelit, koos olemite semantikaga.

2.1 Vajaduste kaardistamine

Vajaduste kaardistamiseks viidi läbi intervjuud MAK töötajatega ning kaardistati olemasolevad töövood. Nõustamisteenuse alustamiseks oli kolm põhilist võimalust (Lisa 3 - Töövoogude diagramm):

- Esiteks otsekontakt nõustajaga läbi maamaja.eu portaali, mille puhul majaomanik saadab valitud nõustajale e-kirja, milles kirjeldab oma soovi. Nõustaja annab keskusele teada, et läheb teenust osutama või et ta ei ole seda võimeline teostama. Kui nõustaja ei saa teenust osutada, teavitab ta sellest kasutajat ning suunab ta keskuse töötaja poole.
- Teiseks on otsekontakt MAK töötajaga, kes soovi saamise järgselt leiab sobiva nõustaja ning lepib mõlema osapoolega kokku nõustamise toimumise.
- Kolmandaks võimaluseks on nõustamised, mis on teostatud läbi otsekontakti nõustamiskspertidega, millest nõustajad ise keskust informeerivad.

Keskuse töötajad vajavad ülevaadet käimasolevatest nõustamistest, et paremini suunata uusi nõustamissoove ning vajadusel sekkuda, kui nõustamise teostamine on pikale veninud.

Kõikide nõustamiste järgselt täidab nõustaja ankeedi, mis on lahendatud vabatekstilise Excel arvutustabeli kujul (Lisa 4 - Aruande ankeet). Täidetud ankeet edastatakse nõustaja poolt keskusele. Rakendatud ankeet ei määra ette mingeid piiranguid, mis kujul andmeid salvestatakse, puuduvad otsesed seosed kahjustuste ja hoonete piltide vahel ning kuna kõik aruanded on eraldiseisvad dokumendid, siis nendes leiduvate andmete omavaheline võrdlemine toimub käsitsi. Kogutavaid andmeid soovitakse kasutada tulevikus erinevates uurimistöodes.

Nõustajate ja majaomanikega suhtlemine on ajamahukas protsess, kuna nõuab mitmeid kirju erinevate osapoolte vahel. Keskuse töötajate sooviks on, et enamus e-kirjade saatmistest toimuks automaatselt vastavalt töö etappidele.

Nõustamisvõrgustiku poolt pakutavat teenust saab kasutada iga maamaja omanik ainult ühekordselt. Edaspidised nõustamised ei ole enam kompenseeritavad ning sellisel juhul peab majaomanik teenuse saamiseks ise otse nõustaja poole pöörduma. Keskuse jaoks on oluline vältida topelt nõustamiste tekkimist. Juhul kui maja on vahepeal omanikku vahetanud on keskusel võimalik olemasolev aruanne uuele omanikule edastada.

2.2 Alternatiivsed lahendused

2.2.1 Google teenused

Paljude toimingute jaoks, mis nõuavad andmete sisestamist, sorteerimist ning haldamist, kasutavad avaliku sektori töötajad tabelarvutusprogramme nagu Microsoft Excel, LibreOffice Calc ja Google Sheets [12, 13, 14]. Tabelarvutusprogrammide eeliseks on see, et nad on enamasti juba kasutajate arvutites olemas ning nende kasutuselevõtt ei vaja ümberõpet ega täiendavaid lisakulutusi. Ka varasemalt loodud ning kasutuses olev ankeet on Excel vormingus.

Esialgse kiire parendusena kaalus töö autor olemasoleva ankeedi välja vahetamist Google Forms ankeedi vastu, mis edastaks oma andmed Google Sheets dokumenti. Google Forms on Google LLC poolt pakutav veebirakendus, mis võimaldab hõlpsasti luua küsitlusi ja ankeete ning ühildub väga hästi teiste Google LLC teenustega [15]. Ka oleks võimalik e-kirjade saatmist automatiseerida ankeedi täitmise järgselt läbi GmailAPI [16].

2.2.2 Trello

Nõustamiste etappide jälgimiseks ja kuvamiseks kaaluti Kanban töölaua kasutusele võtmist. Kanban on nn. „kõhnast“ (*lean*) tootmiskontseptsioonist välja kujunenud töövoogude juhtimismetoodika, mida tänapäeval kasutatakse laialdaselt ka tarkvaraarenduses [17]. Kanban töölaud kujutab endast tahvlit, kus erinevad tööühikuid (nõustamised) on jagatud etappide kaupa veergudesse. Üheks populaarseks veebirakenduseks, kus on võimalik Kanban stiilis töölaudu luua, on Atlassian Corporation Plc poolt pakutav Trello [18]. Trello eeliseks teiste sarnaste toodete ees on

laialdane liidestuste olemasolu, sealhulgas ka liidestumine läbi Zapieri Google Sheets teenusega, mis võimaldab kahepoolset suhtlust [19].

2.2.3 Alternatiivsete lahenduste analüüs

Töö autor ei leidnud terviklikke valmislahendusi, mis pakuks kõiki soovitavaid funktsionaalsusi nagu ühtne kasutajate haldus, liidestumine maaametiga ja töövoogude kuvamine ning automatiseerimine. Alternatiivina arutleti MAK töötajatega koodi-vaba lahendust, mis hõlmaks Google teenuste ja Trello kasutuselevõttu. Välja pakutud lahendus oleks aidanud arenduskulutustelt kokku hoida, kuid kahjuks ei rahuldanud kõiki keskuse vajadusi.

Google Forms ei võimalda selliste korduvate väljade loomist, kus lisatavate objektide arv ei ole kindlaks määratud ning mis sisaldavad mitmeid alamvälju. Lahendusena oleks pidanud kasutama eelgenereeritud tühjade väljadega ankeeti, mis oleks teinud teenuse kasutamise nõustajatele tülikaks ning veaohtrikuks.

Koodivaba lahendus oleks ka kasutajatele olnud keerukam hallata ning kasutada, kuna teenuse vaated oleksid jagunenud mitme erineva teenusepakkuja vahel ning nõudnud eraldi autentimist. Teenuse jagamine mitmete kolmandate osapoolte teenuste vahel oleks tekitanud lisaraskusi turvalisuse ja käideldavuse tagamise seisukohast.

2.3 Funktsionaalne prototüüp

Arendustegevuste alustamisele eelnevalt koostati loodava veebirakenduse funktsionaalne prototüüp, kasutades Axure RP veebirakendust. Axure RP pakub võimalust luua projekti kondikava, kus kasutajaliidese tähtsamad komponendid töötavad samalaadselt päris rakendusega kasutades libaandmeid ilma koodi kirjutamata [20]. Vaadete visandamisel võeti arvesse tulevaste kasutajate eelnevat laialdast kogemust tabelarvutusprogrammidega ning andmete kuvamine teostati valdavalt tabelite kujul. Loodava rakenduse jaoks koostati kahe põhilise vaate prototüübid – nõustamiste ja nõustajate loendivaated (Joonis 1, Joonis 2). Nõustajate vaates on võimalik nõustajaid lisada ning näha olemasolevate nõustajate andmeid. Nõustamiste vaates on näha nõustamise andmeid, olekut ning nõustamisega tegelevat nõustajat, nõustamisi on võimalik lisada ja kustutada. Nõustamisele on võimalik lisada nõustaja või olemasolev nõustaja sealt eemaldada. Lisaks loodi kujundus maamaja.eu kodulehel asetsevale

2.4.1 Funktsionaalsed nõuded

- Veebirakendus on võimeline vastu võtma nõustamissoove, mis sisestatakse kas maamaja.eu portaali või klientrakenduse kaudu.
- Igal nõustamissoovil on staatus, vastavalt nõustamise etapile milles ta parasjagu viibib – „Esitatud“, „Pakkumisel“, „Töös“, „Lõpetatud“, „Kinnitatud“ ja „Keeldutud“.
- Kasutajaliideses on nõustamissoove võimalik vastavalt staatusele filtreerida.
- Nõustamissoovi katastritunnuse põhjal genereeritakse nõustamissoovile asukoha aadress. Katastritunnuse unikaalsus andmebaasis kontrollitakse ning visualiseeritakse klientrakenduses.
- Veebirakendus on võimeline saatma automatiseeritud e-kirju kasutajatele. Näiteks nõustaja määramisel saadetakse e-kiri kinnituslingi ning nõustamissoovi andmetega nõustajale.
- Kasutajaliides võimaldab nõustamissoove filtreerida asukoha (maakond), staatuse ja määratud nõustaja põhiselt.
- Kasutajaliides võimaldab nõustamissoove sorteerida viimase tegevuse toimumise kuupäeva järgi.
- Kasutajaliidese kaudu on võimalik lisada, muuta ja kustutada nõustamissoove.
- Kasutajaliideses on olemas nõustamissoovi detailvaade, mille kaudu on võimalik näha ja muuta nõustamise andmeid.
- Veebirakendus on võimeline talletama nõustamise täpsemaid detaile, nõustamise objektile paikneva(te) hoone(te) põhiselt. Igale nõustamisele on võimalik lisada üks või mitu hoonet.
- Igale hoonetele on võimalik määrata vähemalt kirjeldus, tüüp ja tarind.
- Igale hoonetele on võimalik lisada üks või mitu nõustamise käigus uuritavat kahjustust.

- Igale kahjustusele on võimalik määrata vähemalt tüüp, kirjeldus ja soovitus.
- Igale hoonele on võimalik lisada kuni 10 pilditoimikut. Igale detailile on võimalik lisada kuni 5 pilditoimikut. Piltide maksimaalne suurus 10MB ja minimaalselt on toetatud vähemalt JPEG toimiku formaat.
- Igal pildil on järjekorranumber ning on võimalik juurde lisada pealkiri.
- Nõustamissoovitusi on võimalik kasutajaliidese kaudu eksportida CSV toimiku formaati.
- Veebirakendusel on kolme tüüpi kasutajaid – administraatorid, tavakasutajad ja nõustajad.
- Kõikidel kasutajatel on võimalik veebirakenduses autentida kasutades e-posti aadressi ja salasõna kombinatsiooni.
- Salasõna meeldetuletus on võimalik tellida registreeritud e-posti aadressile.
- Administraatoritel on õigus süsteemi uusi kasutajaid luua.
- Administraatorid saavad lisaks teha kõiki toiminguid, mis on lubatud tavakasutajatele ja nõustajatele.
- Tavakasutajad saavad lisaks teha kõiki toiminguid, mis on lubatud nõustajatele.
- Tavakasutajad saavad nõustamissoovile määrata nõustaja.
- Tavakasutajad saavad hallata (lisada, muuta, kustutada) süsteemis olevaid nõustajaid.
- Nõustajatel on võimalik rakenduses vaadata endale määratud nõustamisi, lisada informatsiooni enda käimasolevatele nõustamistele ja muuta enda profiili andmeid.
- Nõustajal on võimalik poolelioleva nõustamise andmeid muuta ning salvestada.
- Nõustaja saab nõustamise valmides esitada nõustamise tavakasutajatele üle vaatamiseks. Esitatud nõustamisi nõustaja tagantjärele muuta ei saa.

- Tavakasutaja saab nõustamise tagasi nõustajale saata parandamiseks või selle kinnitada vastuvõetuks.
- Kinnitatud nõustamisest genereeritakse raport ning saadetakse link maamaja omanikule, nõustajale ja teadete lugejaks registreerunud administraatoritele.
- Raportid on ligipääsetavad vähemalt 60 päeva.
- Kasutajaliides on eestikeelne.

2.4.2 Mitte-funktsionaalsed nõuded

- Veebirakendus majutatakse Zone.ee keskkonnas maamaja.eu lehe alamdomeenina.
- Andmebaas on ehitatud MariaDB (Zone.ee-s kasutusel olev andmebaasi tüüp) andmebaasi tüüpi kasutades.
- Rakendatakse vigade logimise süsteemi.
- Süsteemi algkood hoitakse privaatses tarkvara versioonikontrolli varamus.
- Veebirakenduse API on kaetud automaattestidega vähemalt 50% ulatuses.
- Kõik süsteemis tehtavad toimingud salvestatakse koos toiminguteostaja informatsiooniga ning ajatempliga.
- Andmeid tavaolukorras lõplikult ei kustutata vaid vajadusel arhiveeritakse.

2.5 Andmemudel

Rakenduse põhiolendid on kasutaja (*user*), nõustamine (*consultation*), katastriüksus (*cadastral_unit*), hoone (*building*), sessioon (*session*), toimik (*file*) ja kahjustus (*detail*). Kuna nõuetes eestikeelsete muutujate kasutamise nõuet ei olnud, kasutati koodis ning andmebaasis läbivalt ingliskeelseid termineid (Lisa 6 - Andmebaasi mudel).

Enamus olemitel on lisamise ja muutmise ajatemplid (*add_time*, *update_time* vastavalt), viide nende toimingute teostajale (*created_by*, *updated_by* vastavalt) ning tõeväärtus

tüüpi muutuja pehme kustutamise staatuse hoidmiseks (*active_flag*). Toimingu teostajate seoseid olemite semantika kirjelduses eraldi välja ei tooda.

2.5.1 Olemite semantika

- Kasutajaid on kolme erinevat tüüpi – administraator, tavakasutaja ning konsultant.
- Kasutajal võib olla mitu sessiooni.
- Igal katastriüksusel on unikaalne katastritunnus - numbriline kood, mis on ette nähtud katastriüksuste identifitseerimiseks ja andmete sidumiseks teiste registritega [21].
- Igal nõustamisel on katastriüksus.
- Nõustamisel võib olla nõustaja.
- Nõustamisel võib olla mitu hoonet.
- Hoonel võib olla mitu kahjustust.
- Hoonel võib olla mitu toimikut.
- Kahjustusel võib olla mitu toimikut.
- Kasutajal võib olla mitu toimikut.

3 Teostus

3.1 Kasutatud tehnoloogiad

Käesolevas alampeatükis antakse põgus ülevaade veebirakenduse loomiseks kasutatud peamistest tehnoloogiatest. Tehnoloogiate valimisel lähtus töö autor oma kogemustest ja oskustest, majutusteenuse pakkuja võimalustest ning levinud praktikatest.

3.1.1 WebStorm

Kogu arendustöö teostamiseks on läbivalt kasutatud JavaScripti jaoks spetsiaalselt mõeldud arendustarkvara WebStorm [22]. WebStorm on integreeritud programmeerimiskeskond, mis toetab nii Node.js kui ka React.js spetsiifilist automaatset koodilõpetamist, omab käsurea ja versioonihalduse integratsioone, koodi refaktoreerimise toetamise tehnoloogiaid jpm.

3.1.2 Git

Projekti koodi versioonihalduseks kasutati Git vabavaralist tarkvara ning koodi privaatset hoidlat majutatakse GitHub koodihoidlas [23] [24].

3.1.3 Zone veebimajutus

Nõuetest tulenevalt majutati veebirakendus Zone Media OÜ (edaspidi Zone) poolt hallataval virtuaalserveril [25]. Node.js rakendus käivitatakse läbi PM2 protsessihaldus tööriista vastavalt Zone-i juhistele [26]. Andmebaasina kasutatakse Zone-i poolt pakutatavat MySQL-i relatsioonilise andmebaasi vabavaralist varianti MariaDB, versiooni 10.2.32. Koodimuudatuste juurutamine Zone serverisse teostatakse manuaalselt üle FTPS ühenduse.

3.1.4 Docker konteinerid

Kuigi Zone ei toeta hetkel veel Docker konteinereid, on need arenduskeskkonnas kasutuses, tagamaks kasutatavate Node.js ja andmebaasi versioonide samasust erinevate keskkondade vahel. Docker konteinerid on standardsed tarkvara ühikud, mis võimaldavad rakenduse pakendada kokku koos kõikide sõltuvustega ning paigaldada praktiliselt igasse serverisse, mis seda toetab [27].

3.1.5 Npm

Projektis kasutatavate JavaScript teekide paigaldamiseks ning haldamiseks kasutati Node.js-ga vaikumisi kaasa tulevat npm käsuviibatööriista ja npm registrit [28]. npm käsuviibatööriista kasutati ka erinevate käskluste käivitamiseks nagu näiteks testide käivitamine, koodi kompileerimine, lokaalse serveri käivitamine, koodianalüüsimine jne.

3.1.6 ESLint

Potentsiaalsete vigade automaatseks tuvastamiseks, problemaatiliste mustrite kasutamise vältimiseks ning ühtse stiili säilitamiseks rakendati staatilist koodianalüüsi tööriista ESLint [29]. Parimate praktikate järgimiseks kasutati ESLint tööriista jaoks Airbnb poolt välja töötatud reeglistikku [30].

3.1.7 Node.js

Projekti tagarakendus (*back-end*) põhineb Node.js-il. Node.js on asünkroonne sündmustepõhine JavaScripti käitussüsteem, mis võimaldab JavaScripti käivitada ilma veebisirvijata ja on loodud peamiselt skaleeritavate võrgurakenduste ehitamiseks [31]. Käesoleva töö puhul oli Node.js valimisel otsustavaks autori pikaajaline kogemus antud tehnoloogiaga ning võimalus kirjutada serverrakendust ning kasutajaliidest samas programmeerimiskeeles.

Töös kasutatav Node.js versioon on 14.15.1, mis sisaldab enamusi JavaScripti uusimaid funktsionaalsusi vastavalt ECMAScript 2020 (ES11) standardile [32]. Üks kasulikumaid nendest on asünkroonne/oota (*async/await*) süntaks, mis lubab kirjutada lihtsamat ja kergemini loetavat asünkroonset koodi võrreldes tagasihelistamisega (*callback*) või lubaduste ahelatega (*promise chains*) [33]. Näitena sellest on Joonis 3 toodud konsultatsiooni kontrolleri koodinäide, kus tehakse kaks päringut andmebaasi, kuid kogu loogika koos erindite käsitlemisega, mahub hõlpsalt kahekümnele reale.

3.1.8 Express.js

Tagarakenduse serveri loomisel kasutati Ekspress.js-i, mis on kiire, minimalistlik ja enimkasutatav veebirakenduste raamistik Node.js-le [34]. Express.js pakub mugavat vahekihti HTTP serveri käitamiseks, marsruutimise haldamiseks, päringute töötlemiseks ning erinevate taaskasutatavate vahefunktsioonide (*middleware*) lisamiseks.

```

1  const asyncController = require('.././lib/async-controller');
2  const errorResponse = require('.././lib/error-response');
3
4  const { getConsultation } = require('.././queries/consultation');
5  const { getBuildingsByConsultationId } = require('.././utils/buildings-util');
6  const { E_DB_QUERY, E_NO_RECORD, E_VALIDATION } = require('.././lib/response-codes');
7
8  module.exports = asyncController( controller: async (req, res) => {
9    · const consultationId = parseInt(req.params.consultationId, radix: 10);
10
11   · if (!consultationId) {
12     · return errorResponse(E_VALIDATION)(new Error('No consultation id provided.));
13   · }
14
15   · const consultation = await getConsultation(consultationId)
16     · .catch(errorResponse(E_DB_QUERY));
17
18   · if (!consultation || (req.user.isConsultant && req.user.id !== consultation.consultant_id)) {
19     · return errorResponse(E_NO_RECORD)(new Error('Consultation not found.));
20   · }
21
22   · const buildings = await getBuildingsByConsultationId(consultationId)
23     · .catch(errorResponse(E_DB_QUERY));
24
25   · consultation.buildings = buildings || [];
26
27   · return res.apiResponse(consultation);
28 }));

```

Joonis 3. Konsultatsiooni kontrolleri koodinäide

3.1.9 Knex.js

Tagarakenduse ja andmebaasi vahelisel suhtlusel kasutati vahekihina Knex.js teeki. Knex.js on SQL päringute ja skeemide ehitamise teek, mis toetab transaktsioone, päringute voogedastamist, andmebaasiühenduste ühendamist (*pooling*) ning standardiseeritud vastuseid olenemata kasutatavast andmebaasi tüübist või dialektist [35]. Knex.js kasutamine võimaldab konfiguratsioonitoimiku uuendamise vaevaga vahetada andmebaasi tüüpi ilma ülejäänud koodi muutmata.

3.1.10 Log4JS-node

Tagarakenduses toimuvate sündmuste ja erindite logimiseks kasutati log4JS-node teeki, mis võimaldab lisaks kirjete toimikutesse logimisele neid edastada ka erinevaid kanaleid pidi [36]. Käesoleva töö puhul oli üheks neist kanalitest Slack sõnumside teenus, mis võimaldab serveris toimuvat reaajas jälgida läbi Slack mobiilirakenduse teavituste [37].

3.1.11 Joi

Kõikide tagarakendusse saadetud muutmis- ja loomispäringute korrektsust kontrolliti Joi skeemide abil. Joi on JavaScripti objektide skeemide kirjeldamise ning valideerimise teek, mis lihtsustab päringute kontrollimist ning tagastab valideerimise ebaõnnestumise korral informatiivseid erindeid [38].

3.1.12 Sendgrid

Üks põhilisi viise, kuidas loodud veebiteenus aitab säästa MAK töötajate aega, on automaatsete e-kirjade saatmine. Selleks kasutab tagarakendus SendGrid API-ga liidestumist [39]. Hetkel on kasutuses nende tasuta plaan, mis võimaldab saata kuni 100 e-kirja päevas, mis rahuldab MAK praegused vajadused. SendGrid-i poolt pakutava teenuse kasutamine tagab kirjade parema kohale jõudmise, annab ülevaate saadetud/kättesaadud e-kirjade loendist ning aitab kokku hoida arenduse ajast, mis kuluks meiliserveri ülesseadmisele ja haldamisele.

3.1.13 Amazon teenused

Kuna kogutav pildimaterjal on muuseumi seisukohast ülitähtsa väärtusega, siis saadetakse kõik pilditoimikud serverisse laadimise järgselt ka Amazon S3 pilveteenuse andmepange (*bucket*) [40]. Säästmaks tagarakenduse ressursse ja võimaldamaks piltide kiiremat laadimist, on pilditoimikute kuvamine kasutajatele lahendatud läbi Amazon CloudFront CDN-i (sisuedastusvõrk) [41]. CDN-i kasutamine võimaldab ka paremat andmepange turvalisust, kuna ligipääs S3 andmepangele on ainult rakendusel endal ja CloudFront teenusel.

3.1.14 Maaameti In-ADS API

Katastritunnuse põhjal aadressi andmete otsinguks kasutatakse Maaameti poolt pakutavat In-ADS API-d [42]. Aadressi andmed päritakse Maaametilt ühekordselt nõustamise loomisel ning talletatakse andmebaasi.

3.1.15 Mocha-sinon-chai-proxyquire

Tagarakenduse automaattestimiseks kasutatakse nelja erinevat teeki:

- Mocha – testimise raamistik, mille peamine ülesanne on testide käivitamine ja tulemuste raporteerimine [43].
- Sinon – teek, mille abil luuakse võlts-objekte ning -funktsioone [44].
- Chai – kinnituste (*assertion*) teek, mis pakub Node.js sisesele *assert* funktsioonile lisana kergemini loetavaid mugavusfunktsioone [45].

- Proxyquire – teek, mida kasutatakse Node.js sõltuvuste kaasamise funktsiooni *require* asendamiseks ning sõltuvuste üle kirjutamiseks võltsversioonidega [46].

Kirjeldatud nelja teegi kasutamine on Node.js rakenduste puhul tavapärane ning pakub arendajatele kiire ja mugava viisi hästi organiseeritud ning arusaadavate automaattestide kirjutamiseks.

3.1.16 React

Projekti kasutajaliidese tegemiseks kasutati React raamistikku ja JavaScripti süntaksit laiendavat keelt JSX. React on Facebooki ja Instagrami poolt loodud vabavaraline JavaScript raamistik, mis on mõeldud peamiselt SPA (*single page applicatio* – ühe-lehe rakendus) tüüpi kasutajaliideste loomiseks. SPA puhul laetakse kasutajaliidese kuvamiseks vajalik kood (JavaScript, HTML ja CSS) kasutaja veebisirvijassee ühe laadimisega, mis muudab kasutajakogemuse sarnaseks operatsioonisüsteemi paigaldatavete rakendustega [47]. JSX (JavaScript XML) võimaldab React komponente kirjutades kasutada HTML elemente läbisegi JavaScript koodiga ilma eraldi templiittoimikuid loomata [48]. JSX koodi ei interpreteerita otse, vaid see kompilleeritakse JavaScriptiks, täpsemalt sellest järgmises alampeatükis.

Antud projekti React kasutajaliidese algsel ülesseadmisel kasutati create-react-app mallidel põhineva SPA rakenduse loomise teegi abi, kuigi hilisemas arenduse faasis genereeritud skripte ja konfiguratsiooni toimikuid ulatuslikult muudeti [49].

3.1.17 Babel ja webpack

Koodi kompilleerimiseks ning pakendamiseks kasutati Babel-it ja webpack-i. Babel on tööriist, mis kompilleerib JSX keeles kirjutatud koodi JavaScripti koodiks, muutes JSX spetsiifilised komponendid React.createElement funktsiooni kutseteks. Babel muudab ka koodi arusaadavaks vanematele veebisirvijate või Node.js versioonidele, mis ei toeta veel uuemaid ECMAScript funktsionaalsusi [50].

Webpack on avatud lähtekoodiga JavaScripti moodulite pakendaja (*bundler*), mis kasutab sisendina sõltuvusi sisaldavaid mooduleid ning genereerib nendest pakid (*bundles*) - staatilised toimikud, mida veebiserver saab veebisirvijale edastada [51]. Webpack poolt teostatavat pakendamist on võimalik täiendada ning konfigureerida mitmete laienduste abil. Käesolevas projektis on kasutusel laiendid, mis teostavad pakside

minimaliseerimise, CSS-toimikute pakendamise ja optimeerimise ning lähtekoodi kaartide (*source map*) loomise.

3.1.18 React Bootstrap ja react-bootstrap-table2

Vaadete loomisel kasutati React Bootstrap kujundusteegi komponente. React Bootstrap on vabavaraline sõltuvustevaba React.js komponentide teek, mis sisaldab stiliseeritud adaptiivseid disainikomponente [52]. Valmis komponentide kasutamine aitab säästa arendustööks kuluvat aega ning vähendab testimise vajadust.

Kuvatavate tabelandmete kujundamiseks kasutati React Bootstrap teegiga ühilduvat react-bootstrap-table2 kujundusteeki, mis omab rikkalikku funktsionaalsust ning on kergesti konfigureeritav ning laiendatav [53].

3.1.19 Redux

Kasutajaliidese oleku haldamiseks kasutati redux raamistikku [54]. Redux puhul hoitakse kogu rakenduse olekut ühes olekupuus, mida otse muuta ei ole võimalik. Olekupuud muutmine toimub läbi taandavale funktsioonile (*reducer*) saadetavate toimingute (*actions*). Toimingute tagajärjel viib taandav funktsioon läbi olekupuud muudatused. React elemendid kasutavad olekupuud andmeid sisendina ning kuvavad kasutajale informatsiooni vastavalt uutele andmetele. Kirjeldatud ühesuunaline andmete liikumine tagab rakenduse erinevate osade vahelise lahususe ning aitab vältida tsükliliste muudatuste esinemist, kus ühe oleku muudatus põhjustab järgmise muudatuse, mis omakorda muudab esimest olekut jne. Näitena olekupuust on lisades ära toodud käesoleva töö kasutajaliidese tühi olekupuud, andmete päringu eelselt (Lisa 7 - Kasutajaliidese tühi olekupuud).

Asünkroonsete kõrvalmõjude (peamiselt suhtlus tagarakendusega) paremaks haldamiseks kasutati Redux toimingute loomisel saaga mustrit redux-saga teegi abiga [55]. Saagad kasutavad JavaScripti generaatorfunktsioone, mis võimaldavad toiminguid kirjutada n.ö. puhaste (*pure*) funktsioonidena, mida on kerge lugeda, kirjutada ning testida. Näitena on lisades ära toodud konsultatsioonile konsultandi lisamise saaga (Lisa 8 - Redux-saga koodinäidis).

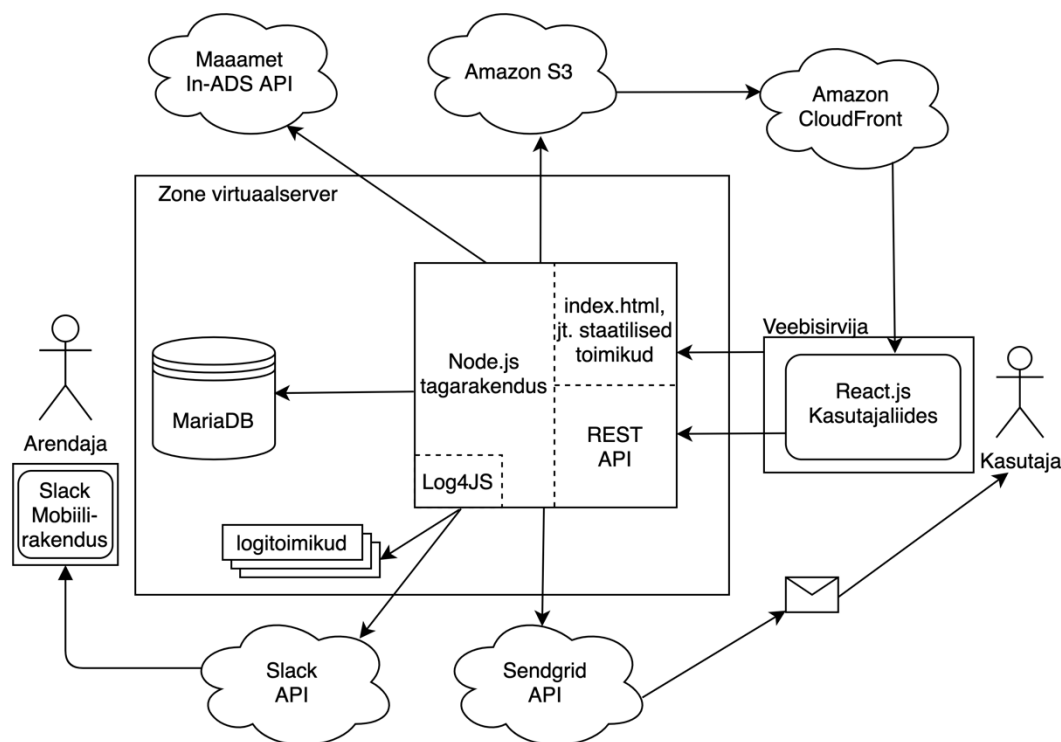
3.1.20 Jest ja react-testing-library

Kasutajaliidese testimiseks kasutati Jest testimise raamistikku ja react-testing-library teeki. Jest tuleb eelpool mainitud create-react-app kasutades automaatselt konfigureerituna kaasa ning on Facebook-i poolt loodud algselt just React.js rakenduste testimiseks. Erinevalt Mocha-st on võltsimise ja keerukamate kinnituste funktsionaalsus raamistikuga kaasas ja eraldi teeke selleks paigaldama ei pea [56]. react-testing-library lisab võimalused DOM (dokumendi objektimudel) sõlmede (*node*) pärimiseks ja nendega interakteerumiseks, eesmärgiga teha seda sarnaselt sellele, kuidas kasutaja kuvatavat dokumenti näeb ning sellel toiminguid teeb [57].

3.2 Arhitektuur

Veebirakendus realiseeriti klient-server arhitektuurimustri alusel, kus tagarakenduseks on Node.js veebiserver ning kasutajaliideseks on React.js veebileht. Veebilehe serverimine kasutaja veebisirvijale on lahendatud läbi sama tagarakenduse. Veebisirvija poolt käivitatud rakendus suhtleb tagarakendusega üle HTTPS-i (turvaline hüpertexti edastusprotokoll), pärides andmeid viimase poolt serveritava REST API kaudu. Andmed edastatakse rakenduste vahel JSON formaadis. Vastavalt REST arhitektuurile on kasutajaliidese ja tagarakenduse vaheline suhtlus olekuvaba (*stateless*) - tagarakendus ei säilita informatsiooni eelnevate päringute kohta vaid iga päringu töötlus teostatakse eraldiseisvana. Autentimise jaoks vajaliku informatsiooni hoidmise eest sessiooni märgiste (*tokens*) näol vastutab klientrakendus. Süsteemi arhitektuur koos eelpool kirjeldatud väliste sõltuvustega, on ära toodud Joonis 4.

Koodi hoitakse ühes hoidlas, jagades põhilised klientrakenduse ja tagarakenduse toimikud alamkaustadesse - vastavalt *src* ja *server*. Klientrakenduse kood on omakorda jaotatud moodulitesse (*modules*), mis koondavad sarnaste vaadete ja nendega seotud redux olekuhalduse koodi. Üle projekti kasutatavad React komponendid on koondatud *components* kausta. Tagarakenduse kood on jagatud kaustadesse funktsionaalsuse järgi. *Controllers* kaustas on rakenduse ärioloogikat teostavad kontrollid. *Middlewares* kaustas on peamiselt päringutöötlemise tegelevate vahefunktsioonide kood. *Queries* kaustas on kõik andmebaasi päringud. *Routes* kaustas on rakenduse marsruutimise loogika, kus määratakse URL-ile vastav kontrollid ning see, milliseid vahefunktsioone peab marsruudil kasutama (Lisa 9 - Koodi kataloogipuu struktuur).



Joonis 4. Veebirakenduse arhitektuur

3.3 REST API

Rakenduses on kasutusel nelja tüüpi HTTP päringumeetodeid - GET andmete pärimiseks, POST uute objektide loomiseks, PUT objektide muutmiseks ning DELETE andmete kustutamaks märkimiseks. Ülevaade loodud veebirakenduse võimalikest päringutest on ära toodud Tabel 1.

Päringute vastused on JSON kujul ning sisaldavad järgmisi muutujaid:

- *Data* - päringule vastavad andmed.
- *Success* - tõeväärtus tüüpi muutuja, mis näitab, kas päring oli edukas.
- *StatusCode* - numbriline päringu staatuse kood.
- *StatusText* - staatust kirjeldav ingliskeelne sõnum.
- *Error* - päringu töötlusel tekkinud erindi sõnum.
- *PublicError* - kasutajale kuvatav eestikeelne veateade.

Erandiks on GET /api/* päringud, mille abil teostatakse nõustamise staatuse muutmisi e-kirjades sisalduvate linkide kaudu, kus päringu vastus tagastatakse HTML kujul. Igale päringule seatakse ka HTTP staatuse kood vastavalt standardile [58], kus 200 tähistab edukat päringut, 400-499 kliendi poolset viga ja >500 serveri poolset viga.

Tabel 1. Ülevaade REST API päringutest

| Meetod | Marsruut | Kirjeldus |
|--------|--|---|
| GET | /:file_name | Kasutajaliidese pakendatud staatiliste toimikute pärimine |
| GET | /client/consultations | Nõustamiste pärimine |
| GET | /client/consultations/search | Nõustamiste otsimine |
| GET | /client/consultations/:id | Nõustamise pärimine id järgi |
| POST | /client/consultations | Uue nõustamise lisamine |
| POST | /client/consultations/:id/consultant/:id | Konsultandi lisamine konsultatsioonile |
| DELETE | /client/consultations/:id | Nõustamise kustutamine |
| PUT | /client/consultations/:id | Nõustamise uuendamine |
| PUT | /client/consultations/:id/confirm | Nõustamise staadiumi edendamine |
| PUT | /client/consultations/:id/reject | Nõustamise tagasi lükkamine |
| PUT | /client/consultations/:id/decline | Nõustamisest loobumine |
| GET | /client/users | Kasutaja pärimine |
| GET | /client/users/:id | Kasutaja pärimine id järgi |
| POST | /client/users | Uue kasutaja lisamine |
| POST | /client/users/resentAccountCreationEmail | Konto loomise e-kirja uuesti saatmine |
| PUT | /client/users/:id | Kasutaja muutmine |
| DELETE | /client/users/:id | Kasutaja kustutamine |
| POST | /client/files | Toimikute üles laadimine |
| GET | /client/files/:file_name | Toimikute pärimine |
| POST | /public/consultation | Uue nõustamise loomine |
| POST | /public/auth/login | Kasutaja autentimine |
| POST | /public/auth/logout | Kasutaja sessiooni kustutamine |
| POST | /public/auth/remind | Kasutaja salasõna meeldetuletuse tellimine |
| GET | /api/consultation/:uuid/consultant/:uuid/decline | Nõustamisest loobumine |
| GET | /api/consultation/:uuid/consultant/:uuid/finish | Nõustamise lõpetamine |
| GET | /api/consultation/:uuid/consultant/:uuid/confirm | Nõustamise staadiumi edendamine |

3.4 Autentimine ja autoriseerimine

Veebiteenusega autentimiseks on kasutusel salasõna ja parooli kombinatsioon. Kasutaja sisestab need autoriseerimisvaates, mille peale klientrakendus teeb päringu tagarakenduse REST API-sse. Tagarakendus kontrollib parooli õigsust andmebaasi salvestatud parooli räsi vastu ning loob ja tagastab korrektse kombinatsiooni puhul sessiooni. Sessioon sisaldab märgist (*token*) ja märgise kehtivusaega. Märkis salvestatakse veebisirvija lokaalsesse mälli (*local storage*) ja redux olekupuusse. Edaspidiste päringute puhul saadetakse märkis päringute päises tagarakendusse. Kasutajad ennast ise süsteemi registreerida ei saa, kuid administraatorkasutajatel on võimalus neid kutsuda. Esmane administraator luuakse süsteemi läbi automaatse andmebaasi migratsiooni, mille järgselt saab esmakasutaja endale kasutajaliidese kaudu tellida parooli meeldetuletuse.

Tagarakenduse puhul on tegevusi, mis lisaks autentimisele nõuavad ka autoriseerimist - konkreetse tegevuse lubamist päringut sooritavale isikule. Näiteks on konsultantide õigused erinevaid tegevusi teha suuresti piiratud neile määratud nõustamistega ning nõustajad ei tohi mingil tingimusel näha teiste kasutajate andmeid ega olla võimelised uusi kasutajaid või konsultante looma. Kuna autoriseerimise vajadus sõltus suuresti päritavast ressursist, siis päris ühtset autoriseerimise mehhanismi ei olnud võimalik rakendada. Näiteks Joonis 3 real 18 kontrollitakse, kas konsultant ikka küsib endale määratud konsultatsiooni. Enamus autoriseerimise kontrollimisest teostatakse siiski väljaspool kontrollereid selleks loodud taaskasutatavate vahefunktsioonide abil.

3.5 Vaated

Antud alampeatükis kirjeldatakse põhjalikumalt loodud veebirakenduse peamisi vaateid. Igal vaatel on ühine menüütulp lehe päises viidetega rakenduse teiste vaadeteni: „**Nõustamised**“ - nõustamisete nimekirja vaade; „**Kasutajad**“ - kasutajate nimekirja vaade (nähtav ainult administraatoritele), „**Nõustajad**“ - nõustajate nimekirja vaade (ei ole nähtav konsultantidele), „**Profiil**“ - kasutaja profiili vaade ja küsimärgi ikoon – abileht (Joonis 5, osa 1). Lisaks menüüs viidatud vaadetele on veebirakendusel kasutaja loomise, nõustaja loomise, nõustamise loomise ning nõustamise detailvaade, nõustamise avaliku rapordi vaade ja autentimisvaade. Väiksema funktsionaalsusega vaateid - autentimisvaade, kasutaja ja nõustaja loomise vaadet, profiilivaadet ning abilehte - pikemalt ei kirjeldata. Samuti ei kirjeldata nõustamise avaliku rapordi vaadet, mis on

sarnane nõustamise detailvaatele, ilma andmete muutmise võimaluseta. Ära toodud vaadete joonised on kasutusel veebilehe abilehel illustreerivate materjalidena.

3.5.1 Peavaade - nõustamiste nimekiri

Autentimise järgselt suunatakse kasutaja veebiteenuse pealehele, milleks on nõustamiste nimekirja vaade. Antud lehel on kasutajal võimalik saada hea ülevaade kõikidest nõustamistest. Nõustamised kuvatakse kasutajale tabeli kujul ning andmeid on võimalik sorteerida ja filtreerida. Kuvatõmmis on ära toodud Joonis 5, kus on nummerdatud punaste ringidega ära märgitud järgnevad osad:

- 1) Menüü - eelpool kirjeldatud viidetega menüütulp.
- 2) Nõustamise filtreerimine staatuse järgi - vaates näidatakse ainult selliseid konsultatsioone, mille staatused on antud valikust aktiveeritud (tumedam allavajutatud hall). **Kõik** kiirvalik - Aktiveerib filtrite vaikeväärtuse - Soovid, Pakkumised ja Töös filtrid. **Soovid** filter - Uued sisse tulnud nõustamised, mille staatus on "Esitatud". **Pakkumised** filter - Nõustajatele kinnitamiseks saadetud nõustamised, mille staatus on "Pakkumisel". **Töös** filter - Hetkel töös olevad nõustamised, millede staatus on "Töös" või "Lõpetatud". Arhiiv filter - "Kinnitatud" või "Keeldatud" staatuses olevad nõustamised, mida konsultandid enam muuta ei saa.
- 3) Nõustamiste otsimine - vaates nähtaval olevaid nõustamisi filtreeritakse selle järgi, kas nad sisaldavad otsingufraase. Fraasi vastet proovitakse leida järgnevate konsultatsiooni väljadest: staatus; katastritunnus; aadressiväljad; kirjeldus; kliendi e-post ja telefon; konsultandi nimi, telefon, e-post ja firma nimi. Detailandmete järgi (nagu ehitusaasta või hoonete andmed) hetkel konsultatsiooni otsida ei saa.
- 4) Nõustamiste alla laadimine CSV toimikuna - nupp, mis käivitab nõustamiste tabeli alla laadimise.
- 5) Lisa uus nõustamine - viide nõustamiste loomise vaatele.

- 6) Detailide näitamine - alla suunatud nokaga ikoonist saab tabelis nähtavaid andmeid laiendada. Laiendatud olekus kuvatakse nõustamise asukoha ja nõustaja täpsemad andmed ning nõustamise kirjeldus pikemalt.
- 7) Nõustamise staatus - igal nõustamisel on määratud staatus mis on üks järgnevatest: „**Esitatud**“ - uued nõustamise soovid, millele ei ole nõustajat veel määratud; „**Pakkumisel**“ - nõustamised, millele on nõustaja määratud ja e-post pakkumisega saadetud, kuid mida nõustaja ei ole veel töösse võtnud; „**Töös**“ - nõustamised mis on hetkel konsultandi poolt tegemisel; „**Lõpetatud**“ - nõustamine on konsultandi poolt läbi viidud, detailid täidetud ning edastatud administraatori või tavakasutaja poolt kinnitamiseks; „**Kinnitatud**“ - töö on tehtud ja kinnitatud, avalik raport on valmis ning osapooltele edastatud, detailide muutmine ei ole enam võimalik; „**Keeldutud**“ - administraator või tavakasutaja on otsustanud antud nõustamist mitte teostada.
- 8) Nõustaja lisamine ja eemaldamine - kui nõustamisel veel konsultanti ei ole, saab selle plussmärki vajutades lisada. Plussmärgil vajutades avaneb konsultandi lisamise valik (ei kirjeldata pikemalt antud töös). Nõustaja lisamisel nõustamisele on võimalik märkida, kas tahetakse saata ka kinnitav e-kiri nõustajale. Kui nõustaja lisatakse koos kinnitusega, siis muudetakse nõustamise staatuseks „Pakkumisel“, vastasel korral aga „Töös“ olekusse. Kui nõustamisele on juba määratud konsultant, siis saab tema miinus märgist eemaldada, mille järgselt muudetakse nõustamise staatus tagasi „Esitatud“ olekusse.
- 9) Erinevad nõustamisega seotud tegevused (vasakult paremale): **Nõustamise detailid** - viide nõustamise detail vaatesse; **Kinnita viimane tegevus** - nõustamise viimine töövoos järgmisesse staatusesse; **Lükka viimane tegevus tagasi** - nõustamise viimine töövoos eelmisesse staatusesse; **Keeldu nõustamisest** - nõustamise staatuseks "Keeldutud" määramine; **Kustuta nõustamine** - Kustutab nõustamise ning sellega seotud detailid, andmeid lõplikult ei kustutata aga nõustamist ei kuvata enam süsteemis ja seda ei ole võimalik rakendusest taastada; **Nõustamise lähtestamine** - kustutab konsultandi poolt sisestatud andmed (ehitised, kahjustused ja pildid), eemaldab konsultandi nõustamisest ja seab konsultatsiooni staatuseks "Esitatud".

| Staatust | Katastritunnus | Maakond | Kirjeldus | Klienti e-post | Klienti telefon | Konsulteerija | Tegevused |
|------------|----------------|------------------|---|------------------------|-----------------|---------------|-----------|
| Esitatud | 77301.003.0050 | Tartu maakond | Tere Pilaan on alustada vana maja restaureerimist [...] | aklient3@klient.klient | | | |
| Pakkumisel | 89901.001:0138 | Viljandi maakond | Rehemaja oma 1000-aastase arenguga on hoonetööd... | bklient@klient.klient | 123123123 | Gustav Adolf | |
| Töös | 30401.002.0227 | Harju maakond | Vaja vana maja üle vaadata. | oklient2@klient.klient | | Gustav Adolf | |
| Lõpetatud | 34202.001.0213 | Lääne maakond | Tere! Krundil on vana lagunenud rehielamu, mille k... | oklient4@klient.klient | | Gustav Adolf | |
| Kinnitatud | 71302.002:1820 | Jõgeva maakond | Sauna kerise taastamine | klient1@klient.klient | 111222333 | Gustav Adolf | |

Joonis 5. Peavaade - nõustamiste nimekiri.

1) Menüü; 2) Nõustamise filtreerimine staatuse järgi; 3) Nõustamiste otsimine; 4) Ekspordi CSV faili; 5) Nõustamise lisamine; 6) Näita detaile; 7) Nõustamise staatus; 8) Nõustaja lisamine/eamaldamine; 9) Nõustamisega seotud tegevused

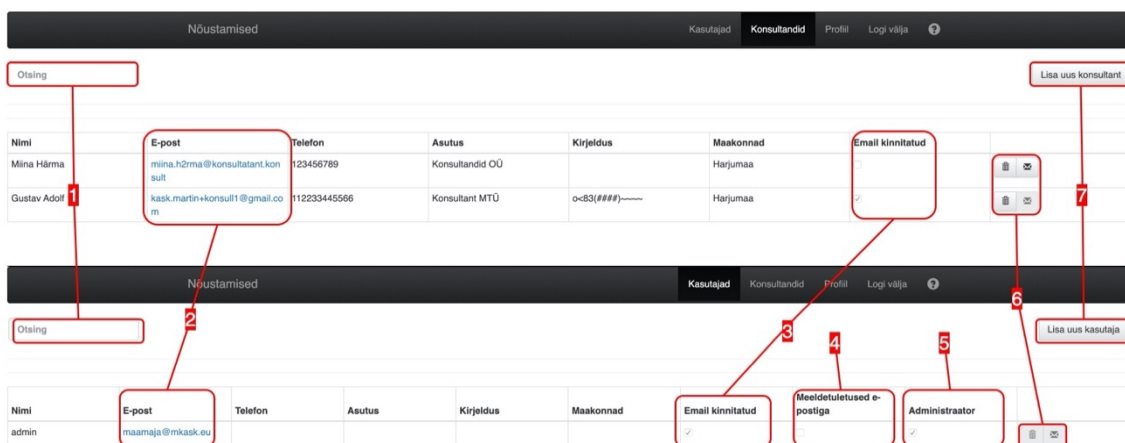
3.5.2 Konsultantide ja kasutajate nimekirja vaated

Konsultantide nimekirja vaates kuvatakse süsteemi loodud nõustajate andmed. Vaade on nähtav ainult administraatoritele ja tavakasutajatele. Kasutajate nimekirja vaates kuvatakse süsteemi loodud tavakasutajate ja administraatorite andmed ning see on nähtav ainult administraatoritele. Mõlemas vaates on võimalik muudetavate väljade peale topeltklõpsates välja väärtusi uuendada. Muudetavateks väljadeks on "Nimi", "Telefon", "Asutus", "Kirjeldus" ja "Maakonnad". Kuvatõmmis vaadetest on ära toodud Joonis 6, kus on nummerdatud punaste ringidega ära märgitud järgnevad osad:

- 1) Kasutajate otsing - vaates nähtaval olevaid kasutajaid filtreeritakse selle järgi, kas nad sisaldavad sisestatud otsingufraase.
- 2) Igal kasutajal on unikaalne e-posti aadress, mida muuta ei ole võimalik. Kui kasutaja või nõustaja e-post muutub, siis on vajalik uue loomine või selle muudatuse tegemine otse andmebaasis.
- 3) Kasutaja lisamisel saadetakse talle e-kiri, koos ajutise parooliga. Kirjas on märgitud, et kasutaja nõustub süsteemi mõllimisel MAK privaatsuspoliitikaga

[59]. Ühelegi kasutajale edasisi automaatseid sõnumeid enne ei saadeta, kui ta on oma e-kirja kinnitanud süsteemi sisse möllimise teel.

- 4) Meeldetuletused e-postiga väärtust saavad tavakasutajad ja administraatorid oma profiili alt ainult ise muuta. Kasutajatele, kellel on see aktiveeritud, saadetakse uute nõustamiste, keeldutud nõustamiste jne. kohta automaatseid teavitusi.
- 5) Väli, mis näitab, kas kasutaja puhul on tegu administraatoriga või tavakasutajaga. Antud välja saab valida ainult kasutajat luues, hiljem seda muuta ei saa.
- 6) Nupud kasutajatega seotud tegevusteks: „**Kustuta kasutaja**“ - kustutab kasutaja, kõik kasutajaga seotud nõustamised jäävad siiski alles ning samasse staadiumisse. „**Saada konto loomise email uuesti**“ - saadab kasutajale uuesti e-kirja sisse möllimiseks vajaliku teabega.
- 7) Viide uue nõustaja ja uue kasutaja loomise vaatele



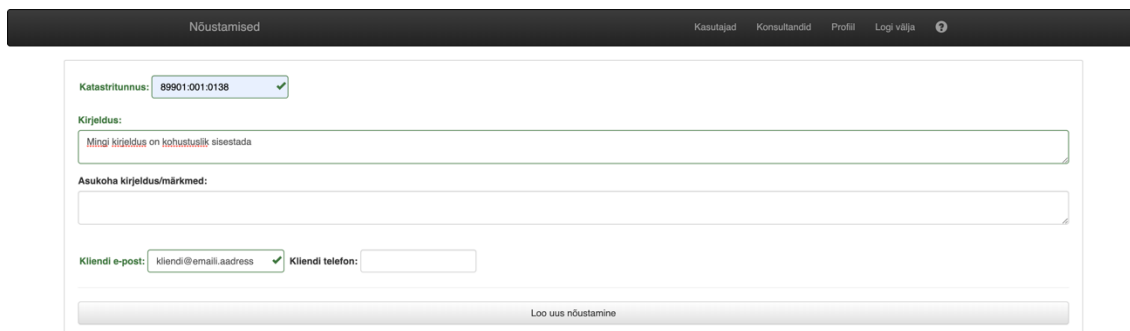
Joonis 6. Konsultantide ja kasutajate nimekirja vaated.

1) Kasutaja otsing; 2) E-post; 3) Email kinnitatud; 4) Teavitused e-postiga; 5) Administraatori väli; 6) Kasutajaga seotud tegevused; 7) Uue kasutaja lisamine;

3.5.3 Nõustamise loomise vaade

Lisaks avalikust MAK kodulehel asetsevast nõustamissoovide ankeedist automaatselt loodavate nõustamiste, on kasutajatel endil võimalik uusi nõustamisi luua läbi kasutajaliidese (Joonis 7). Nõustamise loomisel on kohustuslik lisada maamaja katastritunnus, probleemi kirjeldus ning kliendi e-kirja aadress, lisaks on võimalik märkida asukoha kirjeldus ja kliendi telefon. Andmevälju valideeritakse kasutajaliidese

ning nende sobivus kuvatakse kasutajatele. Konsultantide poolt loodud nõustamised määratakse automaatselt neile endile ning pannakse staatusesse "Töös". Tavakasutajate ja administraatorite poolt loodud nõustamistele konsultanti ei määrata ning need pannakse staatusesse "Esitatud".



Joonis 7. Nõustamise loomise vaade

3.5.4 Nõustamise detailvaade

Nõustamise detailvaade on loodud peamiselt nõustajatele nõustamist puudutavate andmete sisestamiseks. Vaates on lisaks nõustamise lisamise vaates kirjeldatud väljadele võimalik nõustamisele lisada hooneid, hoonetele kahjustusi ning hoonetele ja kahjustustele pilte. Juba loodud nõustamise katastritunnust hiljem muuta ei saa, küll aga saab redigeerida nõustamise kirjeldust, asukoha kirjeldust ja kliendi e-posti ning telefoni numbrit. Joonis 8 on ära toodud väljad ja funktsionaalsused, mida Joonis 7 ei kirjeldatud. Punasega on sellel ära märgitud järgnevad osad:

- 1) Igale nõustamisele on võimalik lisada hooneid, mille kirjeldamiseks on võimalik valida hoone tüüp, põhitarind ning ehitamisaasta ja lisada lühike kirjeldus.
- 2) Hoone eemaldamine nõustamiselt, koos sellega seotud kahjustuste ning piltidega.
- 3) Pärast pildi edukat lisamist kuvatakse see kohe ka vaates. Kõik pildid on nummerdatud vastavalt lisamise järjekorrale.
- 4) Pildi ja sellega seotud kirjelduse eemaldamine nõustamiselt.
- 5) Igale pildile on võimalik lisada lühike kirjeldus, mis vaate selguse huvides on piiratud 100 tähemärgise pikkusega. Pikema seletuse võib vajaduse korral lisada kahjustuse või hoone enda kirjelduses.

- 6) Piltide lisamine kasutajaliides on lahendatud läbi lohista-ja-kukuta (*drag-and-drop*) piirkonna, kuhu on samas võimalik ka klõpsata, avamaks operatsioonisüsteemi toimikute sirvijat lisatavate piltide valimiseks. Toetatud on toimikute laiendid ".png", ".jpg", ".gif" ja ".jpeg" ning pildi maksimaalne suurus on 10MB. Igale hoonetele on võimalik lisada kuni 10 ja kahjustusele kuni 5 pilti.
- 7) Kahjustuse kirjeldamisel on võimalik määrata kahjustunud detaili tüüp ja materjal, kirjeldada kahjustust ning anda kliendile soovitusi kahjustuse eemaldamiseks, parandamiseks, renoveerimiseks jne.
- 8) Kahjustuse eemaldamine hoonelt.
- 9) "Lisa uus hoone" lisab nõustamisele uue täitmata väljadega hoone. Igal hoonel on oma "Lisa uus kahjustus" nupp mis lisab kahjustuse vastavale hoonetele. "Salvesta andmed" nupule klõpsates saadetakse kõik nõustamise andmed tagarakendusse, kus need salvestatakse andmebaasi. Piltide puhul teostatakse toimikute laadimine eraldi ning salvestamisel saadetakse tagarakendusse ainult toimikute metaandmed, kus need seostatakse nõustamisega. Olenevalt autoriseeritud kasutajast ja nõustamise olekust on lisaks salvestamise nupule võimalik näha nuppu "Esita kinnitamiseks", mis funktsioneerib samalaadselt nõustamise nimekirja vaates olevale "Kinnita viimane tegevus" nupule, millele vajutades viiakse nõustamine staatusesse "Lõpetatud".

The screenshot shows a web application interface for managing buildings and damages. It is divided into several sections:

- 1. Hoone (Building):** A header section containing a dropdown menu for 'Hoone tüüp' (Building type) set to 'Elamu' (Residential), a dropdown for 'Hoone tarind' (Building use) set to 'rõhtpalk' (Timber frame), and a text field for 'Ehitamise aasta' (Year of construction) set to '1900'. A trash icon is visible in the top right corner.
- Hoone pildid (Building images):** A section with three image thumbnails labeled '1. Pilt', '2. Pilt', and '3. Pilt'. The first image shows a modern building. Below the images is a text input field with the placeholder 'Vajuta või lohista pildid siia...' (Click or drag images here...).
- Kirjeldus (Description):** A text input field for describing the building.
- 7. Kahjustused (Damages):** A section header for managing damages.
- 1. Kahjustus (Damage):** A sub-section containing:
 - A dropdown for 'Kahjustunud detaili tüüp' (Type of damaged detail) set to 'katus' (Roof) and a dropdown for 'Detaili materjal' (Detail material) set to 'puitlaast' (Wooden shingles).
 - Kahjustuse pildid (Damage images):** A text input field with the placeholder 'Vajuta või lohista pildid siia...'.
 - Kahjustuse kirjeldus (Damage description):** A text input field containing the text 'Katus lekib ning mädaneb.' (Roof leaks and rots).
 - Soovitus (Recommendation):** A text input field containing the text 'Vaja koogu puitlaast välja vahetada.' (Need to replace the shingles).
- 9. Nõustamisega seotud tegevused (Actions related to consultation):** A bottom section containing three buttons: 'Lisa uus kahjustus' (Add new damage), 'Lisa uus hoone' (Add new building), and 'Salvesta andmed' (Save data).

Joonis 8. Nõustamise detailvaade.

1) Hooned; 2) Hoone eemaldamine; 3) Lisatud pildid; 4) Pildi eemaldamine; 5) Pildi kirjeldus; 6) Pildi lisamine; 7) Kahjustused; 8) Kahjustuse eemaldamine; 9) Nõustamisega seotud tegevused

3.6 Kasutajate tagasiside

2020. aasta lõpus viis MAK läbi uuringu nõustajate seas, küsimaks tagasisidet kogu nõustamisvõrgustiku toimimise kohta, kus ühe teemana käsitleti ka käesolevas töös loodud veebiteenust. Küsitletud 13-st nõustajast 10 olid teenusega rahul, üks ei olnud süsteemi veel kasutada jõudnud ning 2 ei olnud teenusega rahul. Puudustena toodi välja probleeme andmete salvestamisel ja nõustamise filtreerimise keerukust. Uute lahendustena sooviti näha automaatse salvestamise võimekust ning automaatseid teavitusi selle kohta, kas majaomanik on raporti kätte saanud. Küsitluse kokkuvõttes toodi välja, et aasta jooksul võeti vastu umbes 150 nõustamissoovi ning lisaks tehti mitmeid broneeringuid järgnevasse aastasse [60].

Praeguseks on veebiteenust juba uuendatud ning mõningad salvestamisel esinenud tõrked eemaldatud. Majaomanikele saadetavate e-kirjade jalusesse on lisatud ka viide tagasisideküsitlusele, mis annab võimaluse nende arvamuste kogumiseks ning sellest lähtuvalt nõustamisteenuse kasutamismugavuse parendamiseks.

Selgitamaks, kuidas on veebirakendus täitnud töös püstitatud eesmärged, käsitleti MAK töötajaid. Elo Lutsepp, veebirakenduse peamine administraator ning nõustamisteenuse koordinaator, on välja toonud põhilise eelisena vana asjakorralduse ees suure kokkuhoiu ajakulus. Kuna detailset tööajakaardistust enne veebirakenduse loomist ei tehtud, siis on täpseid numbreid raske kindlaks teha, kuid subjektiivselt on koordinaatori arvates toimunud suur muutus. Kui varasemalt helistasid ja kirjutasid nõustajad koordinaatorile pikalt ning sagedaselt, siis praeguse korralduse juures on neil võimalik töö vastuvõetuks märkida otse e-kirjast (isiklik kirjavahetus, 14.03.2021). Lisaks on vähenenud aeg, mil esimesest kontaktist jõutakse nõustamise vormistamiseni ning olukorrad, kus majaomanikud loobuvad teenusest või seda üldse ei saa.

Rasmus Kask, MAK direktor, on andmete käideldavuse parendamist kommenteerinud järgnevalt: „Varasemalt ei jäänud nõustamistest üldse sisulisi andmeid, vaid teadmine, et töö on tehtud ja (paremal juhul) mis piirkonnas käidi. Praegune andmebaas võimaldab kahjustuste süsteemset kvalitatiivset analüüsi (kahjustuste tüübid, tarindid, esinemissagedused, ulatus jne.), mida saab kasutada hooldusjuhendite koostamiseks, muuseumide hooldustegevuste tõhusamaks planeerimiseks, teadussuundadeks ning

üldise majaomanikele suunatud kommunikatsiooni tõhustamiseks.“ (isiklik kirjavahetus 15.03.2021).

4 Edasine tegevus

Töö autor on sõlminud MAK-ga hoolduslepingu, mille kohaselt teostatakse pidevat veebirakenduse monitooringut ning vajadusel vigade eemaldamist. Käimas on ka läbirääkimised edasiste arendustööde üle, mis parendaksid kasutajakogemust ning lisaksid rakendusele funktsionaalsusi.

Peamise järgmise suure arendusena on plaanis viia lõppraportite genereerimine PDF kujule ning sessiooni halduse parendamine kasutajaliidese poolel. PDF kujul raportid parendaksid majaomanike kasutuskogemust, andes neile võimaluse nõustamise tulemust selges ning ühtses formaadis alla laadida, samuti loob see võimaluse nimetatud toimikute e-kirja teel edastamiseks. Sessiooni halduse parendamine lubaks kasutajatel aegunud sessiooni uuendamist pooleliolevat tööd katkestamata.

Uute MAK-ga arutatud funktsionaalsuste seas on palju selliseid, mida töö autori aja ja MAK ressursside piiratud tõttu koheselt ette võtta ei ole kavas. Nagu näiteks nõustamise detailvaate väljade automaatsalvestamine, erinevate toimikutüüpide toetus hoonete plaanide ja muude arhiivimist vajavate toimikute salvestamiseks, piltide ümbertöötlemine üles laadimisel toimikute suuruse kahandamiseks ja andmete eksportimise parendamine. Praegune lahendus on keskendunud just MAK töötajate ja nõustajate vahelise suhtluse efektiivsemaks muutmisele, kuid tulevikus võiks mõelda ka kuidas paremini toetada nõustaja ja maamaja omaniku vahelist suhtlust ning töö toimumise aja kooskõlastamist.

Tehnilise poole pealt sooviks töö autor liikuda JavaScripti pealt TypeScript programmeerimiskeele peale, mis on võimaldab lisada JavaScriptile staatilisi tüübikirjeldusi. Ühe uuendusena on kaalumisel ka veebiserveri raamistiku välja vahetamine uuema vastu nagu Fastify või Next.js [61] [62]. Express.js ei ole enam kõige parem valik Node.js veebiserveri loomiseks, kuna viimane uuendus antud teegile tehti aastal 2019. Kuna kasutatavaid JavaScripti teke uuendatakse loojate poolt pidevalt ning parandatakse leitud turvaauke ning vigu, siis tuleb ka neid projektis pidevalt uuendada. Lisaks jätkub töö automaattestidega kattuvuse suurendamiseks, kus eesmärgiks on saavutada kaetus 90% ulatuses nii tagarakenduse kui ka kasutajaliidese puhul.

5 Kokkuvõte

Antud töö esmaseks eesmärgiks oli luua veebirakendus, mille abil MAK töötajad saavad koordineerida vanade maamajade omanikele pakutavaid nõustamisi. Teiseks eesmärgiks oli nõustamise andmete kogumise raportite standardiseerimine. Kolmandaks eesmärgiks oli vähendada MAK töötajate ajakulu läbi töövoogude automatiseerimise.

Töö käigus analüüsiti olemasolevaid töövooge ning viidi läbi intervjuud MAK töötajatega vajaduste kaardistamiseks. Analüüsi põhjal koostati funktsionaalne prototüüp ning kirjeldati loodava rakenduse nõuded. Alternatiivina kaaluti koodi-vaba lahendust, mis kahjuks ei rahuldanud kõiki nõudeid. Rakendus loodi kasutades tagarakenduses Node.js-i koos Express raamistikuga ning kasutajaliideses React-i Bootstrap komponentidega.

Töös püstitatud eesmärgid said täidetud ning loodud rakendus lihtsustab nõustamiste haldamist, nõustajate ja MAK töötajate vahelist suhtlust, andmete kogumist ning säilitamist ühtsel kujul. Vähenes märgatavalt MAK töötajate ajakulu nõustajatega suhtlemisel.

Töö veebirakenduse edasiarendamiseks ning hoolduseks on planeeritud. Järgmise etapina on kavas lõppraportite PDF kujule viimine ning sessiooni halduse arendamine. Tehnilise poole pealt on plaanis uuendada rakenduses kasutatavaid raamistikke ja teeke ning suurendada automaatsete arvude arvu.

Veebirakendus on MAK töötajate ja nõustamisvõrgustikku kuuluvate nõustajate kasutuses alates novembrist 2018. Seisuga 25. aprill 2021 oli kokku loodud 452 nõustamist, millest edukalt oli lõpetatud 197 ja töös 135.

Kasutatud kirjandus

- [1] W3C, „What is the Document Object Model?“, 13 November 2000. [Võrgumaterjal]. Saadaval: <https://www.w3.org/TR/DOM-Level-2-Core/introduction.html>. [Kasutatud 6 Aprill 2021].
- [2] „ECMA-262, ECMAScript® 2020 language specification, 11th edition“, Juuni 2020. [Võrgumaterjal]. Saadaval: <https://www.ecma-international.org/wp-content/uploads/ECMA-262.pdf>. [Kasutatud 8 Aprill 2021].
- [3] ECMA International, „ECMA-404, The JSON Data Interchange Syntax, 2nd edition“, Detsember 2017. [Võrgumaterjal]. Saadaval: https://www.ecma-international.org/wp-content/uploads/ECMA-404_2nd_edition_december_2017.pdf. [Kasutatud 8 Aprill 2017].
- [4] Oracle Corporation, „MySQL“, [Võrgumaterjal]. Saadaval: <https://www.mysql.com/>. [Kasutatud 8 Aprill 2021].
- [5] Keymetrics Inc., „PM2“, [Võrgumaterjal]. Saadaval: <https://pm2.io/>. [Kasutatud 8 Aprill 2021].
- [6] R. T. Fielding, „Architectural Styles and the Design of Network-based Software Architectures“, University of California, Irvine, 2000.
- [7] W3C, „XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition)“, 27 Märts 2018. [Võrgumaterjal]. Saadaval: <https://www.w3.org/TR/xhtml1/#xhtml>. [Kasutatud 8 Aprill 2021].
- [8] W3C, „XML Essentials“, [Võrgumaterjal]. Saadaval: <https://www.w3.org/standards/xml/core>. [Kasutatud 8 Aprill 2021].
- [9] Eesti Vabaõhumuuseum, „Maaarhitektuuri keskus“, [Võrgumaterjal]. Saadaval: <https://maaarhitektuur.ee/meist/>. [Kasutatud 28 Veebruar 2021].
- [10] R. Alatalu, „Eesti taluarhitektuur kui rahvuspäränd. Rahvusliku taluarhitektuuri kestmine ja kaitse arengukava „Maa-arhitektuur ja -maastik. Uurimine ja hoid“ valguses“, Magistritöö, Eesti Kunstiakadeemia, Restaureerimisteaduskond, Tallinn, 2007.
- [11] K. Kallastu, „Nõustamisteenuse analüüs“, Avaldamata, Tallinn, 2017.
- [12] Microsoft Corporation, „Microsoft Excel“, [Võrgumaterjal]. Saadaval: <https://www.microsoft.com/en-ie/microsoft-365/excel>. [Kasutatud 14 Märts 2021].
- [13] The Document Foundation, „LibreOffice Calc“, [Võrgumaterjal]. Saadaval: <https://www.libreoffice.org/discover/calc/>. [Kasutatud 14 Märts 2021].
- [14] Google LLC, „Google Sheets“, [Võrgumaterjal]. Saadaval: <https://www.google.com/sheets/about/>. [Kasutatud 14 Märts 2021].
- [15] Google LLC, [Võrgumaterjal]. Saadaval: <https://www.google.com/forms/about/>. [Kasutatud 14 Märts 2021].
- [16] M. Brawler, „Learn to Magically Send Emails from your Google Form Responses“, 20 Juuni 2018. [Võrgumaterjal]. Saadaval: <https://max->

- brawler.medium.com/learn-to-magically-send-emails-from-your-google-form-responses-8bbdfd3a4d02. [Kasutatud 14 Märts 2021].
- [17] J. S. Womack, D. T. Jones ja D. Roos, The Machine that changed the world, London: Simon & Schuster, 2007.
- [18] Atlassian Corporation Plc, „<https://trello.com/about>“, [Võrgumaterjal]. [Kasutatud 14 Märts 2021].
- [19] Atlassian Corporation Plc, „Importing Data Into Trello“, 8 Jaanuar 2021. [Võrgumaterjal]. Saadaval: <https://help.trello.com/article/751-importing-data-into-trello>. [Kasutatud 14 Märts 2021].
- [20] Axure Software Solutions, Inc, „Axure RP“, [Võrgumaterjal]. Saadaval: <https://www.axure.com>. [Kasutatud 14 Märts 2021].
- [21] Riigiteataja, „Maakatastriseadus“, 06 Mai 2020. [Võrgumaterjal]. Saadaval: <https://www.riigiteataja.ee/akt/13198710>. [Kasutatud 5 Aprill 2021].
- [22] JetBrains s.r.o., „WebStorm“, [Võrgumaterjal]. Saadaval: <https://www.jetbrains.com/webstorm/>. [Kasutatud 5 Aprill 2021].
- [23] „git“, [Võrgumaterjal]. Saadaval: <https://git-scm.com/>. [Kasutatud 7 Aprill 2021].
- [24] GitHub Inc., „GitHub“, [Võrgumaterjal]. Saadaval: <https://github.com/>. [Kasutatud 7 Aprill 2021].
- [25] Zone Media OÜ, „Zone“, [Võrgumaterjal]. Saadaval: <https://www.zone.ee>. [Kasutatud 6 Aprill 2021].
- [26] Zone Media OÜ, „Node.js veebirakendused nüüd Zone virtuaalserveris – kauba peale PM2, mod_proxy ja portide suunamine“, 22 Juuli 2016. [Võrgumaterjal]. Saadaval: <https://blog.zone.ee/2016/07/22/node-js-veebirakendus-websocket/>. [Kasutatud 6 Aprill 2021].
- [27] Docker Inc., „What is a Container?“, [Võrgumaterjal]. Saadaval: <https://www.docker.com/resources/what-container>. [Kasutatud 6 Aprill 2021].
- [28] npm Inc., „npm“, [Võrgumaterjal]. Saadaval: <https://www.npmjs.com/>. [Kasutatud 6 Aprill 2021].
- [29] OpenJS Foundation, „ESLint“, [Võrgumaterjal]. Saadaval: <https://eslint.org/>. [Kasutatud 6 Aprill 2021].
- [30] Airbnb Inc., „Airbnb JavaScript Style Guide“, [Võrgumaterjal]. Saadaval: <https://github.com/airbnb/javascript>. [Kasutatud 6 Aprill 2021].
- [31] OpenJS Foundation, „Node.js“, [Võrgumaterjal]. Saadaval: <https://nodejs.org/en/about/>. [Kasutatud 5 Aprill 2021].
- [32] W. Kapke, „Node.js ES Support“, [Võrgumaterjal]. Saadaval: <https://node.green/>. [Kasutatud 5 Aprill 2021].
- [33] Mozilla Corporation, „MDN Web Docs, async function“, [Võrgumaterjal]. Saadaval: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/async_function. [Kasutatud 5 Aprill 2021].
- [34] OpenJS Foundation, „Express.js“, [Võrgumaterjal]. Saadaval: <https://expressjs.com/>. [Kasutatud 28 Veebruar 2021].
- [35] T. Griesser, I. Savin ja teised, „Knex.js“, [Võrgumaterjal]. Saadaval: <https://knexjs.org/>. [Kasutatud 5 Aprill 2021].
- [36] G. Jones, J. Huang, D. Bell ja teised, „log4js-node“, [Võrgumaterjal]. Saadaval: <https://log4js-node.github.io/log4js-node/>. [Kasutatud 7 Aprill 2021].

- [37] Slack Technologies, Inc, „Slack,“ [Võrgumaterjal]. Saadaval: <https://slack.com/>. [Kasutatud 7 Aprill 2021].
- [38] Sideway Inc., „The most powerful schema description language and data validator for JavaScript,“ [Võrgumaterjal]. Saadaval: <https://joi.dev/>. [Kasutatud 7 Aprill 2021].
- [39] Twilio SendGrid, [Võrgumaterjal]. Saadaval: <https://sendgrid.com/>. [Kasutatud 10 Aprill 2021].
- [40] Amazon Web Services, Inc., „Amazon S3,“ [Võrgumaterjal]. Saadaval: <https://aws.amazon.com/s3/>. [Kasutatud 10 Aprill 2021].
- [41] Amazon Web Services, Inc., „Amazon CloudFront,“ [Võrgumaterjal]. Saadaval: <https://aws.amazon.com/cloudfront/>. [Kasutatud 10 Aprill 2021].
- [42] Maa-amet, „In-ADS arendusjuhend,“ 26 Märts 2021. [Võrgumaterjal]. Saadaval: https://inaadress.maaamet.ee/inaadress/pdf/et/in_aadress_developer_manual.pdf. [Kasutatud 10 Aprill 2021].
- [43] OpenJS Foundation, „Mocha,“ [Võrgumaterjal]. Saadaval: <https://mochajs.org/>. [Kasutatud 7 Aprill 2021].
- [44] M. Roderic, C. Johansen, C.-E. Kopseng ja teised, „Sinon,“ [Võrgumaterjal]. Saadaval: <https://sinonjs.org/>. [Kasutatud 7 Aprill 2021].
- [45] J. Luer, V. Todorov, D. Denicola ja teised, „Chai Assertion Library,“ [Võrgumaterjal]. Saadaval: <https://www.chaijs.com/>. [Kasutatud 7 Aprill 2021].
- [46] T. Lorenz, B. Drucker ja teised, „Proxyquire,“ [Võrgumaterjal]. Saadaval: <https://github.com/thlorenz/proxyquire>. [Kasutatud 7 Aprill 2021].
- [47] Facebook Inc., „React,“ [Võrgumaterjal]. Saadaval: <https://reactjs.org/>. [Kasutatud 6 Aprill 2021].
- [48] Facebook Inc, „Introducing JSX,“ [Võrgumaterjal]. Saadaval: <https://reactjs.org/docs/introducing-jsx.html>. [Kasutatud 6 Aprill 2021].
- [49] Facebook, „create-react-app,“ [Võrgumaterjal]. Saadaval: <https://github.com/facebook/create-react-app>. [Kasutatud 7 Aprill 2021].
- [50] Open Source Collective, „Babel,“ [Võrgumaterjal]. Saadaval: <https://babeljs.io/>. [Kasutatud 7 Aprill 2021].
- [51] Open Source Collective, „Webpack,“ [Võrgumaterjal]. Saadaval: <https://webpack.js.org/concepts/>. [Kasutatud 7 Aprill 2021].
- [52] J. Jia, J. Quense, A. Shemetovskiy ja teised, „React Bootstrap,“ [Võrgumaterjal]. Saadaval: <https://react-bootstrap.github.io/>. [Kasutatud 7 Aprill 2021].
- [53] A. Fang, C. Chen ja teised, „react-bootstrap-table2,“ [Võrgumaterjal]. Saadaval: <https://react-bootstrap-table.github.io/react-bootstrap-table2/>. [Kasutatud 7 Aprill 2021].
- [54] D. Abramov, „Redux,“ [Võrgumaterjal]. Saadaval: <https://redux.js.org/>. [Kasutatud 7 Aprill 2021].
- [55] Y. Elouafi ja teised, „redux-saga,“ [Võrgumaterjal]. Saadaval: <https://github.com/redux-saga/redux-saga>. [Kasutatud 8 Aprill 2021].
- [56] Facebook Inc., „Jest,“ [Võrgumaterjal]. Saadaval: <https://jestjs.io/>. [Kasutatud 7 Aprill 2017].
- [57] K. C. Dodds ja teised, „React Testing Library,“ [Võrgumaterjal]. Saadaval: <https://testing-library.com/docs/react-testing-library/intro>. [Kasutatud 7 Aprill 2021].

- [58] R. T. Fielding, J. Gettys, J. C. Mogul, H. F. Nielsen, L. Masinter, P. J. Leach ja T. Berners-Lee, „RFC 2616 Hypertext Transfer Protocol – HTTP/1.1, June 1999,“ [Võrgumaterjal]. Saadaval: <https://www.rfc-editor.org/info/rfc2616>. [Kasutatud 11 Aprill 2021].
- [59] SA Eesti Vabaõhumuuseum, „Privaatsuspoliitika,“ [Võrgumaterjal]. Saadaval: <http://maarahitektuur.blogspot.com/p/privaatsus.html>. [Kasutatud 14 Aprill 2021].
- [60] E. Lutsepp, „Nõustajate võrgustik 2020, Kokkuvõte ja tagasiside,“ Avaldamata, Tallinn, 2020.
- [61] OpenJS Foundation, „Fastify,“ [Võrgumaterjal]. Saadaval: <https://www.fastify.io/>. [Kasutatud 18 Aprill 2021].
- [62] Vercel, Inc., „Next.js,“ [Võrgumaterjal]. Saadaval: <https://nextjs.org/>. [Kasutatud 18 Aprill 2021].
- [63] Statistikaamet, „KVE02: TAVAELURUUMID HOONE EHTUSAJA JA 2017. AASTA HALDUSREFORMI JÄRGSE MAAKONNA JÄRGI, 1. JAANUAR,“ 1 Jaanuar 2017. [Võrgumaterjal]. Saadaval: https://andmed.stat.ee/et/stat/majandus__kinnisvara__elamumajandus/KVE02. [Kasutatud 16 Märts 2021].

Lisa 1 - Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks¹

Mina, Martin Kask

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „Veebirakendus maamajade konsultatsioonide haldamiseks“, mille juhendajad on Toomas Lepikult ja Elo Lutsepp.
 - 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.

23.03.2021

¹ 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 punktidele 1.1. ja 1.2, siis lihtlitsents nimetatud tähtaja jooksul ei kehti.

Lisa 2 - Väljavõte statistikaameti andmetest

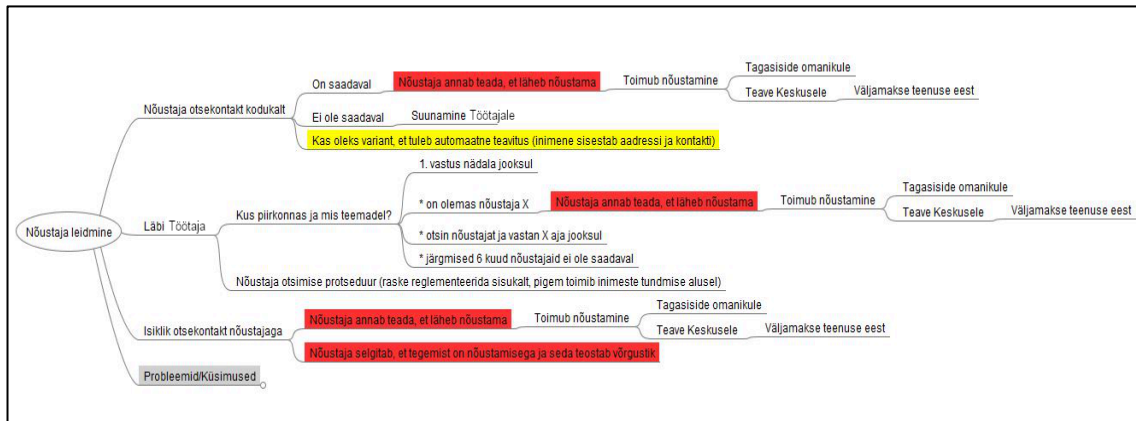
KVE02: TAVAEELURUUMID 2017. AASTA HALDUSREFORMI JÄRGSELT, 1. JAANUAR | Aasta, Maakond, Hoone ehitusaeg ning Näitaja

| | | | Tavaeluruumide arv | Tavaeluruumid, % |
|-------------|-------------------|---------------------------|--------------------|------------------|
| 2017 | Kogu Eesti | Kokku | 707779 | 100,0 |
| | | Enne 1919 | 46850 | 6,6 |
| | | 1919-1945 | 82665 | 11,7 |
| | | Ehitusaeg teadmata | 39421 | 5,6 |
| 2018 | Kogu Eesti | Kokku | 712966 | 100,0 |
| | | Enne 1919 | 46553 | 6,5 |
| | | 1919-1945 | 82060 | 11,5 |
| | | Ehitusaeg teadmata | 43691 | 6,1 |

| | |
|-------------------------|------------|
| Viimane uuendus: | 14.05.2019 |
| Ühikud: | Väärtus |
| Sisemine referentskood: | KVE02 |

(Allikas: Statistikaamet [63])

Lisa 3 - Töövoogude diagramm



(Autor: Rasmus Kask, MAK)

Lisa 4 - Aruande ankeet

Igast objektist ja kahjustusest üks foto

| | | | |
|--------------------|--|----------------|--------|
| Nõustaja | | Nõustamise aeg | |
| Address | | | |
| Nõustatav | | Kasutus | |
| Kontakt | | | |
| Omaniku plaanid | | | |
| Hoone | | Ehitusaasta | Tarind |
| Kahjustused | | | |
| Lahendused | | | |
| | | | |
| | | | |
| | | | |
| Hoone | | Ehitusaasta | Tarind |
| Kahjustused | | | |
| Lahendused | | | |
| | | | |
| | | | |
| | | | |
| Hoone | | Ehitusaasta | Tarind |
| Kahjustused | | | |
| Lahendused | | | |
| | | | |
| | | | |
| | | | |
| Hoone | | Ehitusaasta | Tarind |
| Kahjustused | | | |
| Lahendused | | | |
| | | | |
| | | | |
| | | | |

Eluhoone joonis ja kahjustuste asukohad pöördel

(Autor: Rasmus Kask, MAK)

Lisa 5 - Nõustamissoovide ankeet

Sisesta nõustamissoov

Kirjelda enda probleemi siin.

Objekti katastritunnus

* Juhendid katastritunnuse leidmiseks leiad [siit](#)

e-posti aadress

telefoni number

Kinnitan, et olen lugenud, saan aru ja nõustun [privaatsuspoliitikaga](#). *

Kinnitan, et olen lugenud, saan aru ja nõustun [kasutustingimustega](#). *

I'm not a robot



Valin Nõustaja

Saadan nõustamissoovi

Sisesta nõustamissoov

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Et netus et malesuada fames ac. At consectetur lorem donec massa sapien faucibus et molestie. Adipiscing bibendum est ultricies integer. Suspendisse potenti nullam ac tortor vitae. Ut tellus elementum sagittis vitae et. Rhoncus dolor purus non enim praesent elementum. Eget sit amet tellus cras adipiscing. In hendrerit gravida rutrum quisque. Blandit massa enim nec dui. Ornare arcu dui vivamus arcu felis. Nibh ipsum consequat nisl vel pretium. Senectus et netus et malesuada fames ac. Risus

89901:001:0138

* Juhendid katastritunnuse leidmiseks leiad [siit](#)

maamaja@omanik.ee

12345678

Kinnitan, et olen lugenud, saan aru ja nõustun [privaatsuspoliitikaga](#). *

Kinnitan, et olen lugenud, saan aru ja nõustun [kasutustingimustega](#). *

I'm not a robot



Harjumaa

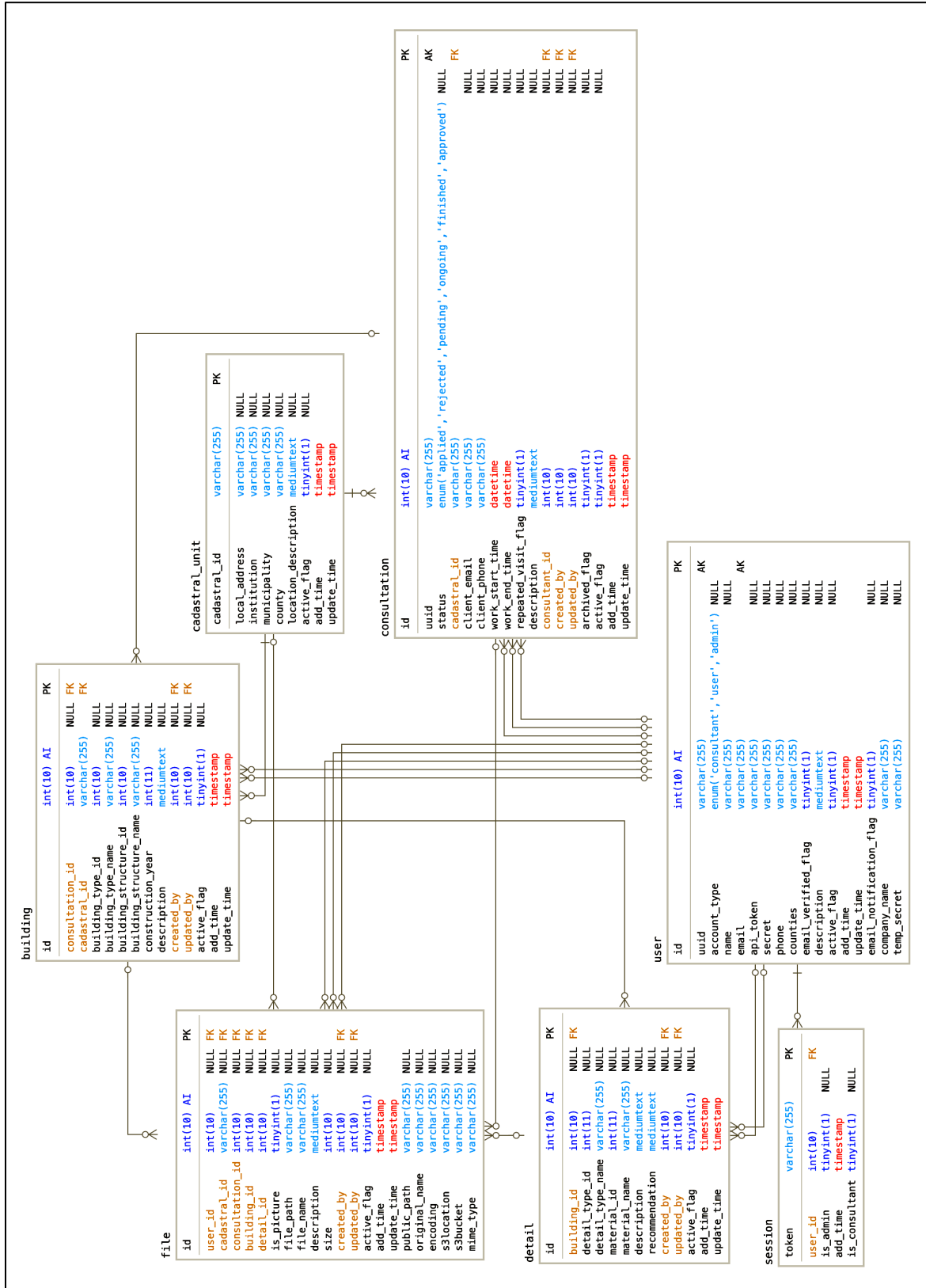
Vanaajamaja MTÜ

* Täpsemat informatsiooni teenusepakujate kohta leiad [siit](#)

Täname! Teie nõustamissoov on saadetud.

Saadan nõustamissoovi

Lisa 6 - Andmebaasi mudel














































Lisa 7 - Kasutajaliidese tühi olekupu

```
{
  auth: {
    authRequest: {
      loading: false,
      error: false
    },
    session: {
      token: false
    }
  },
  consultations: {
    consultationRequestState: {
      loading: false,
      error: false
    },
    consultationsRequestState: {
      loading: false,
      error: false
    },
    consultations: []
  },
  users: {
    consultantsRequestState: {
      loading: false,
      error: false
    },
    consultants: [],
    consultantRequestState: {
      loading: false,
      error: false
    },
    usersRequestState: {
      loading: false,
      error: false
    },
    users: [],
    userRequestState: {
      loading: false,
      error: false
    },
    userSelf: {}
  }
}
```

Lisa 8 - Redux-saga koodinäidis

```
195 function* addConsultantToConsultation({
196   payload: { consultationId, consultantId, sendConfirmation },
197 }) {
198   try {
199     const data = yield call(sessionApi, {
200       path: `/client/consultations/${consultationId}/consultant/${consultantId}${
201         sendConfirmation ? '?send_confirmation=true' : ''
202       }`,
203       method: 'POST',
204     });
205
206     yield put(addConsultantToConsultationActions.success(data));
207   } catch (error) {
208     yield put(addConsultantToConsultationActions.error(error));
209   }
210 }
211
212 export default [
213   takeLatest(addConsultantToConsultationTypes.request, addConsultantToConsultation),
```

Lisa 9 - Koodi kataloogipuu struktuur

- ▶  alters
- ▶  config
- ▶  dev_tools
- ▶  files
- ▶  logs
- ▶  public
- ▶  scripts
- ▼  server
 - ▶  common
 - ▶  config
 - ▶  controllers
 - ▶  lib
 - ▶  middleware
 - ▶  queries
 - ▶  routes
 - ▶  .babelrc
 - ▶  .eslintrc.json
 - ▶  database.js
 - ▶  index.js
 - ▶  routing.js
 - ▶  server.js
- ▼  src
 - ▶  __snapshots__
 - ▶  components
 - ▼  modules
 - ▶  Auth
 - ▶  Consultations
 - ▶  HelpPages
 - ▶  Images
 - ▶  Users
 - ▶  utils
 - ▶  .babelrc
 - ▶  .eslintrc.json
 - ▶  actionTypes.js
 - ▶  App.jsx
 - ▶  App.test.jsx
 - ▶  configureStore.js
 - ▶  createRootReducer.js
 - ▶  index.css
 - ▶  index.jsx
 - ▶  rootSaga.js
 - ▶  setupTests.js
- ▶  test