

TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Elen Liivapuu
Veebisaidi loomine firmale Darkside-Lab OÜ
Bakalaureusetöö (9 EAP)

Juhendaja: Lidia Feklistova, PhD

Tartu 2025

Veebisaidi loomine firmale Darkside-Lab OÜ

Lühikokkuvõte:

Bakalaureusetöö eesmärk on luua kaasaegne ja kasutajasõbralik veebisait ettevõttele Darkside-Lab OÜ, mis tegeleb sõidukite detailinguga. Valminud veebisait sisaldab broneerimissüsteemi, pildigalerii, teenuste ja hinnakirja lehti ning kontaktvormi. Veebisaidi arendamisel kasutati tehnoloogiaid nagu Vue.js, Express.js ja MongoDB. Loodud veebisait vastab kliendi funktsionaalsete ja mittefunktsionaalsete nõuetele, pakkudes kiiret jõudlust ja reageerivat disaini.

Võtmesõnad:

Veebisait, veebirakendus, eessüsteem, tagasüsteem, Vue.js, Express.js, MongoDB, Vercel

CERCS: P175 Informaatika, süsteemiteooria

Building a Custom Website for Darkside-Lab

Abstract:

The objective of this bachelor's thesis is to develop a modern user-friendly website for Darkside-Lab OÜ, a company that specializes in vehicle detailing. The created website includes booking system, gallery, services and prices page, and a contact form. For the development of this website, technologies such as Vue.js, Express.js, and MongoDB were used. The result meets the client's functional and non-functional requirements, offering fast performance and responsive design.

Keywords:

Website, web application, front-end, back-end, Vue.js, Express.js, MongoDB, Vercel

CERCS: P175 Informatics, systems theory

Sisukord

| | |
|---|-----------|
| Sissejuhatus | 4 |
| Mõisted ja terminid | 5 |
| 1. Sarnaste veebisaitide analüüs | 6 |
| 2. Veebidisaini põhiprintsiibid | 7 |
| 2.1 Kasutajakogemus ja kasutajaliidese disain | 7 |
| 2.2 Järjepidevus ja standardid | 7 |
| 2.3 Reageeriv veebidisain | 7 |
| 2.4 Kasutajakeskne disain | 8 |
| 3. Nõuete analüüs | 9 |
| 3.1 Funktsionaalsed nõuded | 9 |
| 3.2 Mittefunktsionaalsed nõuded | 9 |
| 3.3 Prototüüp | 10 |
| 4. Kasutatud veebitehnoloogiad | 11 |
| 4.1 Eessüsteemi tehnoloogiad | 11 |
| 4.2 Tagasüsteemi tehnoloogiad | 12 |
| 5. Valminud veebisait | 14 |
| 5.1 Veebisaidi päis | 14 |
| 5.2 Veebisaidi jalus | 14 |
| 5.3 Veebisaidi vahelehed | 14 |
| 5.3.1 Avaleht | 14 |
| 5.3.2 Galerii | 15 |
| 5.3.3 Hinnakiri | 16 |
| 5.3.4 Broneeri | 16 |
| 5.3.5 Kontakt | 17 |
| 5.4 Administraatori vaade | 18 |
| 5.5 Tagasüsteem | 19 |
| 5.5.1 API lõpp-punktid | 19 |
| 6. Lahenduse analüüs | 20 |
| 6.1 Vastavus funktsionaalsetele nõuetele | 20 |
| 6.2 Vastavus mittefunktsionaalsetele nõuetele | 20 |
| 6.3 Veebisaidi jõudlus | 20 |
| 7. Võimalikud edasiarendused | 22 |
| Kokkuvõte | 23 |
| Kasutatud kirjandus | 24 |
| Lisa 1. Esmane avalehe prototüüp | 25 |

Sissejuhatus

Tänapäeval on veebisait oluline ettevõtte turundustööriist, mis aitab laiendada kliendibaasi, suurendada nähtavust veebis ning tuua potentsiaalseid kliente. Kvaliteetselt kujundatud veebisait toimib kui ettevõtte “digitaalne visiitkaart” (ingl *digital storefront*), luues esmamulje ettevõtte kohta ja tõstes usaldusväärtsust [1]. Veebisait võimaldab kasutajal mugavalt tutvuda ettevõtte teenuste, hinnakirja, varasemate töödega ning annab võimaluse broneerida aeg teenuseks.

Autode detailinguga tegelev ettevõtte Darkside-Lab OÜ (edaspidi klient või ettevõtte) omab ainult Facebook-i konto, mille kaudu jõutakse peamiselt noorema sihtrühmani. Arvestades tänapäevase trendi, klient vajab veebisaiti. See võimaldab tal laiendada oma kliendibaasi, kasutada Google-i reklaame ja otsingumootoritele optimeerimist (ingl *Search Engine Optimization*, SEO) ning suunata potentsiaalseid kliente teenuste broneerimisele veebisaidil.

Töö eesmärk on luua kaasaegne ja kasutajasõbralik veebisait Darkside-Labile, mis hõlbustaks pakutavate teenustega tutvumist ja broneeringute tegemist. Veebisait peab olema visuaalselt atraktiivne, kaasaarav ja intuiitiivselt navigeeritav.

Bakalaureusetöö koosneb kuuest peatükist, mis annavad ülevaate veebisaidi arendamise protsessist. Esmalt tutvustatakse mõisteid ja termineid, mis on vajalikud bakalaureusetöö sujuvamaks mõistmiseks. Esimeses sisulises peatükis analüüsitakse detailingu valdkonna konkurentide veebisaite, tuues esile nende tugevused ja nõrkused, millest lähtuti Darkside-Labi veebisaidi arendamisel. Teises peatükis käsitletakse veebidisaini põhiprintsiipe, sealhulgas kasutajaliidese ja kasutajakogemuse olulisust. Kirjeldatakse järjepidevuse ja standardite järgimise tähtsust, tutvustatakse reageeriva disaini põhimõtteid ning kasutajakeskse disaini rakendamist. Kolmandas peatükis viiakse läbi funktsionaalsete ja mittefunktsionaalsete nõuete analüüs, mis määratleb veebisaidi tööpõhimõtteid. Neljandas peatükis tutvustatakse veebitehnoloogiad, mis olid kasutatud veebisaidi loomisel. Viiendas peatükis kirjeldatakse valminud veebisaidi struktuuri, alamlehti, funktsioone ja disaini. Kuuendas peatükis analüüsitakse valminud lahendust ning hinnatakse selle vastavus seatud nõuetele. Lõputöö lisades on esitatud esmase prototüübi kujundus, vahelehtede disain ja muud materjalid.

Kogu bakalaureusetöö raames kasutati tehisintellekti (keelemudelit ChatGPT 4o¹) mõistete tõlkimisel ning grammatilise ja keelelise struktuuri parandamisel.

¹ <https://openai.com/chatgpt/>

Mõisted ja terminid

Detailing on sõidukite põhjalik hooldus ja professionaalne autopesu, mille eesmärk on säilitada ja taastada sõiduki välimus.

Veebisait (ingl *website*) on ühise veebiaadressi algusosaga veebilehtede kogum, mille eesmärk on pakkuda teavet.

Kasutajaliides (ingl *User Interface*, UI) on rakenduse visuaalne osa, mille kaudu kasutaja suhtleb süsteemiga.

Kasutajakogemus (ingl *User Experience*, UX) on kogemus ja rahulolu veebisaidi kasutamisel.

Eessüsteem (ingl *front-end*) on veebisaidi nähtav ja kasutatav osa.

Tagasüsteem (ingl *back-end*) on serveripoolne osa, mis haldab andmeid ja rakenduse loogikat.

Meediapäringud (ingl *media queries*) on CSS-i reegel, mis kohandab veebilehe stiili vastavalt seadme ekraani suurusele.

Kasutajakeskne disain (ingl *User-centered design*) on disainiprotsess, mis keskendub kasutajate vajadustele ja ootustele.

Käituskeskkond (ingl *Runtime environment*) on tarkvara keskkond, kus programm töötab.

SEO (*Search Engine Optimization*) on optimeerimise protsess, mille eesmärk on suurendada veebisaidi nähtavust otsingumootorite tulemustes. See hõlmab optimaalsete märksõnade kasutamist, metakirjelduste ja sisemiste linkide kasutamist.

1. Sarnaste veebisaitide analüüs

Käesolevas peatükis analüüsitakse autode detailinguga tegelevate ettevõtete veebisaite, et tuvastada selles valdkonnas olevad veebidisaini lahendused ja praktikad. Analüüsi peamiseks eesmärgiks on mõista, milliseid disainielemente ja funktsionaalsusi konkurendid kasutavad ning milliseid puudujääke esineb, et neid arvesse võtta Darkside-Labi veebisaidi arendamisel. Analüüsi jaoks valis bakalaureusetöö autor kolm veebisaiti.

PRO Detailing² veebisait paistab silma tugeva visuaalse kontrasti poolest - tume taust, valge tekst ning erksinised detailid. Need disainielemendid loovad professionaalse ja meeldejääva mulje. Lisaks sellele on saidil loodud e-pood, mis võimaldab kasutajal tutvuda ettevõtte toodete valikuga ja sooritada ostu otse veebilehe kaudu. E-pood muudab veebisaidi mitmekülgsemaks ja suurendab kasutaja huvi ettevõtte toodete ja teenuste vastu. Veebisaidi üheks suureks nõrkuseks on see, et saidil esineb probleeme piltide ja videote laadimisega Safari brauseris, mis võib koheselt negatiivselt mõjutada kasutaja arvamust ettevõttest. Selle probleemi lahendamine võib luua usaldusväärsema esmase mulje ja tagada ühtlase kasutajakogemuse erinevatel platvormidel. Teine oluline nõrkus on seotud e-poe kasutamisega nutitelefonides. Väiksematel ekraanidel e-poe sisu on halvasti kohandatud.

Tartu Car Detailing³ veebisaidi tugevusteks on visuaalne selgus ja hea kontrastsus. See muudab teksti kergesti loetavaks ja parandab ligipääsetavust. Kasutajal on lehel navigeerimine intuiitselt arusaadav. Veebisaidi dünaamilised elemendid ja animatsioonid teevad saidi huvitavamaks, kaasahaaravamaks ning parandavad kasutajakogemust. Veebisaidil leiduvad mõned nõrkused. Mõnes kohas saidil on liiga palju teksti korruga, mis võib üle koormata kasutajat ja teeb raskemaks vajaliku informatsiooni leidmist. Teksti paremini struktureerimine võib aidata selle probleemi lahendamist.

JN Car Detailing⁴ veebisait on minimalistliku disainiga, mis tagab meeldiva ja lihtsa kasutajakogemuse. Veebisaidil kasutatavad illustratsioonid toetavad professionaalset üldmuljet ja kasutajakogemust. Visuaalne tasakaal tekstis ja illustratsioonides annab professionaalse üldmulje. Veebisaidi nõrkusteks võib nimetada järgnevaid aspekte: hinnakirja vahelehel on tekst raskesti loetav, sest teksti on palju ja kasutatud font on liiga väike. Suurema fondi kasutamine parandaks loetavust ja aitaks kasutajal kiiremini vajaliku informatsiooni leida.

Läbiviidud analüüs andis ülevaadet valdkonna trendidest visuaalselt meeldiva ja kasutajasõbraliku veebisaidi loomisel. Ülevaade sarnastest veebisaitidest andis mitmeid ideid, kuidas arendada veebilehte, mis teeks kasutajakogemuse meeldivamaks ja sujuvamaks.

Kokkuvõtteks võib öelda, et autode detailinguga seotud ettevõtete veebisaitidel on oluline roll nii visuaalse mulje kui ka meeldejääva kasutajakogemuse kujundamisel. Valitud veebisaitide tugevuste ja nõrkuste tähelepanemine ja kaardistamine võimaldab Darkside-Labi veebisaidi loomisel arvestada heade lahendustega konkurentide poolt ning vältida konkurentide nõrkusi. Järgmises peatükis uuritakse veebidisaini põhiprintsiipe, mis toetavad nende eesmärkide saavutamist.

² <https://detailing.ee>

³ <https://tartudetailing.ee>

⁴ <https://detailer.ee>

2. Veebidisaini põhiprintsiibid

Veebidisain on oluline osa iga veebisaidi arendusprotsessist, mis otseselt mõjutab kasutajakogemust ja veebisaidi kasutusmugavust. Hoolikalt läbimõeldud veebidisain keskendub kasutaja vajadustele ja ootustele, tagades veebisaidi intuiitiivsuse, ligipääsetavuse ja visuaalselt meeldiva ning meelde jääva kogemuse [2, 3]. Selles peatükis kirjeldatakse veebidisaini põhiprintsiipe, mis on rakendatud Darkside-Labi veebisaidi loomisel, lähtudes kasutajakogemusest, disaini järjepidevusest, reageerivast disainist ning kasutajakesksest lähenemisest.

2.1 Kasutajakogemus ja kasutajaliidese disain

Kasutajakogemus (ingl *user experience*, UX) ja kasutajaliidese disain (ingl *user interface*, UI) on veebidisaini kaks olulist osa, mis täidavad erinevaid, kuid omavahel tihedalt seotud funktsioone.

Kasutajakogemus keskendub sellele, kuidas kasutaja liigub veebisaidil, milline on tema teekond ja kuidas eesmärgini jõutakse. Kasutajakogemuse eesmärk on luua veebisait, mis oleks loogiline, lihtne, tõhus ja kasutajasõbralik, pakkudes samal ajal sujuvat navigeerimist eesmärgi saamiseks. [3]

Kasutajaliidese disain omakorda puudutab visuaalseid elemente, sealhulgas nuppe, värve, tüpograafiat ja paigutust. Kasutajaliidese disaini eesmärgiks on tagada visuaalne atraktiivsus ja intuiitiivne liides, kus kasutaja ei pea liiga palju mõtlema navigeerimise peale. [2]

2.2 Järjepidevus ja standardid

Kasutatavuse ekspert Jakob Nielsen kirjeldab oma töös 10 üldist kasutatavuse printsiipi, mis on olulised kasutajasõbralike digitaalsete liideste loomisel [4]. Üks neist printsiibidest on järjepidevus ja standardid (ingl *consistency and standards*). Selle printsiibi aluseks on *Jakob's Law* [5], mille kohaselt veedavad kasutajad suurema osa oma ajast, kasutades teisi veebirakendusi ja veebisaitide, mitte sinu oma [4]. See tähendab, et kasutajad on harjunud teiste digitoote disainimustritega ning ootavad samu lahendusi ka sinu tootelt. Kasutaja eeldab, et liidese komponendid töötaksid sarnaselt tuttavatele veebisaitidele. Intuiitiivne navigeerimine ja tuttavad disainimustrid võimaldavad kasutajatel veebilehel kiiremini ja mugavamalt soovitud eesmärgini jõuda, mitte õppida kuidas tundmatu veebisait või rakendus toimib.

2.3 Reageeriv veebidisain

Reageeriv veebidisain (ingl *Responsive Web Design*, RWD) on meetod, mis tagab, et veebisait on võimeline kohanduda dünaamiliselt kasutaja erinevatele seadmetele ja ekraani suurustele [6]. Reageeriva veebidisaini eesmärk on pakkuda kasutajale ühtlast kogemust lehe kasutamisel sõltumata kasutatava seadme ekraani suurusest.

Reageeriv disain tugineb kolmele peamisele komponendile [6]:

1. paindlik ruudustik (ingl *fluid grid*) - disain kohandub proportsionaalselt, mitte kindlatele mõõtmetele;
2. paindlikud pildid (ingl *flexible images*) - pildid muudavad oma suurust vastavalt ekraani mõõtmetele, mis väldib liigset venitamist või lõikamist;
3. meediapäringud (ingl *media queries*) - CSS-i reegel, mis võimaldab veebilehel disaini kohandada erinevate seadmete suuruste ja erinevate ekraanide järgi.

Arvestades neid printsiipe saab luua veebisaidi, mis on kasutajasõbralik erinevate seadmete kasutajatele.

2.4 Kasutajakeskne disain

Kasutajakeskne disain (ingl *User-Centered Design*, UCD) on kasutajakogemuse disaini oluline osa. Kasutajakeskse disaini eesmärk on luua kasutaja ootuste ja käitumismustrite põhjal ning pakkuda intuitiivselt arusaadavaid ja praktilisi lahendusi [7].

Kasutajakeskne disain aitab vähendada vigade tekkimist, samal ajal see suurendab kasutajate rahulolu. Kasutajasõbralik disain tagab, et veateated on selged, kasulikud, aitavad vähendada segadust ning süsteem annab kasutajale piisavalt tagasisidet, et suunata kasutajat ja juhendab teda järgmiste sammude tegemisel. Lisaks soovitatakse kasutaja kaasamine juba arendusprotsessi algusest ning prototüübide testimisel. Oluline on saada tagasisidet kasutaja käest, et olla kindel kas veebisait töötab intuitiivselt. [7]

3. Nõuete analüüs

Käesolevas peatükis käsitletakse Darkside-Labi veebisaidile esitatud nõudeid, tuginedes kliendi soovidele ja vajadustele. Nõuete analüüs on arendusprotsessi oluline etapp, kuna selle käigus määratletakse veebisaidi funktsionaalsus ja kasutajakogemus [8]. Nõudeid võib jaotada kaheks [8, 9]:

- funktsionaalsed nõuded (ingl *functional requirements*) määratlevad, milline funktsionaalsus peab saidil olema, et saavutada soovitud tulemusi;
- mittefunktsionaalsed nõuded (ingl *non-functional requirements*) kirjeldavad, kuidas need konkreetsed funktsioonid töötavad ja kuidas neid rakendatakse, samal ajal keskendudes kasutusmugavusele, jõudlusele, ligipääsetavusele ja turvalisusele.

3.1 Funktsionaalsed nõuded

Funktsionaalsete nõuete eesmärgiks on näidata kasutajale pakutavaid teenuseid ja anda võimalust broneerida aega nende teenuste teostamiseks. Veebisait peab võimaldama kasutajatel kiiresti tutvuda teenustega ning intuiitiivselt ja mugavalt sooritada broneeringut.

Kokku veebisaidil peab olema viis vahelehte. Kliendi poolsed funktsionaalsed nõuded nendele on järgmised:

- Avaleht - veebisaidi avalehel peavad olema dünaamilised informatiivsed elemendid ja lisaks suur kontrastne nupp, mis viib broneerimislehele;
- galerii - veebisaidil peab olema pildigalerii, mis näitab ettevõtte varasemalt tehtud töid;
- hinnakiri - veebisait peab sisaldama ülevaadet teenuste hindadest. Kliendi soovil peab see olema toodud pildina. Sellel lehel peavad olema kiirringid broneeringu ja kontaktalamlehtedele;
- broneerimissüsteem - veebisait peab pakkuma veebisaidi külastajale võimalust broneerida teenuseid otse veebisaidi kaudu valides kuupäeva, kellaaega ja sisestades oma kontaktandmeid;
- kontaktinfo ja sotsiaalmeedia - kasutaja peab saama võtta ühendust otse lehe kaudu või saada rohkem informatsiooni ettevõtte sotsiaalmeediast. Selleks on lisatud kontaktvorm ja sotsiaalmeedia link, mis võimaldavad kiiret ligipääsu;
- dünaamilised elemendid - veebisait peab sisaldama dünaamilisi elemente (nt karusellid ja animatsioonid), mis muudavad lehe huvitavamaks ja pilkupüüdvamaks;
- iga vahelehe jalus peab sisaldama firma hüüdlause ja kontaktandmeid;
- igal vahelehe päises on navigeerimismenüü;
- igal vahelehel paremas ülemises nurgas peab olema klikitav logo, mis viib Avalehele;
- administraatori vaates peab olema võimalus vaadata ja kustutada broneeringuid.

3.2 Mittefunktsionaalsed nõuded

Mittefunktsionaalsed nõuded määratlevad, kuidas peavad funktsioonid töötama ja millised omadused on veebisaidil olulised. Nende eesmärgiks on tagada, et veebisait oleks kiire, ligipääsetav ja vastaks kasutaja ootustele.

Kliendiga arutledes jõuti järeldusele, et peamisteks mittefunktsionaalseteks nõueteks on:

- Disain - lehe disain annab edasi ettevõtte visuaalset mainet. Disain peab olema järjepidev ja professionaalne, et jätta usaldusväärse mulje. Veebisaidil peab olema kasutatud tume taust valge tekstiga ja erkkollased detailid;

- jõudlus - veebisait peab laadima vähem kui 2 sekundiga. Selleks tuleb optimeerida pildid, et vähendada laadimisaega ning parandada jõudlust;
- mobiilisõbralikkus, reageeriv disain - veebisait peab nägema olema reageeriv, mis tähendab, et see peab korrektselt ja ühtlaselt edastama informatsiooni erinevate ekraanisuurustega seadmetes;
- SEO (otsingumootoritele optimeerimine) - märksõnade lisamine ja metakirjeldused ning struktureeritud andmete kasutamine parandavad veebisaidi leitavust otsingumootorites;
- ligipääsetavus - veebisait peab järgima WCAG 2.1 AA standardit [10]. Tuleb arvestada elementide kontrastsust ja klaviatuuriga navigeerimise võimalust, mis on olulised aspektid juurdepääsu tagamisel erinevate vajadustega kasutajate jaoks;
- hallatavus - kliendil peab olema võimalus hallata veebisaiti iseseisvalt pärast valmimist.

3.3 Prototüüp

Prototüübi loomine on oluline samm veebidisaini arenduses, sest see annab arendajatele ja disaineritele võimaluse katsetada erinevaid lahendusi ning saada esmase tagasiside kliendilt ja läbi selle täpsustada võimalikke kitsaskohti [11]. Prototüüp annab võimaluse visualiseerida, kuidas hakkab veebisait välja nägema kuidas hakkab navigeerimine toimima.

Veebisaidi avalehe visuaalse lahenduse prototüüp oli loodud, et paremini mõista kliendi vajadusi ja ootusi. Prototüübi loomiseks kasutas bakalaureusetöö autor Figma⁵ veebikeskkonda. See on laialdaselt levinud ja populaarne tööriist disaini ja prototüüpide arendamisel. Üheks väga oluliseks eeliseks on võimalus töötada erinevatel platvormidel ja erinevates brauserites. Lisaks, Figma töötab brauseripõhiselt ja salvestab andmeid pilves. Tänu sellele saavad klient ja arendaja ligi prototüübile, mis on kuvatud täpselt samasugusena kõikidel seadmetel. Figma pakub intuitiivset kasutajaliidest, tänu millele saab kujundada komponentide paigutust.

Loodud avalehe prototüüp võimaldas saada kliendi ootustest paremini aru ja aitas saada esmane ülevaade lehe struktuurist ja võimaliku funktsionaalsusest. Loodud prototüüp (lisa 1) esitati kliendile esialgse tagasiside saamiseks. Kliendiga arutledes muutus prototüüp märgatavalt palju ning valminud veebisait erineb esialgsest prototüübist.

⁵<https://www.figma.com/>

4. Kasutatud veebitehnoloogiad

Selles peatükis analüüsitakse ja põhjendatakse töös kasutatud tehnoloogiad, mida bakalaureusetöö autor valis veebisaidi arendamiseks. Valitud veebitehnoloogiad võimaldasid luua lahenduse, mis tagas veebisaidi funktsionaalsuse, töökiiruse ja kasutajasõbralikkuse. Kliendi poolt ei seatud konkreetseid piiranguid kasutatavate tehnoloogiate osas, mis võimaldas valida kaasaegseid lahendusi.

4.1 Eessüsteemi tehnoloogiad

Eessüsteemi (ingl *front-end*) tehnoloogiate valikul lähtuti kasutajakogemuse ja kasutajaliidese tulevasesest kvaliteedist. Veebisaidi aluseks on **HTML ja CSS**, mis on veebiarenduse alustalad ning võimaldavad luua veebilehe struktuuri ja kujundust. HTML ja CSS annavad võimaluse luua veebilehe peamist struktuuri ja määratleda, kuidas veebileht hakkab visuaalselt välja nägema.

Keerukama interaktiivsuse loomiseks võib kasutada erinevaid JavaScriptil põhinevaid raamistikke. Tänapäeval on laialdaselt kasutatavad raamestikud on näiteks Vue.js⁶, React⁷ ja Angular⁸. React keskendub komponentidele ja pakub suurt paindlikkust, kuid vajab tihti lisateeke. Angular sobib hästi suuremate projektide jaoks, see on usaldusväärne, aga võib osutuda liiga mahukaks. Vue.js on paindlik JavaScripti raamistik, mis on loodud efektiivseks kasutajaliidese arendamiseks. Vue.js on populaarne arendajate seas tänu selle lihtsale süntaksile, paindlikkusele ja madalale õppimiskõverale [12]. Vue.js võimaldab jagada veebilehte komponentideks, mis teeb arenduse modulaarseks. Vue.js on järk-järguliselt kasutuselevõetav raamistik - arendust saab alustada lihtsamatest komponentidest, lisades hiljem keerukamat funktsionaalsust [13]. Just nendel põhjustel valis bakalaureusetöö autor interaktiivsuse loomiseks **Vue.js** raamistiku.

Käesolevas töös on kasutatud Vue.js-i broneerimisüsteemi loomiseks, hallates kasutajaliidese loogikat ja ühendust tagasüsteemiga. Kuigi lisafunktsionaalsust saab lisada ka ainult JavaScripti abiga, Vue.js pakub mitmeid kasutajasõbralikke lahendusi, mis teevad arendusprotsessi lihtsamaks ja säästavad aega. Vue.js on jätkusuutlik ja laiendatav raamistik, mis vajadusel võimaldab lisada uusi funktsioone. Vue.js-i komponentidel põhinev projekti ülesehitus võimaldas luua korduvkasutatavaid elemente, nagu päis, navigeerimisriba ja jalus, mis tagavad veebisaidi järjepidevuse ja teevad edasist arendust kiiremaks ja lihtsamaks.

Üheks oluliseks valmiva veebisaidi osaks peab olema võimalus broneerida aeg teenusteks läbi broneerimissüsteemi. Selleks valiti broneerimissüsteemi arendamiseks **Vue Simple Calendar**⁹ komponenti, mis on mõeldud kalendri lihtsaks integreerimiseks projekti. Selle komponendi abil saab kasutaja mugavalt valida sobiva kuupäeva ja kellaaja broneerimiseks. Lisaks juba hõivatud kellaajad, ei ole nähtavad, mis muudab broneerimist lihtsamaks ja kiiremaks. Vue Simple Calendar võimaldab kohandada kalendri funktsionaalsust vastavalt projekti vajadustele. [14]

Teek **AOS** (Animate on Scroll)¹⁰ võimaldab anda lehele dünaamilisust ja visuaalset atraktiivsust. Tänu AOS-le saab rakendada erinevaid animatsioone ja üleminekuid, mis aktiveerivad, kui kasutaja kerib lehte. [15]. Animatsioonid ja üleminekud aitavad parandada kasutajakogemust, muutes veebisaidi elavamaks ja professionaalsemaks.

⁶ <https://vuejs.org/>

⁷ <https://react.dev/>

⁸ <https://angular.dev/>

⁹ <https://github.com/richardtallent/vue-simple-calendar>

¹⁰ <https://michalsnik.github.io/aos/>

4.2 Tagasüsteemi tehnoloogiad

Tagasüsteemi (ingl *back-end*) arendamisel valis bakalaureusetöö autor tehnoloogiad, mis on paindlikud, usaldusväärsed ja laialdaselt tuntud. Nendeks osutusid Node.js (koos Express.js teegiga) ja MongoDB andmebaasi jaoks. Valitud tehnoloogiad on tuntud paindlikkuse ja kindluse poolest ning võimaldavad hallata päringuid ja andmeid.

Node.js¹¹ on mitmeplatvormiline avatud lähtekoodiga JavaScripti käituskeskkond (ingl *runtime environment*), mis võimaldab kasutada JavaScripti nii kliendi- kui ka serveripoolse arendamisel [16]. Node.js kasutab asünkroonselt sisend/väljund (ingl *Input/Output*) mudelit koos sündmuspõhise programmeerimisega, mis erineb traditsioonilisest mitmelõimelisest lähenemisest [17]. Node.js-i asünkroonsed funktsioonid võimaldavad tõhusalt hallata samaaegselt mitmeid päringuid, mis on väga oluline reaalajas broneerimissüsteemide jaoks [16][17]. Antud projektis on kasutatud Node.js-i broneerimissüsteemi tagasüsteemi loomisel, et hallata kasutajate päringuid ja andmeid MongoDB andmebaasis.

Express.js¹² on minimalistlik ja paindlik Node.js raamistik, mida kasutatakse veebirakenduste ja API-de loomiseks. Express.js on lihtsasti integreeritav teiste Node.js teekidega ning võimaldab kiiremini luua RESTful API-sid ja hallata marsruutimist. [18]

Express.js on kasutatud käesoleva veebisaidi loomisel broneerimissüsteemi arendamisel. Raamistik võimaldab kasutajal saata päringuid ja salvestada andmeid MongoDB andmebaasi. Express.js marsruutimine määrab, kuidas päringuid töödeldakse ja suunatakse. Express.js on kasutatud ka veebisaidi HTML-i sisu edastamiseks kliendile.

MongoDB¹³ on kiire ja paindlik NoSQL andmebaas, mille eripäraks on paindlik andmete salvestamine. Võrreldes relatsiooniliste (SQL) andmebaasidega MongoDB ei vaja kindlat skeemi, mis teeb andmete struktuuri muutmist lihtsamaks ja kiiremaks. [19] MongoDB kasutati broneeringute andmete salvestamiseks ja haldamiseks.

Mongoose¹⁴ on MongoDB ja Node.js rakenduste jaoks loodud skeemipõhine lahendus, mis lihtsustab andmete struktureerimist ja annab võimaluse luua selgeid andmemudeleid [20]. Veebisaidi töötamiseks kasutatav mudeli skeem on toodud joonisel 1. Mongoose on kasutusel MongoDB päringute edastamiseks ja selleks, et integreerida andmebaasi tagasüsteemi.

¹¹ <https://nodejs.org/>

¹² <https://expressjs.com/>

¹³ <https://www.mongodb.com/>

¹⁴ <https://mongoosejs.com/>

```
darkside-lab > models > JS Booking.js > ...
1  const mongoose = require('mongoose');
2
3  const bookingSchema = new mongoose.Schema({
4    startDate: { type: Date, required: true},
5    name:      { type: String, required: true },
6    email:     { type: String, required: true },
7    phone:     { type: String, required: true },
8    comment:   { type: String, required: false },
9    createdAt: { type: Date, default: Date.now }
10 });
11
12 const Booking = mongoose.model('Booking', bookingSchema);
13 module.exports = Booking;
```

Joonis 1. Andmebaasis kasutatav mudeli skeem.

Broneeringu skeem koosneb soovitud broneerimise kuupäevast ja broneerija andmetest ning *createdAt* on aeg, millal päring oli saadetud.

Vercel¹⁵ on veebimajutusteenus, mis on loodud JavaScripti raamistikel, nagu Vue.js, põhinevate veebirakenduste hostimiseks. Vercel võimaldab lihtsat ühendust GitHubi repositooriumiga, ja see automaatselt juurutab rakenduse iga kord, kui kood repositooriumis on muutunud. Vercel pakub tasuta veebimajutust väiksemate projektide jaoks ning genereerib unikaalse URL-i, mille kaudu saab veebisaidi testida ja kliendiga jagada. Selles töös oli Vercel kasutatud valminud veebisaidi majutamiseks.

¹⁵ <https://vercel.com/>

5. Valminud veebisait

Selles peatükis vaadeldakse valminud veebisaidi struktuuri ja funktsionaalsust. Veebisait on saadaval aadressil <https://darkside-lab.vercel.app/> ning on nutiseadmete sõbralik. Veebisaidi lähtekood on saadaval GitHub repositooriumis <https://github.com/elenliivapuu/darkside-lab/>.

5.1 Veebisaidi päis

Päis on veebisaidi oluline komponent, mis jääb igal vahelehel samaks, et tagada järjepidevus ning lihtsustab navigeerimist saidil. Nii saab kasutaja kiiremini vajaliku informatsiooni leida. Antud veebisaidil päis on realiseeritud korduva komponendina, mis esineb igal vahelehel.

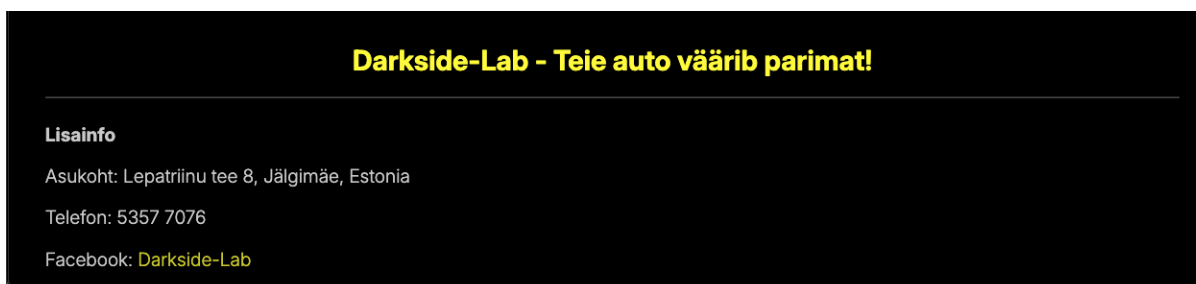
Valminud päis (Joonis 2) koosneb ettevõtte logost ja navigatsiooniribast. Logole vajutades jõuab kasutaja avalehele [2]. Navigatsiooniriba sisaldab järgmisi vahelehtede linke: “Avaleht”, “Galerii”, “Hinnakiri”, “Broneeri” ja “Kontakt”. Navigatsiooniriba elementidel kursoriga liikumisel muutub elemendi taustavärv halliks, mis annab kasutajale visuaalset tagasisidet. Aktiivne vaheleht on näidatud erkkollase taustaga, mille järgi saab kasutaja kiiresti aru, mis vahelehel ta asub. Navigatsiooniriba on kleepuv (ingl *sticky*), mis tähendab, et see on ekraanil nähtav ka siis, kui lehte keritakse allapoole.



Joonis 2. Veebisaidi päis

5.2 Veebisaidi jalus

Veebisaidi jalus on teine korduv komponent, mis esineb igal vahelehel. Valminud veebisaidi jalus (Joonis 3) sisaldab järgmisi elemente: ettevõtte iseloomustav lause “Darkside-Lab - Teie auto väärrib parimat!”, kontaktandmed (aadress ja telefoninumber), sotsiaalmeedia link. Jalus aitab kasutajal kiiresti leida vajaliku informatsiooni ettevõtte kohta.



Joonis 3. Veebisaidi jalus

5.3 Veebisaidi vahelehed

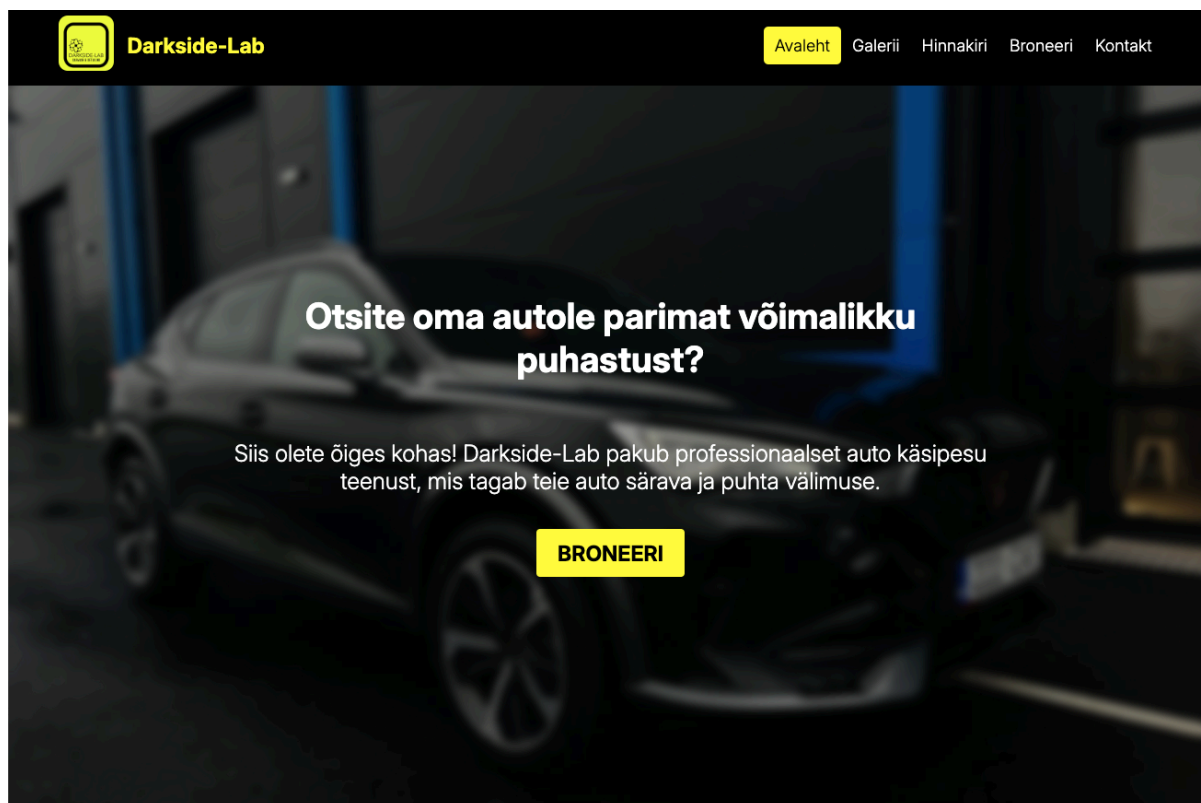
Järgnevalt vaadatakse lähemalt valminud veebisaidi vahelehti ja nende funktsionaalsust. Kokku on saidil saadaval viis vahelehte. Igal vahelehel on oma kindel eesmärk.

5.3.1 Avaleht

Veebisaidi avaleht on sageli esimene koht, kuhu kasutaja satub. Sellel on tähtis roll kasutaja tähelepanu haaramisel ja hoidmisel. Avalehe disain peab olema selge ja meeldejääv.

Avaleht koosneb järgmistest osadest: päis, hero-sektsioon (ingl *hero-section*) ja tegevusele kutsuv broneeringu nupp (ing *Call to Action*, CTA) (joonis 4). Sellele järgneb “Mis meid eristab?” sektsioon, kus on toodud välja punktid, mille poolest on Darkside-Lab erinev

teistest ettevõtetest. Kerides edasi ilmub seksioon “Miks valida auto käsipesu?”, kus on kirjeldatud millised eelised on käsipesul masinpesula ees (joonis 5).



Joonis 4. Veebisaidi Avalehe ülaosa.



Joonis 5. Veebisaidi alusosa arvutivaates koos seksiooniga “Miks valida auto käsipesu?” seksioon.

5.3.2 Galerii

Galerii vaheleht pakub veebisaidi külastajale võimalust tutvuda ettevõtte varasemalt tehtud töödega (joonis 6). Galerii vaheleht on loodud minimalistliku disainiga, et hoida fookus tehtud töödel. Pildid on paigutatud ruudustikus (ingl *grid layout*), mis tagab, et vaheleht on minimalistlik ja puhas.



Galerii



Joonis 6. Galerii vaheleht koos ettevõtte tehtud töödega

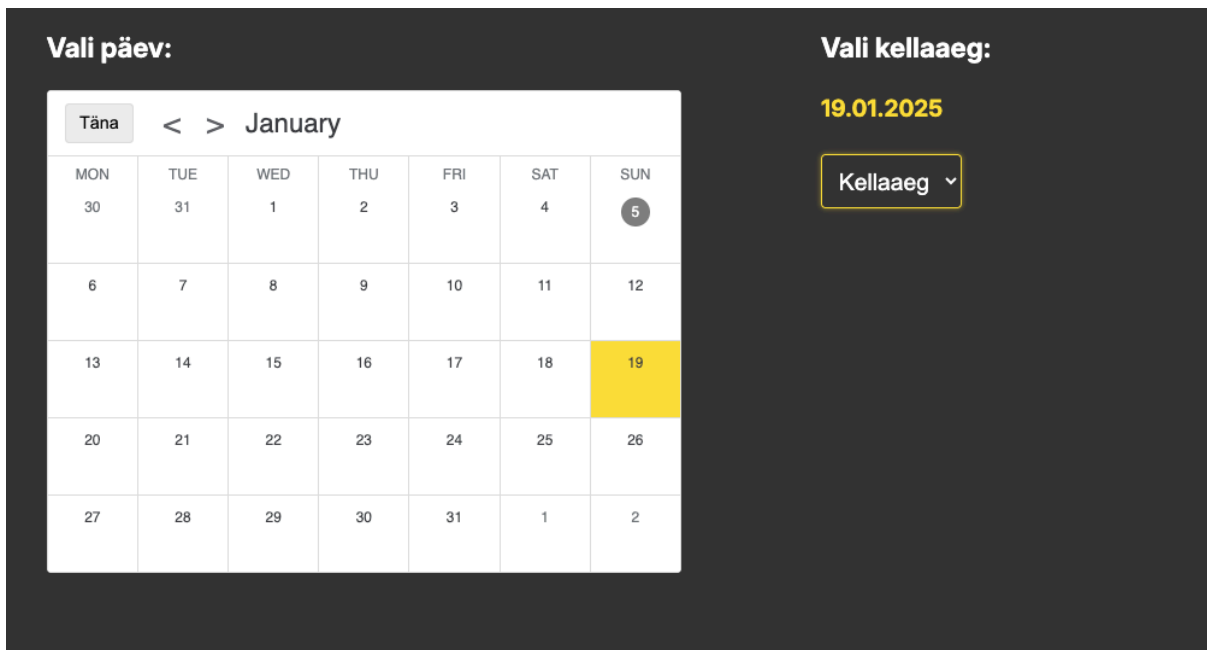
5.3.3 Hinnakiri

Hinnakirja vaheleht pakub ülevaadet ettevõtte teenustest ja nende hindadest. Vastavalt kliendi soovile koosneb hinnakiri vaheleht kahest pildist, kus kasutaja saab näha teenuseid ja nende hinda. Kaks pilti on paigutatud kõrvuti. Vahelehe allpool on tegevusele kutsuv nupp "Huvitatud? Broneeri kohe" või "Võta ühendust!". Vajutades nupul "Broneeri kohe" jõuab veebisaidi külastaja "Broneeri" vahelehele. Vajutades nupul "Võta ühendust" suunatakse kasutajat "Kontakt" vahelehele.

5.3.4 Broneeri

Broneerimise vaheleht võimaldab broneerida teenuse osutamisaega. Valminud lahendus on selline, et veebisaidi külastaja saaks kohe aru kuidas ta peab lehel navigeerima. "Broneeri" vaheleht koosneb mitmest osast.

Kalendri komponent (joonis 7) võimaldab valida sobiva päeva klikkides kalendris sellele päevale. Seejärel saab edasi valida vaba kellaaja (realiseeritud rippmenüü (ingl *dropdown menu*) abil). Kellaaja rippmenüü näitab ainult vabu aegu.



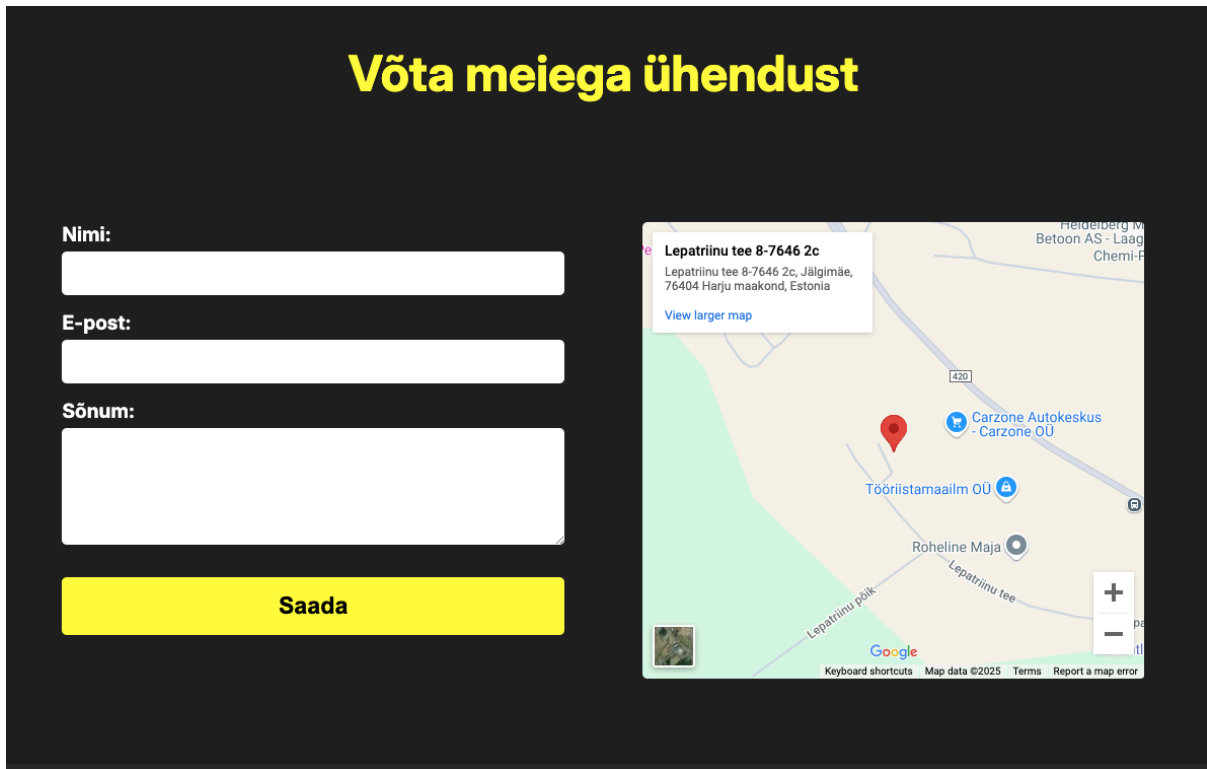
Joonis 7. Broneeri vahelehe kalendrivaade.

Sellel vahelehel olev "Sisesta andmed" plokk võimaldab veebisaidi külastajale jätta oma kontaktandmed. Nendeks on nimi, telefon ja e-posti aadress. Pärast andmete sisestamist tuleb vajutada suure kontrastsusega olevale nupule "Broneeri". Kinnituseks, et broneering on saadetud näeb kasutaja kinnitust "Broneering on edukalt salvestatud!". Kui tekib mingi viga, siis tuleb ekraanile veateade: "Broneeringu salvestamine ebaõnnestus."

Broneerimissüsteem on seotud tagasüsteemiga, mis salvestab broneeringud MongoDB andmebaasi.

5.3.5 Kontakt

Sellel vahelehel on veebisaidi külastajal võimalus ettevõttega ühendust võtta (Joonis 8). Kontakt vaheleht koosneb suurest pealkirjast "Võta meiega ühendust", kontaktvormist, kus on väljad "Nimi", "E-post" ja "Sõnum". Sõnumi saatmiseks tuleb vajutada suure kontrastsusega nupul "Saada".



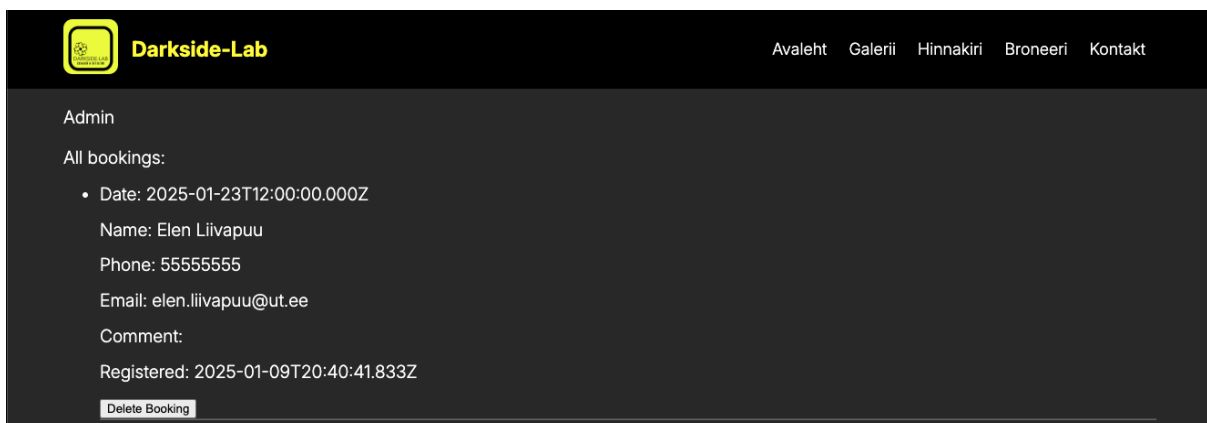
Joonis 8. Kontakt vaheleht.

Kontakt vormi kõrval asub Google Maps kaart, kus on näidatud ettevõtte asukoht kaardil.

5.4 Administraatori vaade

Administraatori vaade on loodud broneeringute haldamiseks. Administraatori vaates saab näha kõiki olemasolevaid broneeringuid ning vajadusel saab neid kustutada.

Administraatori õigustega kasutajal on olemas täielik ligipääs broneeringute andmestikule, sealhulgas kliendi kontaktandmetele. Tavalise kasutaja jaoks on nähtavad ainult broneerimiseks vabad kuupäevad ja kellaajad, ilma isikuandmeteta.



Joonis 9. Administraatori vaade

Joonisel 9 on näidatud broneeringu andmed, mis on administraatorile saadaval. Broneeringu kustutamiseks tuleb administraatoril vajutada nupule.

5.5 Tagasüsteem

Selles alampeatükis kirjeldatakse valminud veebisaidi tagasüsteemi ehitust, API lõpp-punkte ja nende funktsionaalsust. Tagasüsteemi eesmärk on hallata broneeringuid, võimaldades kasutajal teha uusi broneeringuid, vaadata olemasolevaid ja neid vajadusel kustutada. Tagasüsteem on loodud Node.js-i ja Express.js-i abil. Need tagavad kiiret ja tõhusat serveripoolset loogikat. Andmete salvestamiseks kasutatakse MongoDB koos Mongoose teegiga.

5.5.1 API lõpp-punktid

Veebisaidi tagasüsteemil on kolm peamist API lõpp-punkti, mis vastutavad broneeringute haldamise eest:

- **POST** /api/bookings - uue broneeringu lisamine. Kasutaja poolt edastatakse broneeringu andmed: (nimi, telefon, e-post, kuupäev ja kellaaeg) ja salvestatakse andmebaasi;
- **GET** /api/bookings - olemasolevate broneeringute kuvamine;
- **DELETE** /api/bookings - broneeringu kustutamine.

Lõpp-punkt GET võtab parameetrina autentimisvõtme (ingl *token*), mille abil kontrollitakse, kas kasutajal on administraatori õigused. Kui kasutaja on autenditud, siis tagastatakse kõik broneeringud koos broneerija andmetega. See on kasutuses administraatori vaates, kus on vaja kuvada kogu broneeringu informatsiooni. Kui aga kasutaja ei ole autenditud, siis tagastatakse ainult broneeritud kuupäevad ja kellaajad. Seda kasutatakse broneerimise vaates veebisaidil, kus kasutajale näidatakse ainult saadaolevad ajad. Administraatori õigused on kontrollitud ka DELETE lõpp-punkti puhul, kuna ainult administraator peab saama broneeringuid kustutada. Kui kasutaja ei ole autenditud, siis tagastatakse veeteade ja broneeringut ei kustutata. Kõik API päringute vastused tagastatakse JSON-formaadis, mis lihtsustab edasist andmete töötlemist.

Tagasüsteemi server edastab Vue.js-i poolt loodud ja kompileeritud HTML-failid vaikimisi lõpp-punkti "/" (juur, ingl *root*) kaudu. See võimaldab kasutajaliidest kuvada kohe, kui lõpp-punkti avatakse.

6. Lahenduse analüüs

Selles peatükis analüüsitakse valminud veebisaidi vastavust kliendi funktsionaalsete ja mittefunktsionaalsete nõuetele, seatud eesmärkidele ja ootustele. Töö eesmärk oli luua kasutajasõbralik ja kaasaegne veebisait ettevõttele Darkside-Lab, mis oleks atraktiivne ja kaasahaarav ning võimaldaks broneerida teenuste osutamisaega.

Veebisaidi loomisel lähtuti veebidisaini põhiprintsiipidest, mida kirjeldati peatükis 2. Eriti suurt tähelepanu pöörati kasutajakogemusele, järjepidevusele ja ligipääsetavusele.

6.1 Vastavus funktsionaalsetele nõuetele

Valminud veebisait vastab kõigile esitatud funktsionaalsetele nõuetele.

Veebisaidil on olemas vaheleht “Avaleht”, mis sisaldab infot ettevõttest ja pakutavatest teenustest. Veebisaidil sai integreeritud pildigalerii alamlehel “Galerii”, kus kasutaja saab tutvuda ettevõtte varasemalt tehtud töödega. Galerii pildid näitavad ettevõtte professionaalsust. Hinnakirja kuvamine alamlehel “Hinnakiri” annab ülevaate teenustest ja nende maksumusest. Vastavalt kliendi soovile see info on toodud piltidena. “Kontakt” vaheleht koosneb kontaktvormist, kus on väljad nime, e-posti ja sõnumi saatmiseks (vajutades nupul “Saada”). Samuti lehel on toodud ettevõtte asukoht Google Maps kaardil. Igal valminud vahelehel on sama sisuga jalus ja päis. Klakkides firma logole suunatakse veebisaidi külastajat avalehele.

Veebisaidil on loodud broneerimissüsteem, kus veebisaidi külastaja saab kalendris valida sobiva kuupäeva ja kellaaja teenuse broneerimiseks. Külastajale on nähtavad valimiseks ainult vabad kellaajad. Kui aeg on valitud, tuleb kasutajal sisesta enda andmed broneerimissüsteemi kontaktvormi ja vajutada nupule “Broneeri”.

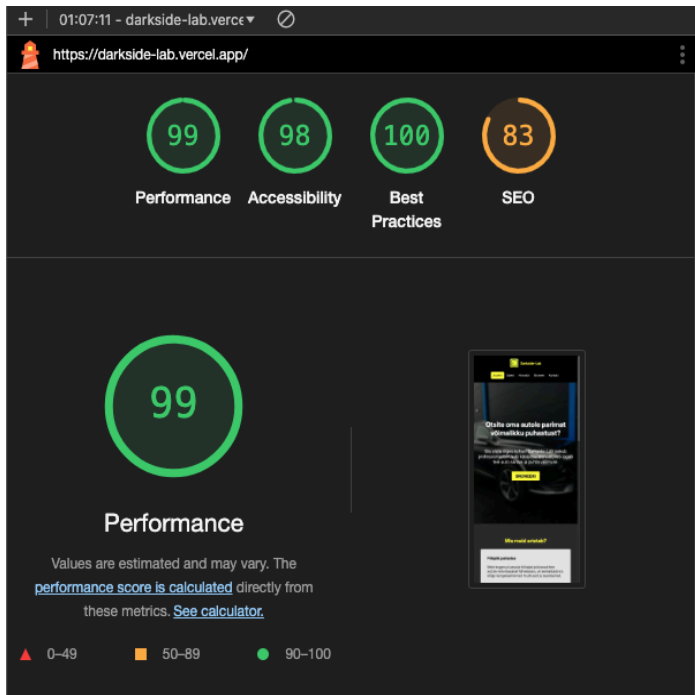
Kokkuvõtvalt võib öelda, et kõik funktsionaalsed nõuded, mis olid püstitatud projekti alguses on täidetud.

6.2 Vastavus mittefunktsionaalsetele nõuetele

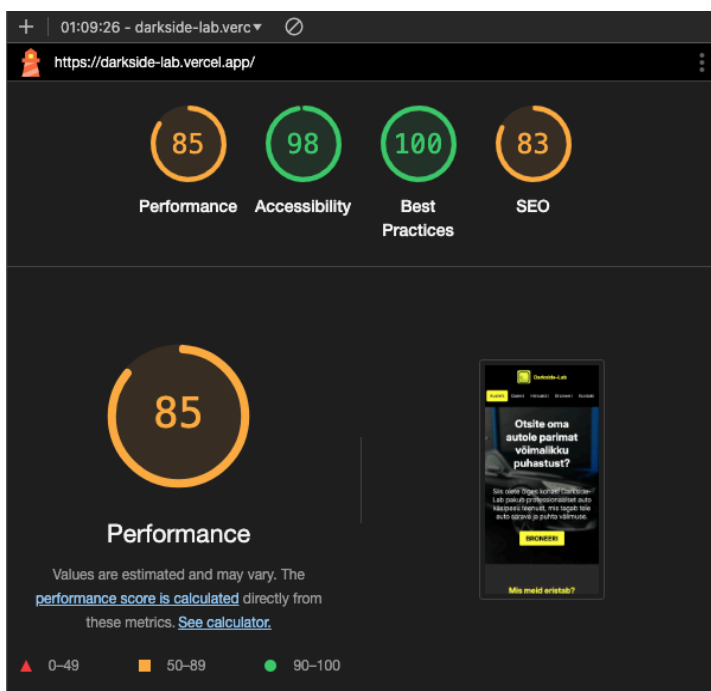
Tähelepanu pöörati ka mittefunktsionaalsetele nõuetele, mis teevad kasutajakogemuse sujuvamaks ja meeldivamaks. Sain reageeriv disain kohandab veebisaidi sisu vastavalt ekraani suurusele, tagades ühtlase funktsionaalsuse ja disaini erinevatel ekraanidel ja seadmetel. Veebisaidi kood ja metaandmed on optimeeritud otsingumootoritele, mis parandab positsiooni otsingutulemustes. Kiirus ja jõudlus on testitud Chrome Lighthouse tööriistaga, mis näitas häid tulemusi SEO osas, ligipääsetavusel ja laadimiskiirusel. Sellest räägitakse lähemalt järgmises alapeatükis.

6.3 Veebisaidi jõudlus

Valminud veebisaidi jõudluse hindamiseks kasutati Chrome Lighthouse tööriista, mis analüüsis veebisaidi kiirust, SEO-d ja ligipääsetavust (joonis 10 ja joonis 11).



Joonis 10. Google Lighthouse testi tulemus arvutivaates.



Joonis 11. Google Lighthouse testi tulemus mobiilivaates.

Kõikide testide tulemus on üle 80, mida võib pidada heaks tulemuseks. Tulemused kinnitavad, et veebisait vastab seatud kvaliteedinõuetele.

7. Võimalikud edasiarendused

Valminud veebisait vastab põhilistele kliendi nõuetele, kuid on mõned funktsionaalsused, mida võiks lisada ja edasi arendada. Üheks võimalikuks edasiarenduseks võib olla sisselogimis- ja registreerimissüsteem, mis võimaldab kasutajal luua konto ja vaadata enda broneeringute ajalugu.

Teiseks potentsiaalseks edasiarenduseks võiks olla administraatori tööriistade laiendamine. See võimaldaks hallata veebisaidi tõhusamalt, sealhulgas galerii haldamist ja broneeringute muutmist. Hetkel veebisaidi majutuseks kasutatakse Verceli, kuid selle asemel võiks kaaluda domeeni ostu.

Need edasiarendused aitaksid muuta veebisaidi kasutamiseks mugavamaks ja tõhusamaks.

Kokkuvõte

Käesoleva bakalaureusetöö raames loodi veebisait ettevõttele Darkside-Lab OÜ, mis tegeleb sõidukite detailinguga. Töö eesmärk oli luua kliendile kaasaegse, kasutajasõbralikku veebisaidi, mis vastab kliendi funktsionaalsetele ja mittefunktsionaalsetele nõuetele. Veebisait sisaldab broneerimissüsteemi, pildigaleriid, teenuste ja nende hinnakirja ning kontaktvormi. Valminud veebisait annab Darkside-Labile professionaalse ja atraktiivse digitaalse visiitkaardi, mis aitab ettevõttel suurendada kliendibaasi ja parandada teenuste kättesaadavust. Kliendi sõnul ta on rahul loodud veebisaidiga ning võtab selle kasutamisele esimesel võimalusel.

Veebisaidi arendamisel kasutati kaasaegseid tehnoloogiaid, nagu Vue.js, Express.js ja MongoDB. Erilist tähelepanu pöörati kasutajakogemusele, visuaalsele atraktiivsusele ja reageerivale disainile, et tagada veebisaidi kasutavust erinevatel seadmetel. Valminud veebisait on testitud Chrome Lighthouse tööriistaga. Testide tulemused kinnitasid veebisaidi head jõudlust ja SEO-d ning ligipääsetavust. Eesmärgid, mis olid püstitatud veebisaidile on saavutatud, kuigi edasiarenduseks võib lisada autentimissüsteem ja viia veebisait kliendi domeenile üle. Töö andis autorile väärtusliku kogemuse klientidega suhtlemises ja koostöös ning teadmisi kaasaegse veebiarenduse põhimõtetest.

Kasutatud kirjandus

- [1] Chadwick, S. Digital Marketing Essentials, 4th ed. New York: Pearson, 2021.
- [2] Krug, S. Don't Make Me Think: A Common Sense Approach to Web Usability, 3rd ed. Berkeley, CA: New Riders, 2014.
- [3] Garrett, J. J. The Elements of User Experience: User-Centered Design for the Web and Beyond, 2nd ed. Berkeley, CA: New Riders, 2010.
- [4] Nielsen, J., 10 Usability Heuristics for User Interface Design. Nielsen Norman Group, Kättesaadav: <https://www.nngroup.com/books/usability-engineering> ().
- [5] Nielsen Norman Group, "Jakob's Law of Internet User Experience." Kättesaadav: <https://www.nngroup.com/videos/jakobs-law-internet-ux/> (26.09.2024).
- [6] Marcotte, E. "Responsive Web Design," A List Apart, Kättesaadav: <https://alistapart.com/article/responsive-web-design> (15.10.2024).
- [7] Norman, D. The Design of Everyday Things, Revised and Expanded Edition. Basic Books, 2013.
- [8] Sommerville, I. Software Engineering, 9th ed. Addison-Wesley, 2011, lk. 85.
- [9] Larman, C. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, 3rd ed. Prentice Hall, 2010.
- [10] Web Content Accessibility Guidelines (WCAG) 2.1: <https://www.w3.org/TR/WCAG21/> (13.11.2024).
- [11] Brown, T., Change by Design: How Design Thinking Creates New Alternatives for Business and Society. Harper Business, 2009.
- [12] Breedis, R. JavaScripti kasutajaliidese raamistike võrdlus. https://comserv.cs.ut.ee/ati_thesis/datasheet.php?id=72077 (07.12.2025).
- [13] Vue.js. <https://vuejs.org/> (18.12.2024).
- [14] VueSimpleCalendar. <https://www.npmjs.com/package/vue-simple-calendar> (03.01.2025).
- [15] Animate on Scroll. <https://michalsnik.github.io/aos/> (03.01.2025).
- [16] Node.js. <https://nodejs.org/en/about> (05.01.2025).
- [17] Tilkov, S., & Vinoski, S. Node.js: Using JavaScript to Build High-Performance Network Programs. IEEE Internet Computing.
- [18] Express.js. <https://expressjs.com> (05.01.2025).
- [19] Chodorow, K. MongoDB: The Definitive Guide. O'Reilly Media, 2019.
- [20] Mongoose. <https://mongoosejs.com> (06.01.2025).
- [21] Vercel. <https://vercel.com/docs> (06.01.2025).

Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Elen Liivapuu,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose Veebisaidi loomine firmale Darkside-Lab OÜ, mille juhendaja on Lidia Feklistova, reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.
2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.
3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.
4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Elen Liivapuu

09.01.2025