

TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Alvar Antson

Veebipoe loomine ettevõttele Emart Auto OÜ

Bakalaureusetöö (9 EAP)

Juhendaja: Helle Hein

Tartu 2021

Veebipoe loomine ettevõttele Emart Auto OÜ

Lühikokkuvõte:

Bakalaureusetöö eesmärgiks on luua veebipood ettevõttele Emart Auto OÜ. Töö sisaldab ettevõtte tausta kirjeldust, kliendi nõuete kirjeldust, kasutatud tehnoloogiate kirjeldust, arendusprotsessi kirjeldust, projekti ja projekti andmebaasi struktuuri kirjeldust ning tehtud töö analüüsi.

Võtmesõnad:

veebipood, Django, Python, veebileht

CERCS: P175 Informaatika

The creation of a webstore for company Emart Auto OÜ

Abstract:

The aim of the Bachelor's thesis is to create an online store for Emart Auto OY. The work includes the description of the company's background, the description of customer requirements, the description of the technologies used, the description of the development process, the description of the structure of the project and the project database, and an analysis of the work done.

Keywords:

web store, Django, Python, website

CERCS: P175 Informatics

1. Sissejuhatus	5
2. Varasemalt ellu viidud lahendused ja nende analüüs	6
3. Nõuete analüüs	7
4. Kasutatud tehnoloogiad	8
5. Arendusprotsess	10
5.1 Esilehe moodul	10
5.2 Sirvimise moodul	10
5.3 Ostu moodul	11
5.4 Päise moodul	11
5.5 Administreerimise ja halduse moodul	11
6. Andmebaasi mudelid, andmed ja päringud	13
6.1 Üldine andmestruktuur	13
6.2 Keelepõhine andmestruktuur	14
6.2.1 Toodete ja kategooriate tõlked	14
6.2.2 Lehe vaadete tõlked	14
7. Funktsionaalsus	16
7.1 Avalikud vaated	16
7.1.1 Esilehe mooduli vaated	16
7.1.2 Sirvimise mooduli vaated	16
7.1.3 Ostu mooduli vaated	17
7.1.4 Päise mooduli vaated	17
8. Disain ja eesrakendus	19
8.1 Disainiotsused	19
8.2 Lehe disainide skaleerimine	20
8.3.2 Telefonivaadete disainid	23
9. Lehe analüüs	24

9.1 Google PageSpeed analüüs	24
9.2 SEO Site Checkup analüüs	25
10. Kokkuvõte	27
Kasutatud kirjandus	28

1. Sissejuhatus

Ettevõtte Emart Auto OÜ tegeleb auto tarvikute ja varuosade jaekaubandusega. Ettevõtte alustas 1998. aastal Kuusalus varuosapoe, remonditöökoja ja toidupoena ning on tänaseks arenenud Fixus Lasnamäe OÜ keti alla [8]. Sellele vaatamata puudub siiani internetipõhise kaubanduse võimekus. Tootekataloog on suur, ligikaudu 100000 toodet ning hinnavahe on varieerub mõnest sendist kuni mõnekümne tuhandeni. Müüdavat seob ühine autopõhine temaatika, mis leiab kajastust ka tulevase veebikaubamaja disainis. Ettevõtte laieneb luues Lasnamäele uut kõrgtehnoloogiat sisaldavat autoremonditöökoda. Emart Auto OÜ soov on siduda loodav e-pood mainitud töökojaga, et tekitada sidus terviklik teenus, mis innustaks klienti kasutama ettevõtte töökodasid varuosade ostul ning tõmmata tähelepanu varuosade turul tekkivale töökojale ning viia klienti kurssi toodete valikuga ja hindadega.

Fixuse keti all on kõik poed eraldi ettevõtted. Kõikidel poodidel on vaba voli luua oma teenuseid. Fixuse brändil on hetkel Eestis kaks veebipoodi, mis on üle kümne aasta töös olnud. Esimene neist ehk fixus24.ee on Fixus Mustamäe OÜ veebipood ning Balti Autoosade AS ettevõttele kuuluv veebipood pood.fixus.ee on Fixuse brändi oma.

Käesoleva bakalaureusetöö eesmärk on luua ettevõttele Emart Auto OÜ veebipood koos kaasaegsete ja turvatehniliste lahendustega.

Rakenduslik bakalaureusetöö on jaotatud kümneks peatükiks. Teises peatükis on varem tehtud tööde analüüs. Kolmandas peatükis on kirjeldatud kliendi nõudeid. Neljandas peatükis on kirjeldatud kasutatud tehnoloogiaid. Viies peatükk käsitleb arendusprotsessi kulgu. Kuues peatükk kirjeldab andmebaasi struktuuri ja loodud mudelit. Seitsmes peatükk käsitleb projekti raames loodud funktsionaalsust. Kaheksas peatükk käsitleb disaini ja loodud disainiotsuseid. Üheksandas peatükis on analüüsitud tehtud tööd. Kümnes peatükis on bakalaureusetöö kokkuvõte.

2. Varasemalt ellu viidud lahendused ja nende analüüs

Varasemalt on ettevõtte Emart Auto OÜ ja Fixus Lasnamäe OÜ loonud sarnased kodulehed Python keelel põhineval veebirakenduste raamistikul Django [1]. Lisaks eelmises peatükis välja toodud Fixuse brändi e-poodidele, on Fixus Lasnamäe OÜ viie kilomeetri raadiuses peamised konkurendid <https://www.motonet.ee>, <https://alvadi.ee/> ja <https://autovaruosad.ee/>, kellest peamiseks konkurendiks on Motonet. Motonet Eesti üks suurimatest esindustest asub Fixus Lasnamäe OÜ poest ja ühtlasi ka ehitatavast autoremonditöökojast viiesaja meetri kaugusel.

Kõigil mainitud poodidel on peamine funktsionaalsus neljas järgnevas osas: tooteotsing ning sirvimine, ostukorvi ja maksmise süsteem, esileht allahindluste ja reklaamitud kauba kuvamiseks ning haldussüsteem.

Tooteotsing on kõigil poodidel sarnase ülesehitustega. Implementeeritud on tootefotode kuvamine, toodete filtreerimine ja tootelehtede vaated. Ostukorvi süsteem töötab kõigil lehtedel samuti sarnase funktsionaalsusega. Võimalik on lisada, eemaldada ja muuta tooteid ning toodete koguseid, vormistada ostu ning maksta. Esilehel on kõigil poodidel uudistooted ning erinevate kriteeriumite põhjal esile toodud tooted.

3. Nõuete analüüs

Kliendiga suheldes selgus, et Emart Auto OÜ ettevõttes puudub programmeerimise oskusega spetsialist, seega rakendus peab olema hallatav ilma IT taustata töötajate poolt. Lisaks on soov jätta lahtiseks makseteenuste võimalused, et saaks tulevikus eemaldada, lisada, muuta ja rakendada erinevaid teenuspakkujaid.

Hetkel on autode varuosi müüvatel poodidel trend implementeerida rakendusse auto registrimärki kasutatav otsingumootor, mis otsib poest vastava autoga ühilduvad varuosad ja serveerib otsingu tulemused kliendile. Selle bakalaureusetöö raames aga sai Emart Auto OÜ esindajaga kokku lepitud, et vastav süsteem oleks liialt ajamahukas implementeerida ning otsustasime jätta eelkirjeldatud võimekuse lisamise käesoleva töö skoobist välja.

Peamised nõuded:

1. Võimalik sirvida, filtreerida, näha ja hallata tooteid.
2. Võimalik luua ja hallata administreerivaid kasutajaid ning neile erinevaid õigusi omistada.
3. Võimalik sirvida tervet lehte vahetades keelt ning jooksvalt lisada, kustutada ja muuta keeli ning tõlkeid.
4. Automatiseeritud suhtlus ladudega, et vältida käsitsi andmesisestust.
5. Automatiseeritud perioodilised toodete impordid ladudest
 - tarneaja arvutamiseks,
 - tarnekoguse uuendamiseks,
 - toodete ja tootegruppide muudatustel, lisamistel ja kustutustel.

Haldusnõuded:

1. Süsteem peab võimaldama hallata
 - allahindlusi grupiti ja tootepõhiselt,
 - tooteinfot, pilte ning tootetõlkeid,
 - lehe tõlkeid ja sisu,
 - kasutajaid, õigusgruppe ja kasutajate õigusgruppe.
2. Süsteemi server peab võimaldama kaughaldusega ligipääsu.

Mittefunktsionaalsed nõuded:

1. Veebipoel olevat infot on võimalik lugeda mitmes erinevas keeles.
2. Leht peab olema laetav maksimaalselt sekundi jooksul.
3. Lehe sirvimisel ei teki tõrkeid.

4. Kasutatud tehnoloogiad

Kliendiga suheldes selgus, et otstarbekas on kasutada järgnevaid keeli ning tehnoloogiaid tänu varasemalt loodud lahendustele ning ennetavalt süsteemide tulevikus kokku integreerimisele:

1. Django,
2. Python,
3. HTML,
4. CSS,
5. JavaScript,
6. MySQL (SQL),
7. Bootstrap,
8. JQuery,
9. Stripe,
10. Nginx.

Täpsemalt selgus, et peamised kaks kartust on tuleviku muudatuste ja bakalaureusetöö järgsete arenduste implementeerimise keerukus ja ajakulu ning veebikeskkonna üldine turvalisus, mistõttu võeti peamiseks tehnoloogiaks Django veebiraamistik [1]. Teised tehnoloogiate valikud on orienteeritud just selle raamistiku kasutuselevõtust.

Django valiti tänu raamistikku sisse ehitatud turvameetmetele.

Django on Pythonil üles ehitatud veebirakenduste raamistik, mis seob Pythonit HTML-i, CSS-i ja JavaScripti. Django raamistik loodi 2005. aastal, et tagada projektidele mastaapsus, kiirus ja turvalisus. Lisaks Djangole on töös kasutusel veel Bootstrap ja JQuery raamistikud. Bootstrap [2] on HTML-i, CSS-i ja Javascripti kasutatav teek, mis muudab disainikeelte HTML ja CSS kasutamist läbi ette loodud stilistika. Peamine Bootstrap raamistiku rakendus on veebilehe disaini kuvamine igale seadmele sõbralikult. JQuery [3] on Javascripti kasutatav teek, mida rakendatakse HTML elementide animeerimisel ja manipuleerimisel. Lisaks on võimalik JQueryga kirjutada otse JavaScripti, näiteks toodete filtreerimisel ja erinevate tootehindade arvutuste lihtsustamisel.

Antud veebiraamistikus on Django loojate poolt implementeeritud mitmeid kaitsemeetmeid, millest tuntumad on järgnevad: *cross-site scripting (XSS) protection*, *cross-site request forgery (CSRF) protection*, *SQL injection protection*, *Clickjacking protection*, *SSL/HTTPS*,

Host header validation [1]. Django raamistik võtab turvalisuse implementeerimisega arendajalt enamuse tööd ära ning turvaaukude avastamisel teavitab kasutajat koheselt või koguni peatab veebirakenduse töö kuni oht on eemaldatud. Lisaks turvalisusele on antud raamistiku struktuur üles ehitatud väiksematele moodulitele, et tagada tulevikus lihtne segmenteeritavus ja võimekus uuele arendusele [1]. Näiteks on Eestis Django raamistikul ehitatud mõned suurimad veebirakendused Ecoop, Postimehe koduleht ja Veebimajutuse koduleht [5].

Bootstrap on disainiraamistik, mis seob HTML, CSS, JavaScript ja JQuery baasil loodud elemente kasutajaliidese kujunduse loomiseks [6].

Domeenihalduseks valiti arutelu käigus Zone Media ja kliendiga arutelu käigus selgus soov hoida kõiki teenuseid ja servereid ühe haldaja juures, seega valikuks osutus DataCode OÜ veebimajutus, kus veebiserver hakkab jooksuma Nginx mootoril [5].

5. Arendusprotsess

Django koodistruktuurile omaselt jaguneb veebipood mooduliteks [1]. Igal moodulil on oma andmebaasi mudel, läbi mille toimub objektorienteeritud andmehoiustus ning kõik moodulid suhtlevad omavahel. Projekt jaguneb viieks peamiseks järgnevaks vaherakenduseks ehk mooduliks: esilehe moodul, sirvimise moodul, ostu moodul, päise moodul, administreerimise ja halduse moodul (vt. Joonis 1).

5.1 Esilehe moodul

Esilehe moodul ehk *index* moodul on moodul, kus paikneb kõik esilehega ja esmase külastusega seostuv info. Selles moodulis

- luuakse kõik esialgsed sessiooniinfo sissekanded,
- serveeritakse kasutajale ette esilehe kood, mis kasutajale tõlgendub ümber lehe vaateks,
- näidatakse browser moodulist toodete vaadetega populaarseimad tooteid,
- kaardistatakse külastaja info,
- näidatakse kasutajale navbar moodulist laetud lehe tõlked.

5.2 Sirvimise moodul

Sirvimise moodul ehk *browser* moodul on moodul, kus paikneb kõik toodetega seostuv info. Selles moodulis hoiustatakse toodete müügiinfot, tõlkeid toodete kohta ja toodete pilte. Lisaks serveeritakse kasutajale toodete sirvija vaated koos koodiga, mis tõlgendub ümber lehe vaateks, kus saab otsida ja filtreerida tooteid. Lisaks kaardistatakse külastajate toodete vaatamised ning otsingud, serveeritakse ette *navbar* moodulist lehe tõlked ja tehakse kõik toodetega seostuvad arvutused ja päringud. Samuti on sirvimise moodulis implementeeritud API toodete importimisloogikaga ning tootehaldusloogikaga.

5.3 Ostu moodul

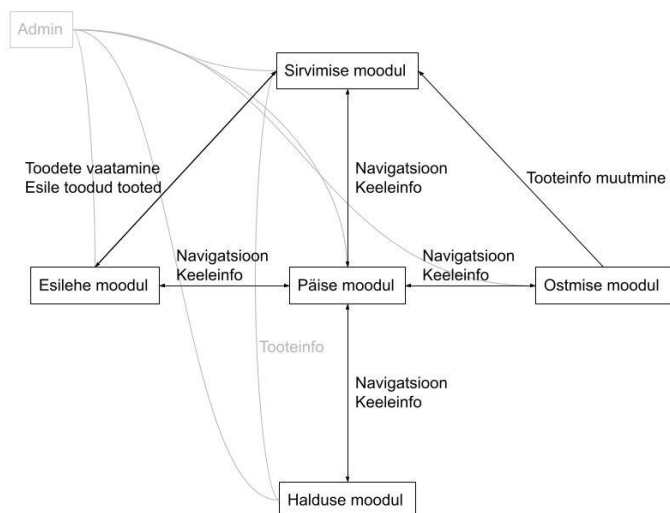
Ostu moodul ehk *checkout* moodul on moodul, kus paikneb kogu müügiga seostuv info. Selles moodulis viiakse läbi maksed, ostukorvi haldus, ostutšekkide genereerimine ja hoiustus. Lisaks serveeritakse ette *navbar* moodulist lehe tõlked ja laetakse *browser* moodulist ostukorvile sarnaste toodete vaateid, et neid pakkuda kliendile.

5.4 Päise moodul

Päise moodul ehk *navbar* moodul on moodul, kus paiknevad kõik lehe tõlked ja üldine navigatsiooniga seostuv informatsioon. Päise moodul on integreeritud igasse teise moodulisse välja arvatud administreerimise ja halduse moodul. Mooduli peamine eesmärk on serverida kasutajale iga parasjagu külastatava vahelehe tõlked ning talletada üldist kontaktinfot ja päise tõlkeid.

5.5 Administreerimise ja halduse moodul

Administreerimise ja halduse moodul ehk *admin* moodul on moodul, kus paikneb kogu andmebaasi halduse funktsionaalsus ning kasutajatele õiguste jagamise funktsionaalsus. *Admin* moodulisse on juurdepääs ainult privilegeeritud kasutajatel, kes on pandud vastavasse õigusgruppi. Administreerimisõigusega kontoga sisenedes lisanduvad kasutajale veel kõigi teiste moodulite haldamise tööriistad. Suur hulk *admin* moodulist on genereeritud Django raamistiku poolt, mida muudeti vastavalt kliendi soovidele [1].



Joonis 1: Veebilehe moodulite suhtlus.

6. Andmebaasi mudelid, andmed ja päringud

Loodud veebipoe funktsionaalsuses on vaja mitmekeelsust ja optimeeritust, mis tuleneb andmete rohkusest, ning lihtsat struktuuri, et tagada administreerimise kergus ja kiirus (vt Joonis 2).

Django veebiraamistikule omaselt toimub suhtlus andmebaasiga läbi Python programmeerimiskeeles defineeritud klasside. Django baaskood kirjeldab klassid ja nende objektides hoitud andmed ümber SQL keeles funktsioneerivateks tabeliteks ning sissekanneteks ja vastavalt toimub ka päringute tegemine läbi loodud klasside.

Kõik projektiga seotud andmed hoitakse vahelehtede ehk vaherakenduste põhiselt ning andmete pärimine toimub vaherakenduste vaadetes, kuhu on kirjutatud ka ülejäänud vaadete funktsionaalsus.

Päritud ja töödeldud andmed saadetakse vaadetest eesrakenduste mallidele kaasa vastuste süsteemi kaudu, mis on Python keeles defineeritud sõnastikuna, millele saab ligi eesrakenduse mallis DTL (Django Template Language) abil, mille loogika, kirjastiil ja funktsionaalsus on ehitatud üles väga sarnaselt Jinja2 mallide loomise teegile [1].

6.1 Üldine andmestruktuur

Projekti vaherakendusi saab kategoriseerida kahte järgnevasse gruppi: avalikud ja mitteavalikud vaherakendused.

Mitteavalikud vaherakendused on peamiselt loodud administratiivsel eesmärgil või töötajate haldamise eesmärgil ning ei sisalda tõlkeid. Vaherakendustes hoiustatud andmed on ajaloolised, näiteks ostukorvide ajalugu, statistiliseks analüüsiks vajaminevad andmed, näiteks külastatud klientide riigid ja keelevalikud või raamatupidamiseks vajaminevad andmed nagu näiteks kviitungite moodustamiseks vajaminevad andmed.

Avalikud vaherakendused on peamiselt loodud veebipoe külastajate jaoks ning hoiustavad lehe tõlkeid, tooteinfot ning tootetõlkeid.

Projekti üldine keelte süsteem paikneb päise rakenduses “Navbar”, klassis “Lang”, ning on kasutatud kõikides teistes tõlkeklassides, kus on vihjatud välisvõtmega “Lang” klassi objekti.

Kõik välisvõtmelised suhted on loodud kaskaadi kustutamise printsiibil, ehk kui kustub peaklassi objekt, kustuvad vastavad alamklasside objektid [10].

6.2 Keelepõhine andmestruktuur

Tõlgete hoiustamise struktuur andmehoiu mõttes jaguneb selle projekti raames kaheks: toodete tõlked ning lehe vaadete tõlked.

Tõlgete peaklass “Lang” on klass, kus hoitakse keele nime, lipu ikooni ning Google standardil keelekoodi [9] (vt Joonis 1).

6.2.1 Toodete ja kategooriate tõlked

Toodete ja kategooriate tõlked hoitakse eraldi klassis, mis on seoses toote peaklassiga läbi välisvõtme ja tõlgete peaklassiga läbi välisvõtme. Kõik toodete ja kategooriatega seostuv info hoitakse sirvija vaherakenduses “Browser”.

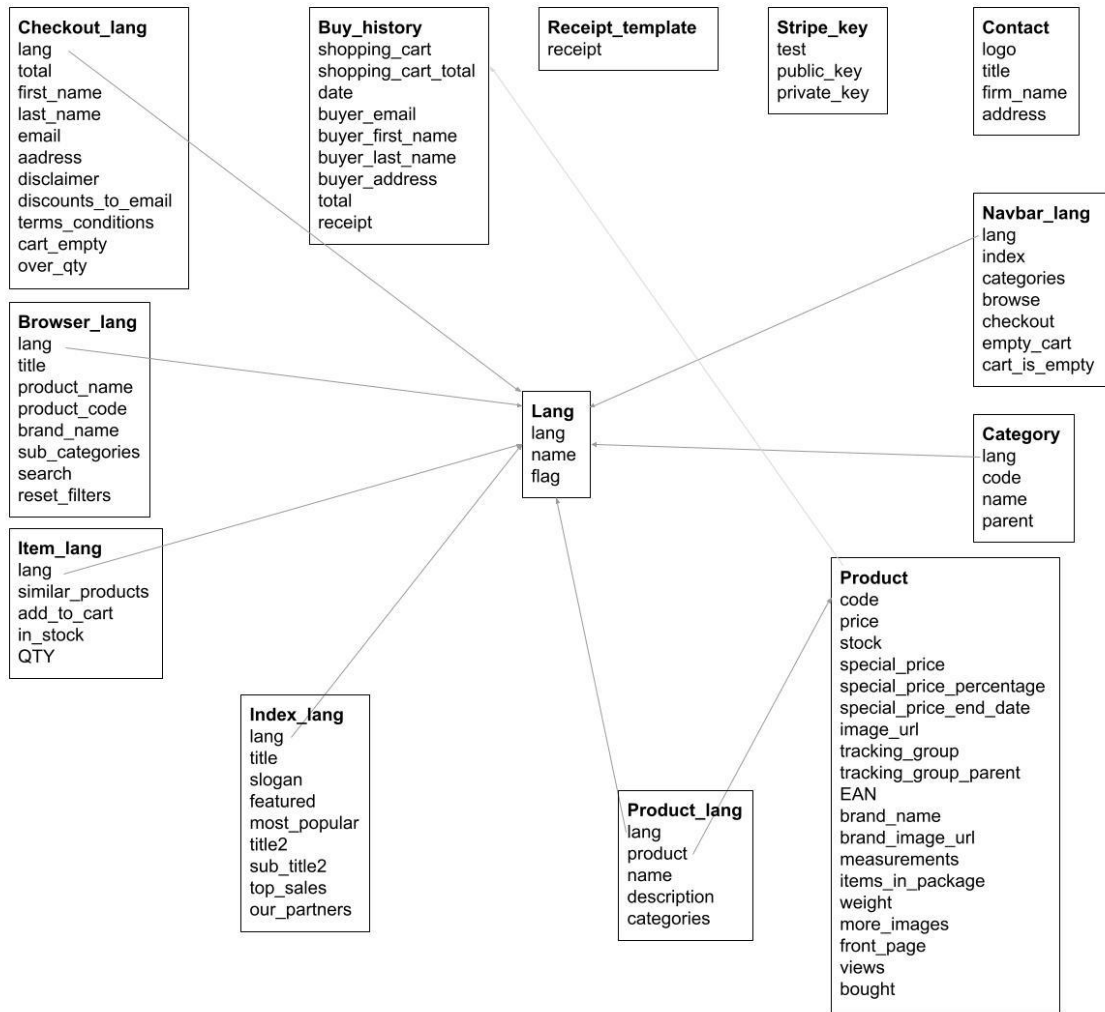
Toote peaklass “Product” on seoses toote tõlkeklassiga “Product_lang” ning toote tõlkeklass ja kategooriaklass on seoses “Lang” ehk keele peaklassiga. Tooted ei ole otseselt seotud kategooriatega, aga kategooriad on üles ehitatud sõnedepõhises struktuuris, mille infot hoiavad mõlemad klassid ning kuuluvuse kontroll on alamsõne sisaldamise kontrollimises.

6.2.2 Lehe vaadete tõlked

Lehe vaadete tõlked paiknevad igas avalikkusele mõeldud vaherakenduses. Vaadetõlgete nomenklatuur on ehitatud loogikal, et esmalt on nimetatud vaherakenduse nimi, seejärel alakriips ja “lang”. Näiteks, kui avalehe vaherakenduse nimi on “Index”, siis vastava tõlkeklassi nimi on “Index_lang” ning vastav alamklass sisaldab antud vaherakenduse vaadete tõlkeid, välisvõtit peamisesse keeleklassi “Lang” näiteks järgnevalt:

- klass “Index_lang”
- lang, välisvõti klassi “Lang”
- ...

Lisaks on sarnaselt hoiustatud toodete sirvimiseks mõeldud vaherakenduses “Browser” erinevate nuppude tõlked, näiteks sarnased tooted, lisa ostukorvi, tarnekogus, jm.



Joonis 2: Veebilehe andmebaasi joonis.

7. Funktsionaalsus

Django veebiraamistikus paikneb kogu funktsionaalsus vaaherakenduste vaadetes. Vaated on funktsioonid, mis võtavad sisendiks veebitaotluse ning tagastavad veebivaste, mis koosneb veebitaotlusest, eesrakenduse mallist ja kontekstist ehk sõnastikust, mis sisaldab arendaja poolt kaasa antud muutujatest eesrakenduse mallis rakendamiseks [1]. Projekti raames algab andmeedastus projekti peakaustas asuvatest sätete ja lingi failidest. Peasätete failis on defineeritud kõik projekti käitamiseks vajalik ning lisaks on defineeritud projekti meedia ning staatiliste failide kaustad [1]. Pealinkide failis on defineeritud kõik suunamised moodulite lingifailidele ning suunamised väikeikooni, robotite tekstifailile ning lehe kaardistuse tekstifailile. Moodulite lingifailid ühendavad suunamised ja moodulite vaated, et ühendada iga vahelink õige vaatega [1]. Django veebiraamistikule omaselt on kõik kirjutatud Python programmeerimiskeeles [1].

Sarnaselt andmestruktuurile saab kategoriseerida veebipoe funktsionaalsust (vt Joonis 3) ehk vaaherakenduste vaateid kahte gruppi: avalikud vaated ja administratiivsed vaated.

7.1 Avalikud vaated

Avalikud vaated asuvad esilehe, sirvimise, ostu ja päise moodulites. Iga avaliku vaate alguses on viide päisemooduli funktsioonile “init_lang”, mille eesmärgiks on kontrollida, et külastaja sessiooniinfos oleks olemas keelemuutujad, keelevelik, ja navigatsioonikategooriad ning puudumise juhul teha esmased sissekanded ning luua vastavad muutujad sessiooniinfosse. Lisaks tehakse igas avalikus vaates andmebaasipäringud vastava vahelehe tõlgetele.

7.1.1 Esilehe mooduli vaated

Esilehe mooduli vaadetes on kolm andmebaasipäringut, kus laetakse mällu defineeritud muutuja jagu kõige populaarsemaid tooteid, allahinnatud tooteid ja esilehele välja pandud tooteid ning pannakse päringud eesrakenduse malli kontekstiga kaasa.

7.1.2 Sirvimise mooduli vaated

Sirvimise mooduli vaadetes on defineeritud toodete filtreerimise funktsionaalsus, tootevaated, toodete importimisloogika ja kategooriate lehitsemise funktsionaalsus. Toodete filtreerimise

süsteem on toote nimepõhine filtreerimine, brändi nimepõhine filtreerimine, toote koodipõhine filtreerimine ja hinnapõhine filtreerimine. Lisaks saab kategooria lehitseja abil filtreerida tooteid ka tootegruppide ehk toodete kategooriate kaudu. Filtreeritud tooted lastakse läbi *paginator* teegi, mille eesmärk on laadida 24 toodet korraga ning seejärel laetakse andmed eesrakenduse malli. Juhul kui kasutaja on kategooriate kaudu tooteid filtreerinud, saadetakse malli kaasa ka kategooriad. Juhul kui kasutaja on valinud vaatamiseks ühe kindla toote, serveeritakse talle ette tootevaade, mis laeb üksiku toote vaatamise malli ning saadab malli lisaks tootetõlgetele ka sarnaseid tooteid. Toodete importimises on kasutatud päringut Motoral Eesti lao API-st, kus tõlgendatakse ümber toorest andmestikust andmebaasi sissekanded ning kontrollitakse toote eksisteerimist andmebaasis. Motoral Eesti lao API väljastab andmeid csv formaadis, mille alusel on loodud tooteinfo talletamiseks mudel. Väljastatud andmestikus on kõik müüdavavad tooted, toodete info, mis on müümiseks vajalik, toodete eestikeelne tõlge ning tootegrupid.

7.1.3 Ostu mooduli vaated

Ostu mooduli vaadetes on ostukorvi vaade ning erinevad ostukorvi loogikat sisaldavad funktsioonid. Ostukorvi funktsioonidega on lingipõhiselt ära defineeritud ostukorvi lisamise, ostukorvi tühjendamise, ostukorvis oleva toote koguse muutmise ja ostukorvi alustamise loogika. Lisaks on ostu moodulis defineeritud funktsioonid maksete taotlemiseks ning ostujärgselt tarnekoguste ja müügistatistika uuendamise funktsionaalsus.

7.1.4 Päise mooduli vaated

Päise moodulis ei ole ühtegi vaadet ja on kaks funktsiooni, millest esimese eesmärk on kasutaja sessiooniinfo puudumisel luua uued sissekanded ning teise eesmärk on registreerida kasutajapoolne keele muutmine sessiooniinfosse. Päise mooduli loogikas tekitatud ja hoitud sessiooniinfo on enamjaolt päisega seotud info nagu näiteks soovitud keel, navigatsiooniribas ehk päises valitavad keeled ja kategooriate tõlked.

7.2 Administratiivsed vaated

Administratiivsed vaated paiknevad kõik halduse vaaherakenduses. Administratiivsetes vaadetes on peamine vaade haldusvaade, mis on töötajapaneeli navigatsioonivaade. Seal

8. Disain ja eesrakendus

Django veebiraamistikus on eesrakenduse renderdamisloogika ehitatud DTL (Django Template Language) abil, mis seob sisuliselt HTML programmeerimiskeele Python programmeerimiskeelega. Kogu visuaalne programmeerimine on tehtud HTML ja CSS keeltes ning disaini implementeerimisel on kasutatud JQuery ja Bootstrap 4 teeke.

Eesrakenduse malli struktuur koosneb kas kolmest või neljast komponendist vastavalt vaaherakendusele ja parasjagu aktiveeritud vaatele. Esimesena laetakse alati mallu vaate baasmall, mida enne ei loeta kuni laetakse mallu päise mall, kus on kõik teekide importimised, üldine stiil ja päise ehk navigatsiooniriba mall. Seejärel kui päis on läbitud, töötatakse läbi vahelehe sisu. Juhul, kui malli on kirjutatud toodete kuvamine, laetakse igal toote iteratsioonil mallu toote info ja läbitakse toote mall ja stiil. Viimaks laetakse mallu jalus, kus on jaluse mall ning üleüldine JavaScriptis kirjutatud loogika. Vaatepõhine stiil imporditakse sisu malli alguses ning JavaScriptis kirjutatud loogika malli lõpus.

8.1 Disainiotsused

Kuna ettevõtte Emart Auto OÜ on tihedalt seotud brändiga Fixus, on tekkinud Emart Auto OÜ brändile disainimise standardid, mis tulenevad osalt Fixuse disaini kataloogist ja tavadest ning osaliselt ettevõtte ajaloolistest disainielementidest, näiteks Emart Auto OÜ logo. Nii ettevõtte tootemargis kui ka lehe disainis on defineeritud baasvärvid ja kirjastiil. Veebisaidi peamine värvitoon on sinine värvikoodiga #0077B6 ning Emart Auto OÜ logo punasest tulenev punane värvikoodiga #D32626 (vt Tabel 1). Ülejäänud veebipoes kasutusel olevad värvid on tuletatud sinise erinevatest varjunditest. Kasutusel olevad kaks peamist kirjašrifti on Montserrat ja Proxima Nova. Selle lehe raames on peamiseks šrift defineeritud Montserratiks. Peamiselt on lehe sisu kirjutatud peenema paksusega, vahemikus 250 kuni 350, ning pealkirjades on kasutusel vastavalt pealkirja suurusele suuremad paksused.

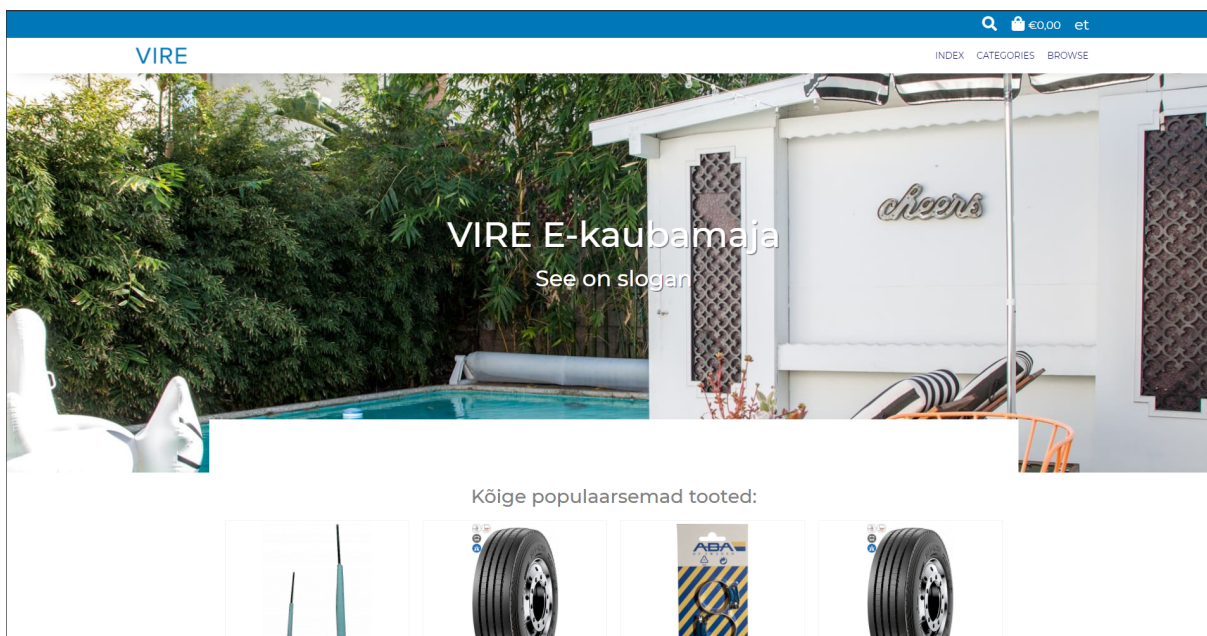
Tabel 1: Veebisaidi värvid ja värvikoodid.

#03045e tumesinine	#0077b6 sinine	#00b4d8 heledam sinine	#90e0ef helesinine	#caf0f8 hall	#d32626 punane	rgba(155,155,155, .1) ääris	rgba(0,0,0,.4) tumedam ääris
-----------------------	-------------------	---------------------------	-----------------------	-----------------	-------------------	-----------------------------------	---------------------------------

8.2 Lehe disainide skaleerimine


Lehe külastajad jagunevad peamiselt kahte kategooriasse: telefonipõhised kasutajad (vt Joonised 9-14) ja arvutipõhised kasutajad (vt Joonised 4-8). Üldiselt ei pea kirjutama eraldi lehe malle mõlemale kasutaja kategooriale tänu Bootstrap teegile, mis võimaldab lihtsasti implementeerida lehe sisu skaleerimise. Bootstrap võimaldab implementeerida seksioonisiltidele resolutsioonipõhiseid protsentlauseid, mis tähendab, et saab sisuga luua dünaamilisi joonestikke, mis säilitab kasutaja jaoks sisu loetavuse [1].

Eraldi loodi aga päise navigatsioonimenüü. Esialgu saadetakse mõlema vaate kood kasutajale, seejärel otsustatakse veebibrauseri põhjal, kumb vaade ära peita (telefon või arvuti). Arvutipõhine navigatsioonimenüü on horisontaalse disainiga, kus on navigatsiooniriba sisse implementeeritud ka ostukorv (vt Joonised 10,11). Telefonivaate päise navigatsioonimenüü on esialgu kahe nupu taha ära peidetud. Telefoni ekraani paremal üleval on päisemenüüd kuvama panev nupp, mis toob esile vertikaalse disainiga päisemenüü ning vasakul üleval ostukorvi kuvama panev nupp (vt Joonis 9).



Joonis 4: Esilehe ekraanitõmmis.

VIRE €0,00 et
INDEX CATEGORIES BROWSE





5840053 - 21-RRT110 -

REHVISEEN MAXISEAL. 3MM. TORKEAUKUDELE. TTK TRUFLEX PANG

PANG

3 mm varre läbimõõduga "rehviremondiseen" sobib mootorratta- ja sõiduautorehvidele (ka RFT) kiirusindeksiga T. H. V. W. Z kuni 3 mm läbimõõduga turvise torkeaukude, mis ei ole suurema nurga all kui 25°, paikamiseks. Lihtsalt kasutatav, tugev ja vastupidav. Maksimaalne nakkuvus - isevulkaniseeruv täidis ("seene jalg") + keemiliselt vulkaniseeruv kummipaik (kasuta kummililim "Supersolution" või "Blue Maxibond Cement"). Kvaliteeditit ületab kõik rehvitootjate rehvirmondistandardid ja vastab kõikidele globaalsetele rehvitootuse standarditele. Sobib nii radiaal- kui ka diagonaalrehvide paikamiseks.

 0.0 kg

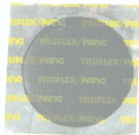
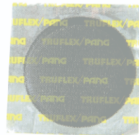
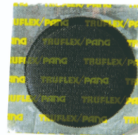

 60 tk




€1.00 | ~~€2.00~~

Until: April 23, 2021
Tarnekogus: 197

Sarnased tooted [Lisa korvi](#) **Tk: 1**

Sarnased tooted


 PAIK SISEKUMMILE 30MM. PP-0. TRUFLEX PANG €0.25 PANG Tarnekogus: 0 Sarnased tooted	 PAIK SISEKUMMILE 35MM. PP-1. TRUFLEX PANG €0.35 PANG Tarnekogus: 0 Sarnased tooted	 PAIK SISEKUMMILE 45MM. PP-2. TRUFLEX PANG €0.60 PANG Tarnekogus: 102 Sarnased tooted	 PAIK SISEKUMMILE 55MM. PP-3. TRUFLEX PANG €1.07 PANG Tarnekogus: 76 Sarnased tooted
---	---	---	--

Joonis 5: Veebilehe tootevaade.

VIRE €1,00 et
INDEX CATEGORIES BROWSE

KEEMIA TOOTED



Toote nimi

Toote kood

Brändi nimi


Sooduskaubad ainult


€ 0
●
●
 € 100000


[OTSI](#) [NULLI FILTRID](#)


Alamkategoriad

1 / 56





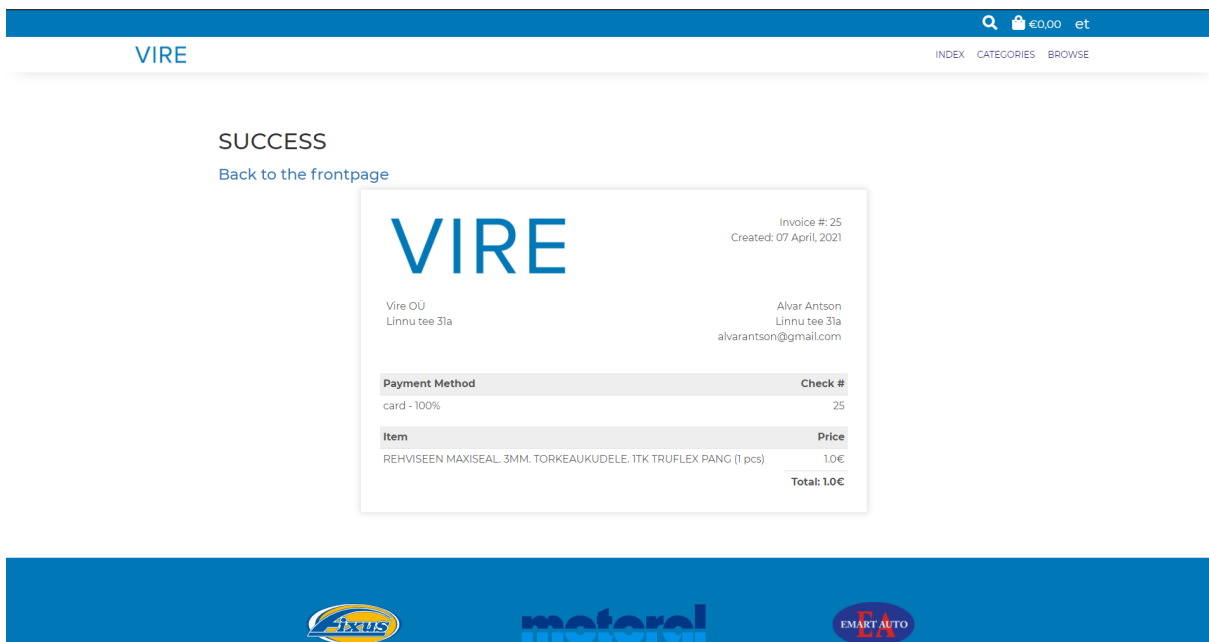




Joonis 6: Veebilehe toodete sirvimise vaade.

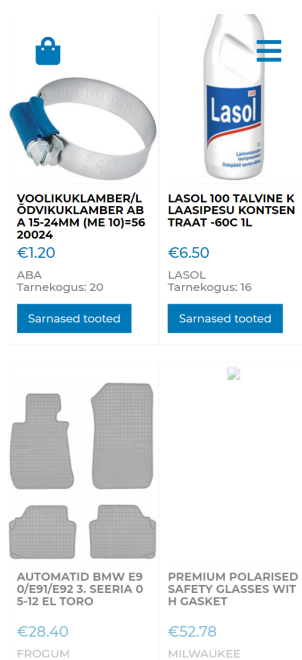


Joonis 7: Veebilehe ostukorvi vaade.

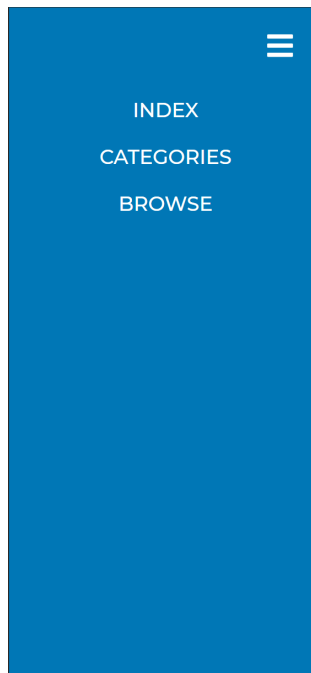


Joonis 8: Veebilehe kvitungi vaade.

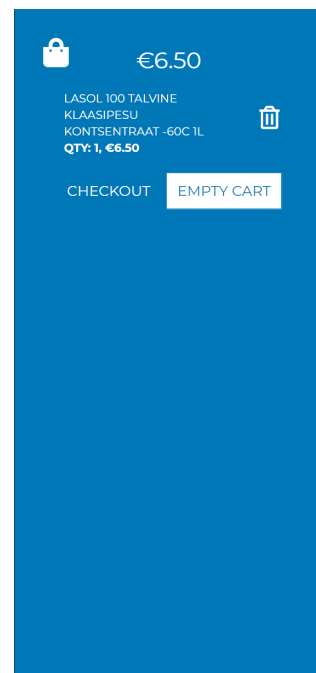
8.3.2 Telefonivaadete disainid



Joonis 9: Telefoni toodete sirvimise vaade.



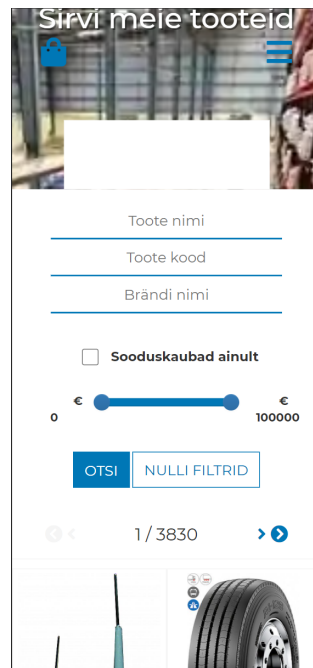
Joonis 10: Telefoni navigatsiooniriba.



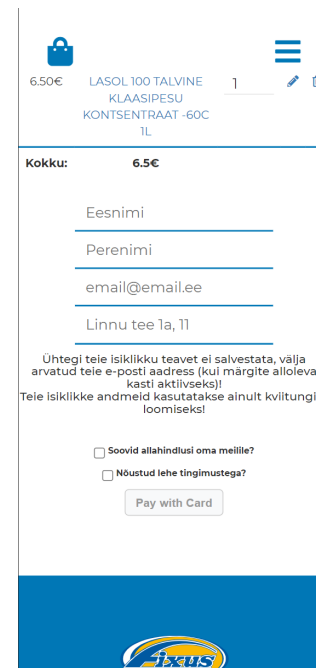
Joonis 11: Telefoni päise ostukorvi ülevaade.



Joonis 12: Telefoni tootevaade.



Joonis 13: Telefoni toodete filtreerimise vaade.



Joonis 14: Telefoni ostukorvi vaade.

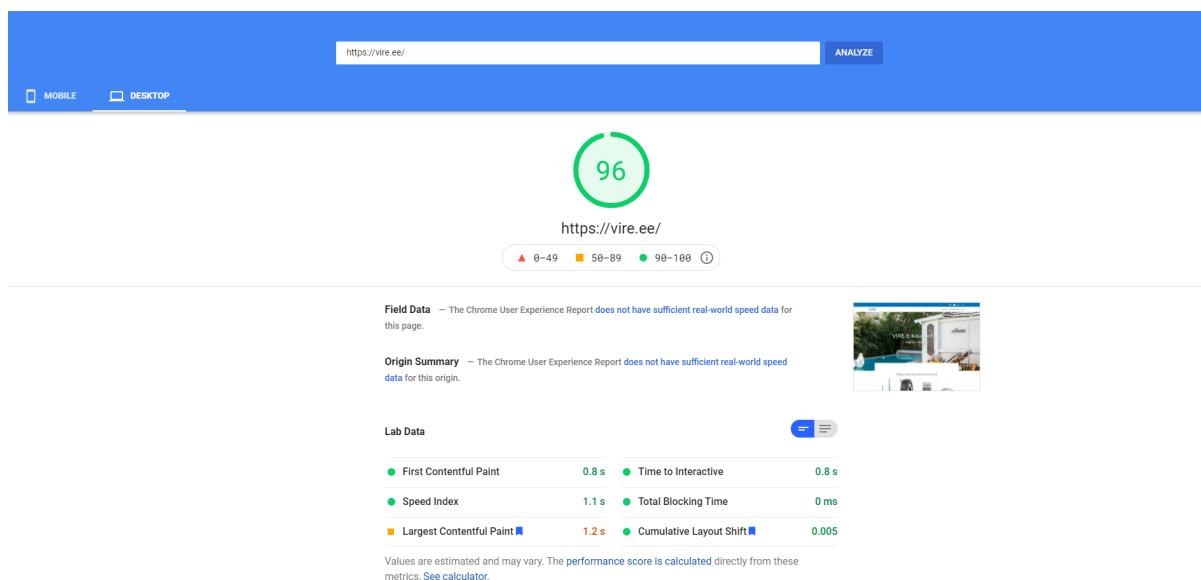
9. Lehe analüüs

Veebipoe analüüs koosneb Google-i PageSpeed Insight analüüsist [11] ja SEO Site Checkup Google'i otsingumootori optimeerimise analüüsist [12].

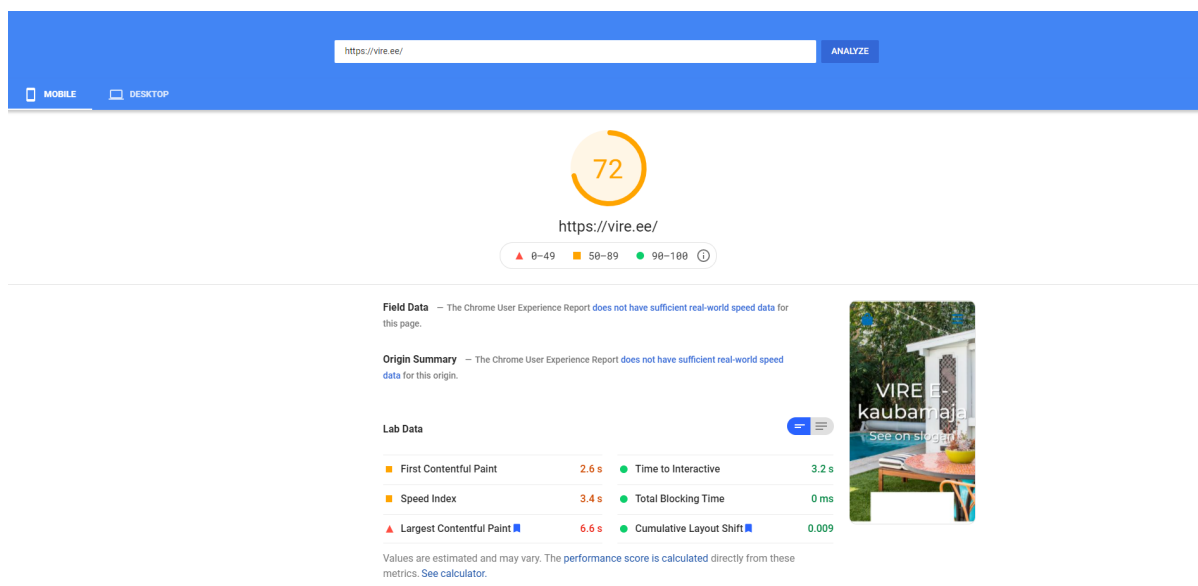
9.1 Google PageSpeed analüüs

Google PageSpeed analüüs on mõeldud veebilehe kiiruse ning laaditavuse testimiseks. Google PageSpeed sirvib robotite abiga analüüsitava lehe läbi ning seejärel moodustab rapordi, milles toob välja soovitusi lehe optimeerimiseks ning annab kasutajale tagasisidet lehe teekide ja üldise laaditavuse ajakulude kohta [11] (vt Joonised 15,16).

PageSpeed testidel selgus, et arvutikasutajatele on leht paremini optimeeritud kui telefone kasutavatele külastajatele. Peamine laadimise erinevus tuli külastajale serveeritavatest pildisuurustest. Mõlematele kasutajatele laetakse samad tootefotod, aga Google'i hinnangul ei ole tarvis telefonirakenduses kasutada nii suuri pilte. Probleemi lahendamise keerukuse tagab piltide serveris mittehoidmine. Kõik tootefotodega seonduv laetakse Motorola Eesti API kaudu ning toodete piltidel on fikseeritud suurused.



Joonis 15: Veebilehe arvutiversiooni kiirustesti tulemus lehel Google PageSpeed.



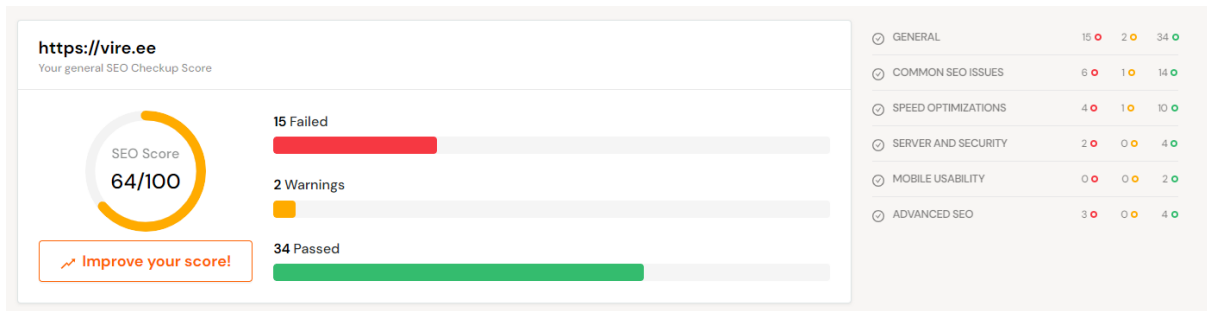
Joonis 16: Veebilehe telefoni versiooni kiirustesti tulemus lehel Google PageSpeed.

9.2 SEO Site Checkup analüüs

SEO Site Checkup analüüs [12] on mõeldud Google'i otsingumootori vastete leidmise optimeerimiseks ning on üks mitmest tasuta teenusepakkujast (vt Joonis 17).

SEO Site Checkup tööriist leidis, et peamised murekohad on lehe linkide mittesõbralikus nomenklatuuris, meta võtmesõnade mittedünaamilises kaardistamises, www.vire.ee ja vire.ee mitte ühte kohta suunamises, sotsiaalmeedia mitte kasutamises ning veateadete nagu näiteks 404 veateate puhul külastajale avaliku lehe stiilis mitte vaate serverimine.

Kõige suurem murekoht väljatoodutest on veebilehe toodete ja kategooriate aadressides. Kuna terve leht on ehitatud toote- ja kategooriakoodide ümber, on inimesele loetavaid nimesid keeruline aadressina implementeerida tänu lehe mitmekeelsusele. Sotsiaalmeediat ei saa lehele implementeerida, kuna tegemist on alustava brändiga ning sotsiaalmeedia lehti ei ole veel loodud. Ühte kohta suunamine on serverihaldaja lahendada ning saab lahenduse kui majutaja on lõplikult projekti üles seadnud. Hetkel on leht demo faasis. Veateadete vaated on ette loodud ning serveritakse automaatselt kui leht väljub demo faasist.



Joonis 17: SEO Site Checkup tööriista veebilehe analüüsimise tulemus.

10. Kokkuvõte

Käesoleva bakalaureusetöö raames valmis uus veebipood ettevõttele Emart Auto OÜ.

Töö annab ülevaate kasutatud tehnoloogiatest, projekti struktuurist, andmebaasi ja andmemudelite struktuurist, kliendi nõuetest, arendusprotsessist ja tehtud töö analüüsist.

Klient jäi tööga rahule, veebileht täitis kõiki kliendi nõudeid ning kliendi plaan laiendada internetipõhisele jaekaubandusele läheb käiku 2021 sügisel.

Loodud veebileht on üleval aadressil www.vire.ee ja on testimise versioonis ning oste päriselt veel teha ei saa.

Kasutatud kirjandus

- [1] Django, “Django dokumentatsioon”. Kasutatud 13.01.2021, <https://www.djangoproject.com/>
- [2] Bootstrap, “Bootstrap dokumentatsioon”. Kasutatud 13.01.2021, <https://getbootstrap.com/>
- [3] JQuery, “Jquery dokumentatsioon”. Kasutatud 13.01.2021, <https://jquery.com/>
- [4] Cybernetica AS, “AKIT – Andmekaitse ja infoturbe leksikon”. Kasutatud 14.01.2021, <https://akit.cyber.ee/>
- [5] Thorgate, “Tehtud projektid”. Kasutatud 13.01.2021, <https://thorgate.eu/>
- [6] Merili Osula, “Veebisaidi loomine trükifirmale Musterprint”. Kasutatud 14.01.2021, https://comserv.cs.ut.ee/ati_thesis/datasheet.php?id=70449&year=2020
- [7] Nginx, “Nginx dokumentatsioon”. Kasutatud 14.01.2021, <https://www.nginx.com/>
- [8] Emart Auto OÜ, “Meist infoleht”. Kasutatud 12.01.2021, <http://emartauto.ee/>
- [9] Google Language Codes, “Google keelekoodid”. Kasutatud 28.03.2021, <https://developers.google.com/admin-sdk/directory/v1/languages>
- [10] Tech On The Net SQL Server Cascade, “SQL Server: Foreign Keys with cascade delete”. Kasutatud 28.03.2021, https://www.techonthenet.com/sql_server/foreign_keys/foreign_delete.php#:~:text=A%20foreign%20key%20with%20cascade%20delete%20means%20that%20if%20a,cascade%20delete%20in%20SQL%20Server.
- [11] Google PageSpeed, “PageSpeed insights”. Kasutatud 26.04.2021, <https://developers.google.com/>
- [12] SEO Site Checkup, “SEO Site Checkup audit”. Kasutatud 26.04.2021, <https://seositecheckup.com/>

Lisad

I. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Alvar Antson,

(autori nimi)

1. Annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose
Veebipoe loomine ettevõttele Emart Auto OÜ,
(lõputöö pealkiri)

mille juhendaja on Helle Hein,
(juhendaja nimi)

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
Commonsi 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.

Alvar Antson

07.05.2021