

Tallinna Ülikool
Digitehnoloogiaste Instituut

Adobe Photoshop tarkvara kasutamine veebidisaini ja eeskomponendi arendamiseks

Bakalaureusetöö

Autor: Paul Nestor

Juhendaja: Romil Rõbtšenkov

Autor: „2016
Juhendaja: „2016
Instituudi direktor: „2016

Tallinn 2016

Autorideklaratsioon

Deklareerin, et käesolev bakalaureusetöö on minu töö tulemus ja seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

..... (kuupäev) (autor)

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina, Paul Nestor (sünnikuupäev: 19. detsember 1993)

1. annan Tallinna Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „Adobe Photoshop tarkvara kasutamine veebidisaini ja eeskomponendi arendamiseks ” mille juhendaja on Romil Rõbtšenkov, säilitamiseks ja üldsusele kättesaadavaks tegemiseks Tallinna Ülikooli Akadeemilise Raamatukogu repositooriumis.
2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.
3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tallinnas, (allkiri ja kuupäev)

Sisukord

Sissejuhatus	5
1 Veebilehe loomise protsess	6
1.1 Disainiprotsess	6
1.1.1 Nõuete analüüs	6
1.1.2 Sõrestikmudeli koostamine.....	8
1.1.3 Graafilise kavandi ja disainifaili koostamine	8
1.2 Eeskomponendi arendusprotsess	10
2 Adobe Photoshopi tutvustus	14
3 Adobe Photoshopi kasutamine veebilehe loomisel	16
3.1 Veebidisaini koostamine	16
3.2 Veebi eeskomponendi arendamine	21
3.3 Hinnang tulemustele	23
Kokkuvõte	29
Kasutatud kirjandus	30
Summary.....	32

Sissejuhatus

Interneti kasutamine on saanud igapäevaseks – veebilehtedel käiakse lugemas uudiseid, vaatamas ilmateadet ning internetipangas makstakse arveid. Selleks, et kõiki neid toiminguid mugav teha oleks, vajab kasutaja head veebilehe disaini. Halb kasutajakogemus võib inimese viia teenusepakkuja väljavahetamiseni. Seetõttu pannakse suurt rõhku võimalikult ilusa disaini ja mugava kasutajaliidese loomisele veebilehe arenduses.

Veebidisainerite töö produktiivsemaks muutmise eesmärgil on laialdaselt kasutusele võetud Adobe Photoshop¹ tarkvara, mis võimaldab väga mugavalt graafikat vormindada ja loodud disaini projekti käigus kliendile näidata. Hiljem on võimalik disainifail üle anda eeskomponendi arendajale, kes alustab veebilehe kliendipoolse rakenduse välja töötamisega, kasutades selleks peamiselt HTML, CSS ning JavaScript koodi.

Käesoleva bakalaureusetöö eesmärgiks on tutvustada veebilehe disaini ja eeskomponendi loomise töövoogu ning anda ülevaade, kuidas seda tehakse Adobe Photoshop tarkvara abil. Autor on Tallinna Ülikooli Arvutigraafika õppeaines Photoshopi kasutamist õppinud, kuid ainekava ei näinud ette veebidisaini loomist, mistõttu käsitletakse seda protsessi töös. Ühtlasi on autoril isiklik huvi veebilehe eeskomponendi arenduse vastu ning kuna Photoshop on ka selleks tööks väga laialt kasutatav, siis püüab autor uurida antud tarkvara rolli veebilehe arendusprotsessis.

Bakalaureusetöö eesmärkide saavutamiseks kavatseb autor uurida tänapäeval kasutatavaid veebiarenduse tööprotsesse. Seejärel anda ülevaate Adobe Photoshop programmist ning siis reaalse näite abil teha läbi disaini ja eeskomponendi loomise protsessid, et tutvustada meetodikaid ning tööd lihtsustavaid vahendeid.

Töö jaotub kolmeks alamosaks, millest esimeses tutvustatakse lähemalt veebilehe arenduse töövoogu, alustades disaini loomisest, lõpetades programmeerimisega. Teises osas antakse ülevaade Adobe Photoshop tarkvarast, selle ajaloost, kasutuskohtadest ja võimalustest. Töö kolmandas osas tutvustab autor veebidisaini loomises ja eeskomponendi arenduses kasutatavaid Photoshopi meetodikaid ja töövahendeid. Ühtlasi püüab autor uurimustöö käigus saadud teadmiste põhjal Photoshop tarkvaraga abiga luua veebidisaini ning eeskomponenti.

¹ Adobe Photoshop tarkvara kättesaadav aadressil: <http://www.photoshop.com/>

1 Veebilehe loomise protsess

Tänapäeva veebiarendus on väga keeruline protsess ning hõlmab enda alla mitmeid töövoolüüsid. Disainiprotsessis võivad osaleda lisaks visuaalse disaini arendajale tihtipeale ka eraldi kasutajakogemuse ja kasutajaliidese disainerid. Nende loodud töö võtab vastu eeskomponendi arendaja (ingl *front end developer*), ning loob enamasti HTML, CSS ja JavaScript koodi kasutades kliendipoolse rakenduse. Seejärel suunatakse valminud veebilehe visuaalne osa programmeerijale (ingl *back end developer*), kes arendustöö käigus loob vajalikud serveripoolsed rakendused. Lisaks võivad töövoos eraldi osaleda projektijuht, dokumenteerija ja testija.

Järgnevalt antakse põhjalikum ülevaade disaineri ja eeskomponendi arendaja töövoost ja töövahenditest.

1.1 Disainiprotsess

Tänapäeva veebiarenduses on oluline osa disainil. Rõhku pannakse nii kasutajamugavusele, kui ka visuaalsele väljanägemisele. Veebidisainer (ingl *web designer*) loob toote visuaalse poole, kogub ja loob kasutatava graafika. Suuremates veebidisainiga tegelevates töörühmades on tihti ülesanded laiali jagatud, mis võimaldab spetsialiseeruda ning anda häid tulemusi. Enamjaolt on tööülesanded jaotatud järgmiselt (Rinde, kuupäev puudub):

- graafikadisainer (ingl *web graphics designer*) loob ja töötleb graafikafaile (illustratsioonid, bännerid jms), optimeerib neid veebi jaoks;
- kasutajaliidese disainer (ingl *user interface designer*) hoolitseb, et veebileht oleks mugavalt kasutatav (navigatsioon jms);
- kasutajakogemuse disainer (ingl *User eXperience designer*) tegeleb parima kasutajakogemuse loomisega.

1.1.1 Nõuete analüüs

Nõuete analüüs on esimene samm veebilehe valmimise protsessis. Veebiprojekti alustades on kliendil tavaliselt mõned ideed loodava lehe osas. Seevastu arendajatel on ennekõike vaja detailset ülevaadet nõudmistest ja ootustest, et tagada kliendile soovitud tulemus. Nõuete analüüsi tulemus on soovitatav dokumenteerida (Requirements analysis, kuupäev puudub).

Nõuete analüüs püüab tagada selle, et ettevõtte eesmärgid aitaksid kasutajal rahuldada vajadused ning lõpptulemus pakuks seda, milleks leht on kavandatud. Kui tegemist on uuendatava veebiprojektiga, siis analüüsitakse põhjalikult olemasoleva rakenduse puudusi. Nõuete analüüs on protsess, mis algab ajurünnakuga ning kordub igas arendustsüklis. Nõuete analüüsi dokumenti täiendatakse ning sellest saab kasulik juhised edasiste otsuste tegemisel arendusprotsessi juures (Arel, 2016).

Disaini juures on oluline kas ja millisel määral vajab loodav lehekülge kohanduvust (ingl *responsive design*), ehk elementide paigutuse ja suuruse sobitamist vastavalt lehte külastatava seadme ekraanisuurusele. Kuna tänapäeva veebilehekülge külastatakse järjest enam nutiseadmetest, siis on oluline välja selgitada, kas kohanduva veebidisaini loomine on vajalik ja mitmele eri mõõdus ekraanile on seda vaja luua.

Nõuete analüüsi juures selgitatakse välja, milliseid raamistikke (ingl *framework*) projekti juures kasutatakse. Veebilehe eeskomponendi puhul on raamistiku näol tegemist enamjaolt HTML, CSS ja JavaScript failide kogumiga, mis lihtsustavad lehe loomist. Raamistiku eesmärk on pakkuda üldkasutatavat struktuuri selleks, et arendajad ei peaks hakkama seda ise looma ning saaksid olemasolevat koodi kasutada. Sel viisil võib arendusprotsessi aeg oluliselt lühendada (awwwards-team, 2016).

Näiteks elementide joonduse (ingl *grid system*) paigutamiseks on olemas Golden Grid System² ja 960 Grid System³ raamistikud. Visuaalse väljanägemise parendamiseks Bootstrap⁴, aga ka Material Design raamistikud, näiteks Polymer⁵ või Materialize⁶. Hetkel laialdaselt kasutatav Material Design⁷ on lameda disaini (ingl *flat design*) kategooriasse kuuluv, Google'i poolt välja töötatud rakenduste disainikeel, kus elemendid muudetakse realistlikumaks tekitades neile justkui z-telje kihid, suurendades sellega ruumilisust. Algselt Android operatsioonisüsteemis kasutusel olnud meetod on hakanud leidma üha rohkem kasutust veebis, sealhulgas ka enamustel Google'i lehtedel.

² Golden Grid Systemi koduleht: <http://www.jonikorpi.com/golden-grid-system/>

³ 960 Grid Systemi koduleht: <http://960.gs/>

⁴ Bootstrapi koduleht: <http://getbootstrap.com/>

⁵ Polymeri koduleht: <https://www.polymer-project.org/1.0/>

⁶ Materialize'i koduleht: <http://materializecss.com/>

⁷ Google Material Designi koduleht: <http://www.google.com/design/spec/material-design/introduction.html#>

1.1.2 Sõrestikmudeli koostamine

Nõuete analüüsi juures koostatakse esimesed disainiversioonid, enamasti sõrestikmudelid (ingl *wireframe*), et illustreerida võimalikku lähenemist ideele või nõudele. Sõrestikmudeliga esitatakse esialgne visuaalne elementide (näiteks tekstide, nuppude ja piltide) paigutus ning tavaliselt luuakse see pliiatsiga paberile. Sel viisil on kliendil lihtsam aru saada valmivast lehest ning mudel aitab täpsustada nõudeid lõpptulemusele või vastavalt vajadusele lihtsamini elementide paigutust muuta. Kui sõrestikmudel on kliendi poolt heakskiidu saanud, asutakse disainiprotsessi juurde (Arel, 2016).

1.1.3 Graafilise kavandi ja disainifaili koostamine

Disainiprotsessi alguses võtab disainer nõuete analüüsi tulemusel valminud sõrestikmudeli ning hakkab selle põhjal valmistama kasutajaliidese prototüüpi (ingl *mock-up*), mille loomiseks on olemas mitmeid erinevaid võimalusi. Lihtsamini käsitletavad ja kättesaadavad rakendused on leitavad veebist, millest tuntuimad on näiteks Moqups⁸ ja Balsamiq⁹. Loetletud rakendused võimaldavad hiirega lohistades lisada kavandile erinevaid elemente, sealhulgas tekste, klikitavaid nuppe, ikoone ning määrata ka värve. Samuti saab nendes rakendustes luua alamlehekülgi ja nendevahelisi seoseid, mis võimaldab läbi mängida erinevaid kasutaja lehestiku navigeerimise stsenaariume. Kasutajakogemust on sellisel viisil hea testida, et vältida hiljem ilmnedu võivaid vigu ja kontrollida, kas kõik algselt seatud nõuded on täidetud. Prototüüpimiseks kasutatakse ka Invision Appi¹⁰, kuhu on võimalik sisestada graafilise disaini failid erinevate lehtedena ning seal määrata klikitavaid nuppe ja seoseid lehtede vahel. Sellisel viisil on päris kasutajatel võimalus läbi mängida lehestiku navigeerimise stsenaariume ning anda tagasisidet.

Kui sobiv prototüüp on valminud, alustatakse graafilise kavandi loomisega. Kliendiga läbirääkides otsustab veebidisainer, milline kirjastiil, värvilahendus ja kompositsioon veebilehele tulevad. Graafilisele kavandile tulevad kõik veebilehe lõppversioonis esinevad elemendid ning kavandi lõppversioon on eeskomponendi arendajale edastatav disainifail. Graafilise kavandi koostamiseks on olemas hulgaliselt erinevaid võimalusi, ning järgnevalt antakse ülevaade laialdasemalt kasutatavatest vahenditest.

⁸ Moqups on kättesaadav aadressil: <https://moqups.com/>

⁹ Balsamiq on kättesaadav aadressil: <https://balsamiq.com/>

¹⁰ Invision App on kättesaadav aadressil: <https://www.invisionapp.com/>

Kõige lihtsamini käsitletavad veebilehe loomise keskkonnad on mall-tüüpi (ingl *template-based*), kus paljude etteantud mallide seast saab valida sobiva kujunduse ja vastavalt soovile seda redigeerida. Tihti võimaldavad teenusepakkujad majutust enda alamdomeeni piires, kui tegemist pole liiga mahuka lehega. Mall-tüüpi veebilehe loomise keskkond sobib aga pigem algajale, kuna sisu tekitamine on lihtne ja kiire, kuid väheste võimalustega ja piiratud paindlikkusega. Näidetena võib mall-tüüpi veebilehe loomise teenusepakkujatest välja tuua populaarse Wix¹¹ ja eestimaise Voog¹² keskkonna.

Graafilise kavandi koostamiseks on laialdaselt kasutuses veebidisaini tarkvarad, mis lasevad veebilehe eeskomponenti luua, muuta ja uuendada. Selliste tarkvarade idee on disaineritele teha lihtsamaks veebielementide kallal töötamise läbi graafilise kasutajaliidese ning antud meetodit nimetatakse lühendiga WYSIWYG (ingl *what you see is what you get*), ehk „see, mida sa näed, on see, mida sa saad“. Sellist tüüpi tarkvara genereerib ise graafilisest veebielemendist vajaliku koodi või mõningatel juhtudel laseb ka koodi redigeerida ning vastavalt sellele muutub graafiline element. Üks enimkasutatavaid WYSIWYG tüüpi programme on Adobe Dreamweaver CC¹³. Nii disainerid, kui ka eeskomponendi arendajad kasutavad Dreamweaverit, kuna see laseb luua ja muuta loodava veebilehe elementide graafilist asetust, aga ka lähtekoodi. Adobe Dreamweaver CC võimaldab integreerida ka muud Adobe tarkvara, näiteks pilditöötamiseks vajalikku Photoshop CC'd (Arguin, Heald, & Team, 2013). Adobe Dreamweaver on kasutamiseks nii Mac OS, kui ka Windows operatsioonisüsteemiga arvutitel.

Veebidisaini loomisel annab kõige laialdasemad võimalused graafilise redaktori kasutamine. Sellisel viisil disaini loomine võib olla kliendile küll väga kulukas, kuid meetod annab personaalse ja kliendi soovidest lähtuvalt konstrueeritud tulemuse. Antud meetod tagab ka mugava võimaluse loodavat disaini kogu protsessi jooksul kliendi ja töörühmaga jagada. Enamkasutatavad programmid graafilise veebidisaini loomiseks on Sketch¹⁴ ning Adobe tarkvaradest Photoshop CC ja Illustrator CC¹⁵. Loetletud tarkvaradest Sketch ja Illustrator on praegusel ajal ka üha rohkem kasutust leidmas, kuna võimaldavad vektorgraafika loomist ja töötlemist. Valminud SVG graafikat saab veebis kuvada, muuta ja animeerida tänu HTML5, CSS3 ja JavaScript tehnoloogiate tulekule. Kuna SVG ei sõltu resolutsioonist, on

¹¹ Wix on kättesaadav aadressil: <http://www.wix.com/>

¹² Voog on kättesaadav aadressil: <http://www.voog.com/>

¹³ Adobe Dreamweaver CC tarkvara on kättesaadav aadressil: <http://www.adobe.com/ee/products/dreamweaver.html>

¹⁴ Sketchi tarkvara on kättesaadav aadressil: <http://www.sketchapp.com/>

¹⁵ Adobe Illustrator CC tarkvara on kättesaadav aadressil: <http://www.adobe.com/ee/products/illustrator.html>

väikesemahuline ja võimaldab lehe laadimiskiirust vähendada, siis on tänapäeval SVG kasutamine just eriti laialdane.

Kuna autor käsitleb töös Adobe Photoshopi põhjalikumalt, siis tutvustatakse seda tarkvara eraldi peatükis (vt ptk 2).

1.2 Eeskomponendi arendusprotsess

Eeskomponendi arendaja (ingl *front end developer*) kasutab veebidisaineri poolt koostatud disainifaili ning selle põhjal loob kliendipoolse rakenduse veebilehest, ehk selle, mida kasutaja näeb ja mille kaudu suhtleb rakendusega. Eeskomponendi arendaja vajalike oskuste tase sõltub suuresti projektidest, mille kallal ta töötab. Väiksemamahuliste veebilehtede loomise puhul piisab enamasti ainult HTML ja CSS keelte oskusest, kuid keerukamate ja mahukamate tööde puhul tuleb tänapäeva veebiarenduse puhul kasutusele võtta mitmeid vahendeid, mis võimaldavad ja kiirendavad projekti loomist.

Eeskomponendi arendusprotsess algab disainifailist piltide väljalõikamisega (ingl *slice*) ja nende paigutamiseiga veebilehele. Komponendid (nt lingid, tekstid, tabelid, vormid) paigutatakse lehele kasutades HTML ja CSS koodi. Hüperteksti-märgistuskeel (ingl *Hyper Text Markup Language*) ehk HTML on enimlevinud kodeerimissüsteem veebidokumentide loomiseks. HTML märgendid määravad ära selle, kuidas veebileht arvutiekraanil välja näeb (Vallaste, HTML (HyperText Markup Language), kuupäev puudub). Kaskaadlaadistik ehk CSS (ingl *Cascading Style Sheets*) on veebilehtede valmistajatele ja kasutajatele mõeldud laadistik. Laadileht (ingl *style sheets*) koosneb reeglitest, mis teatavad brauserile, kuidas dokumenti kuvada. Andes laadilehel elementidele väärtusi, saab näiteks ära määrata kirjatüüpe, värve, suurusi (Vallaste, CSS (1) (Cascading Style Sheets), kuupäev puudub). Elementide interaktiivsust, näiteks animeerimist, tekkimist või kadumist, määratakse enamasti JavaScript koodi kirjutades. JavaScript suudab suhelda HTML keeles kirjutatud lähtekoodiga ja võimaldab muuta veebilehed dünaamiliseks (Vallaste, JavaScript, kuupäev puudub). Juhul kui kasutatakse mõnda sisuhaldustarkvara (ingl *content management system*, lühend *CMS*), siis tuleb sellele paralleelselt teha ka liidestust.

HTML'i, CSS'i ja JavaScripti ning ka teiste programmeerimiskeelte kirjutamise lihtsustamiseks kasutavad eeskomponendi arendajad spetsiaalseid, kohandatavaid

tekstiredaktoreid. Tuntuimad neist on näiteks SublimeText3¹⁶, Notepad++¹⁷ ja Atom¹⁸. Mõningad tekstiredaktorid võimaldavad lisada pistikprogramme (ingl *plug-in*), mis veel omakorda koodi kirjutamist lihtsustavad. Enimkasutatavate pistikprogrammide seas on HTML'i kirjutamise puhul näiteks Emmet¹⁹ ja JavaScripti kodeerimise puhul JSHint²⁰.

Suuremamahuliste projektide korral võidakse kasutada HTML'i, CSS'i ja JavaScripti koodi kirjutamiseks eelprotsessoreid (ingl *preprocessor*), mille puhul on tegemist tarkvaraga, mis töötleb toorandmed põhiprogrammile sobivale kujule. Toorandmed on HTML'i ja CSS'i ja JavaScripti eelprotsessorite puhul enamasti lihtsustatud kood, mis võimaldab kasutada näiteks muutujaid, tehteid, funktsioone või ära jätta sümboleid. Eelprotsessorite kasutamine muudab koodikirjutamise kiiremaks ja kompaktsemaks, ühtlasi võib teatud puhkudel genereeritav valmiskood olla pisendatud (ingl *minified*), mis muudab lehelaadimise kiiremaks. Enimkasutatavate eelprotsessorite seas on HTML'i puhul näiteks Haml²¹, CSS'i puhul Sass²² ja JavaScripti puhul CoffeeScript²³.



```
$darkblue: #00008B;
$border-color: #CCCCCC; //gray
$base-font-size: 16px;

body {
  font-size: $base-font-size;
}
header {
  border-bottom: 1px solid $border-color;
}
a {
  color: $darkblue;
}

body {
  font-size: 16px;
}
header {
  border-bottom: 1px solid #CCCCCC;
}
a {
  color: #00008B;
}
```

Joonis 1. Eelprotsessor Sass'i kood (vasakul) ja sellest kompileeritud CSS kood (paremal)²⁴

Eelprotsessorieid suudavad töödelda tegumijooksutajad (ingl *task runner*) mis ongi eelkõige mõeldud erinevate tööde automatiseerimiseks. Arendaja saab tegumijooksutajas näiteks määrata Sass failide minimeerimise (ingl *minify*), projektifailide kokkuliitmise (ingl *concatenate*) ja brauseri eesliidete lisamise. Tuntuimad sellised tööriistad on Gulp²⁵ ja Grunt²⁶.

¹⁶ Sublime Text 3 on kättesaadav aadressil: <https://www.sublimetext.com/3>

¹⁷ Notepad++ on kättesaadav aadressil: <https://notepad-plus-plus.org/>

¹⁸ Atom on kättesaadav aadressil: <https://atom.io/>

¹⁹ Emmet on kättesaadav aadressil: <http://emmet.io/>

²⁰ JSHint on kättesaadav aadressil: <http://jshint.com/>

²¹ Haml on kättesaadav aadressil: <http://haml.info/>

²² Sass on kättesaadav aadressil: <http://sass-lang.com/>

²³ CoffeeScript on kättesaadav aadressil: <http://coffeescript.org/>

²⁴ Autor Christina Truong, koodinäited võetud aadressilt: <http://blogs.adobe.com/dreamweaver/2015/02/getting-started-with-css-preprocessors-less-and-sass.html>

²⁵ Gulp on kättesaadav aadressil: <http://gulpjs.com/>

²⁶ Grunt on kättesaadav aadressil: <http://gruntjs.com/>

Kuna iga uus alustatav projekt on üldjoontes samade komponentide ja struktuuriga, siis ka projektifailide loomiseks kasutatakse automatiseeritud vahendeid. Näiteks saab luua omale sobivate failide struktuuri ja sisuga trafareti (ingl *boilerplate*) ja seda iga uue projekti loomisel kasutada. Sarnast funktsiooni täidab ka Yeoman²⁷, mis lisaks uute projektifailide tekitamisele käivitab ka tegumijooksutaja ja paketi halduri (ingl *package manager*).

Vastavalt nõuetele võib eeskomponendi arendajal veebilehe loomise juures vaja minna mudel-vaade-kontrolleri, ehk *Model–View–Controller* (lühendatud MVC) kasutamist. Tegemist on kliendipoolse raamistikuga, mis võimaldab andmeid sõltumatult kuvada ning hoiab eeskomponendi serveripoolsest eraldi. Tuntuimad sellised JavaScripti raamistikud on näiteks Angular²⁸, Backbone²⁹ ja React³⁰.

Kõikide projektide puhul on kasulik ka versioonihalduse kasutamine, mis veebilehe puhul võimaldab erinevate versioonide haldamise, kui nende kallal töötab paralleelselt mitu inimest. Sellisel juhul on võimalik kõigil projekti kallal töötavatel osapooltel saada projektist viimase versiooni täielik koopia, seda muuta ja üles laadida ning vajadusel ka naasta vana versiooni juurde. Tuntuimad sellised töövahendid on näiteks Git³¹ ja SVN³².

Veebilehe eeskomponendi loomise vältel kasutatakse selle vaatamiseks, testimiseks ja silumiseks (ingl *debug*) enamjaolt veebibrauserit. Tänapäeva veebibrauseritega tulevad kaasa veebiarendust lihtsustavad tööriistad, näiteks Google Chrome’il kannab see nimetust Developer Tools³³, mille puhul koonduvad tööriistapaketi alla näiteks (Google, kuupäev puudub):

- elementide paneel (ingl *elements panel*), kust näeb kogu veebilehe struktureeritud informatsiooni;
- konsool (ingl *console panel*), kuhu saab logida informatsiooni, et vastavalt sellele rakendust siluda ning sisestada koodikäske;
- võrgu paneel (ingl *network panel*), kust näeb elementide lehele laadimise kiirust ja päringute silumiseks.

²⁷ Yeoman on kättesaadav aadressil: <http://yeoman.io/>

²⁸ Angular on kättesaadav aadressil: <https://angularjs.org/>

²⁹ Backbone on kättesaadav aadressil: <http://backbonejs.org/>

³⁰ React on kättesaadav aadressil: <https://facebook.github.io/react/>

³¹ Git on kättesaadav aadressil: <https://github.com/>

³² SVN on kättesaadav aadressil: <https://subversion.apache.org/>

³³ Developer Tools info aadressil: <https://developer.chrome.com/devtools>

Lisaks võimaldavad brauserid lisada erinevaid pistikprogramme omakorda täiendavate ülesannete lahendamiseks. Näiteks JSON viewer³⁴, mille abiga saab json faili mugavamalt vaadata või HTML5 Outliner³⁵, mis loob HTML *heading* (päise) ja *section* (sektsiooni) elementidest sisukorra.

³⁴ JSON viewer Chrome'ile on kättesaadav aadressil: <https://chrome.google.com/webstore/detail/json-viewer/gbmdgpbipfallnflgajpaliibnhdgobh>

³⁵ HTML5 Outliner Chrome'ile on kättesaadav aadressil: <https://chrome.google.com/webstore/detail/html5-outliner/afoibpobokebhgfknfndkgemglggomo>

2 Adobe Photoshopi tutvustus

Adobe Photoshop on üks tänapäeva tähtsamaid graafikatöötlusprogramme. Photoshop võimaldab hõlpsat piltide redigeerimist kuid on samas üsna keerukas tarkvara, mille rohketes paneelides, menüüdes ja tööriistades võib tavalisel arvutikasutajal olla väga raske orienteeruda (Bauer, 2013). Järgnevalt antakse ülevaade Photoshopi ajaloost ning Adobe Photoshop CC tarkvarast, kui Photoshopi viimasest versioonist ja antud töös kasutatavast programmist.

Adobe Photoshopile pani 80-ndate aastate lõpus aluse tollane doktorantuuri tudeng Thomas Knoll, kui kirjutas oma diplomitööd, milles keskendus digitaalpiltide töötlemisele. Töö arenedes asus Knoll 1987. aastal oma Mac operatsioonisüsteemiga arvutile arendama pilditöötlusprogrammi. Esimese rakenduse loomise järel kaasas Thomas arendusse ka oma venna Johni, kelle disainioskused tarkvara loomisel abiks olid. Aastaks 1988 oli loodav tarkvara juba rohkete funktsionaalsustega ning see sai ka oma nime – Photoshop. Seejärel otsustasid vennad lõpetada beetaversiooni arendusega ning püüdsid tarkvara maha müüa. Leitigi soovija kelleks oli BarneyScan ettevõtte, mis müüs 200 litsentseeritud tarkvara koopiat koos enda toodetavate skänneritega. Ei läinud kaua, kuni Adobe avastas Photoshopi potentsiaali, ning 1989. aasta aprillis osteti vendadelt tarkvara ära (Schewe, 2000). Adobe tõi turule oma esimese versiooni Photoshopist 1990. aasta veebruaris ja versiooninumbriks sai 1.0 ning see jooksis Mac operatsioonisüsteemiga arvutitel. Järgneva kümne aastaga müüdi Photoshop tarkvarast 3 miljonit koopiat (Shustek, 2013). Praeguseks on tarkvarast välja tulnud 21. versioon Adobe Photoshop CC 2015. Photoshopist on saanud üks tuntumaid tarkvarasid maailmas ning sellest on tulnud isegi tegusõna – photoshoppima (ingl *to photoshop*), ehk Photoshop tarkvaraga pilti redigeerima. Photoshopiga loodud või muudetud graafikat on võimalik leida pea kõikjalt.

Adobe Photoshop CC on 2013. aastal välja lastud Photoshopi kõige uuem versioon, millest on välja tulnud ka juba CC 2014 ja CC 2015 versioonid. Tarkvara nimetuses kasutatav lühend CC viitab Adobe Creative Cloud tarkvarapaketele, mille alla koonduvad kõik Adobe'i loomevahendid. Adobe Photoshopi on võimalik soetada erinevate pakettide läbi, aga ka eraldi programmina. Näiteks, üksikrakendusena soetades tuleb tarkvara eest välja käia 23,99 eurot kuus, samas Photography paketina on kuumakse 11,99 eurot ning paketti kuulub ka Lightroom CC (Adobe, kuupäev puudub). Oluliseima uuendusena on uude Photoshopi loodud montaažilaudade (ingl *artboard*) kasutamine. See annab võimaluse töötada korraga paljude stsenaariumitega korraga ühes failis. Montaažilauda saab luua juba mallina sisestatud mõõtude

järgi, aga ka valides vajalikud kihid (ingl *layers*) ning moodustades neist uus montaažilaud (Wimberly, 2015). Teise uuendusena on Photoshoppi eelvaatena lisatud Adobe'i poolt veel väljatöötamises olev Design Space keskkond, mis võimaldab lihtsustada tänapäeva veebi-, kasutajakogemuse- ja rakendusedisainerite tööd, kuna võimaldab optimeerida tavapäraseid disainitöö ülesandeid nagu produktiivsem arvutihiire kasutamine ja erinevate paneelide ja dialoogide haldamine. Design Space keskkonnast saab vajadusel tagasi suunduda Photoshoppi graafika redigeerimise juurde (Clark, 2015). Lisaks tuleb Photoshopiga kaasa Adobe Preview CC tööriist, millega saab erinevates iOS seadmetes Preview CC iOS³⁶ app'i abil kuvada oma rakenduse või veebidisaini eelvaadet. Kasuliku vahendina on Photoshopi külge ühendatud Creative Cloud rakendus, kust saab otsida paljude mallide ja pistikprogrammide seast endale sobivad ning paigaldada need. Ühtlasi hoiab Creative Cloud rakendus kogu tarkvarapaketi rakendused uuendatuna.

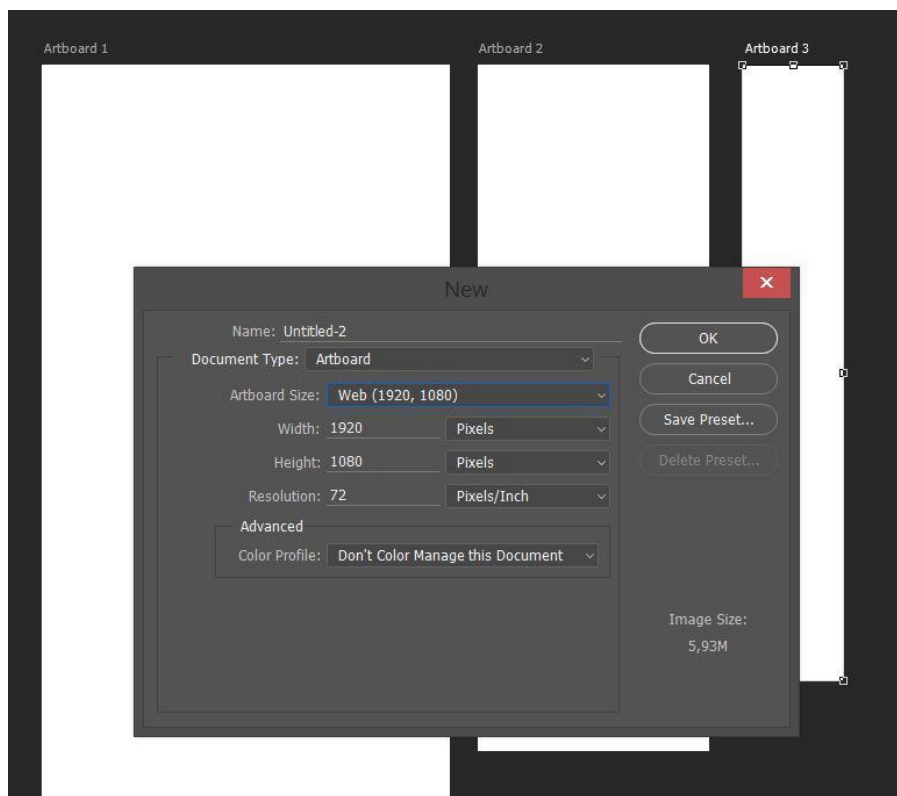
³⁶ Preview CC iOS app on kättesaadav aadressil: <https://itunes.apple.com/us/app/adobe-preview-cc/id973272286?ls=1&mt=8>

3 Adobe Photoshopi kasutamine veebilehe loomisel

Kuigi Photoshop oli algselt mõeldud digitaalfotode redigeerimiseks, on sellest praeguseks välja kasvanud väga laialdaste võimalustega tarkvara. Olenemata asjaolust, et leidub programme, mis on mõeldud otseselt veebiarenduseks, ehk kus graafikat ja lähtekoodi saab korraga redigeerida, siis kasutatakse veebilehe loomiseks ka Photoshopi.

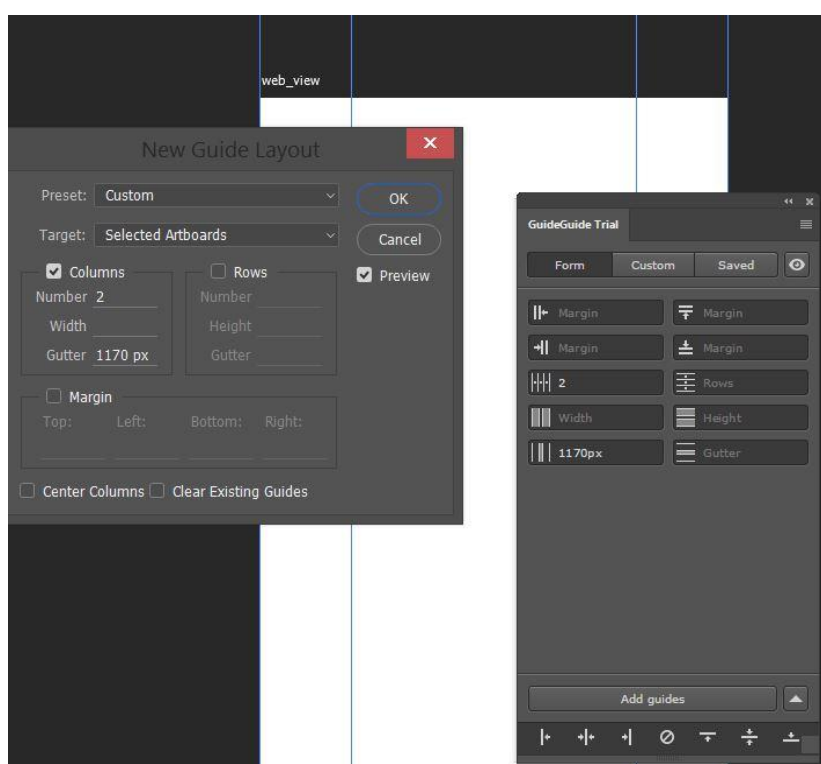
3.1 Veebidisaini koostamine

Alustades Photoshopis uue projektiga, pannakse esialgu paika dokumendi sätted (vt Joonis 2). Kui veebilehte luuakse arvutiekraanile, on dokumendi laiuks soovitatav määrata 1920 pikslit, kuna see võimaldab veebilehel hiljem full HD pilte kuvada. Hiljem määrab eeskomponendi arendaja maksimaalseks lehe sisuosa laiuks 1920 pikslit. Dokumendi kõrguse määramine pole oluline, kuna see muutub jooksvalt vastavalt sellele, kui palju sisuelemente lehele tuleb. Olulise uuendusena on Photoshop CC 2015 versioonis kasutusele võetud dokumenditüübi määramine, kust saab valida Web formaadi, millele hiljem montaažilaudu lisada. Sellisel viisil on ühes dokumendis erinevatele montaažilaudadele võimalik disainida erinevaid lehekülje- või ekraanivaateid.



Joonis 2. Uue dokumendi loomine Adobe Photoshop CC 2015 programmis

Kui dokumendi seaded on määratud siis kuvab Photoshop tühja lehe. Esimese sammuna tasuks siinkohal ära määrata sisuala laius, mille saab ära märkida vertikaalsete abijoontega (ingl *guides*). Näiteks Bootstrap'i Grid süsteemi kasutades on arvutiekraanile loodava veebilehe sisuala soovitatavaks laiuks määrata 1170 pikslit, mis jätab 1200 pikslitele ekraanidele vajalikud 15 pikslit mõlemalt poolt tühjaks. Järgides edasi Bootstrap'i Grid süsteemi, siis on soovitatav ära määrata ka 12 sisuveergu. Photoshop CC 2015 versioonis on olemas New Guide Layout tööriist, mis võimaldab mugavalt sisestada dokumenti etteantud parameetritele vastavad abijooned. Vanemates versioonides on selleks laialt kasutatav GuideGuide³⁷ pistikprogramm.



Joonis 3. New Guide Layout tööriist vasakul ja GuidesGuides pistikprogramm paremal

Veebidisainiga töötamiseks on kasulik kohandada Photoshopi tööruum. Selleks, et disainiga mugav tegeleda oleks, peaksid tööruumi olema sisestatud vajaminevad aknad. Olulisemad on Layers (kihtide), Info (teabe) ja Character (fondi) aknad, vaja võib minna ka Color (värvivaliku), History (ajaloo) ja Navigator (navigaatori) aknaid. Samuti võiks nähtaval olla tööakna äärtes asuvad joonlauad (ingl *rulers*) ning mõõtesüsteemiks tuleks kindlasti määrata pikslid. Autor koostas lähtuvalt soovitustest tööruumi faili mis on avalikult kättesaadav³⁸.

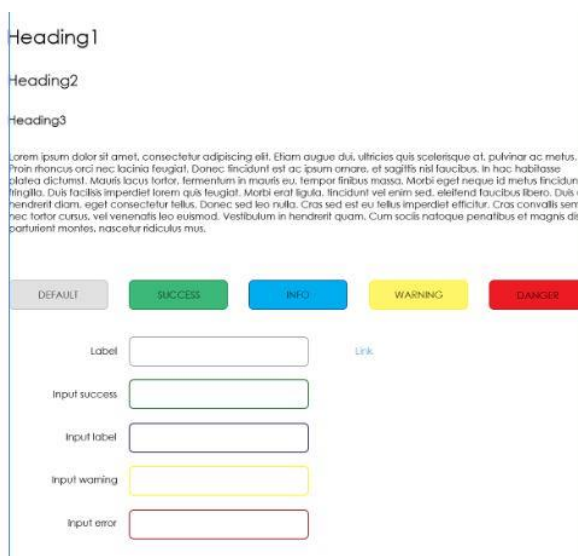
³⁷ GuideGuide pistikprogramm on kättesaadaval aadressil: <http://guideguide.me/>

³⁸ Tööruumi fail on kättesaadav aadressil: <http://www.tlu.ee/~paul657/workspace/>

Tarkvaraga pidevalt töötades on soovitatav töö kiirendamiseks ära õppida enim vajaminevad klahvikombinatsioonid, mille abil on paljud tööriistad ja tegevused kättesaadavad. Disainiprotsessi puhul on enim vajaminevad klahvikombinatsioonid:

- V klahvi alt kättesaadav liigutamise tööriist (ingl *move tool*) elementide valimiseks ja liigutamiseks;
- M klahvi alt kättesaadav märkimise tööriist (ingl *marquee tool*) mitme elemendi valimiseks korraga;
- H klahvi alt kättesaadav või tühiku klahvi all hoides kasutatav käe tööriist (ingl *hand tool*) tööruumis liikumiseks;
- Ctrl/Cmd+ C valiku kopeerimiseks;
- Ctrl/Cmd + V valiku kleepimiseks;
- Ctrl/Cmd + D valiku tühistamiseks;
- Ctrl/Cmd + S dokumendi salvestamiseks.

Kui tööruum kohandatud, siis algab loominguiline tegevus. Disainer asub koostama loodava veebilehe komponentide lehte (vt Joonis 4). Vastavalt paika pandud värvipaletile stiliseeritakse ära erinevad veebilehel esinevad elemendid nagu kirjatüübid, nupud, vormiväljad, tabelid.

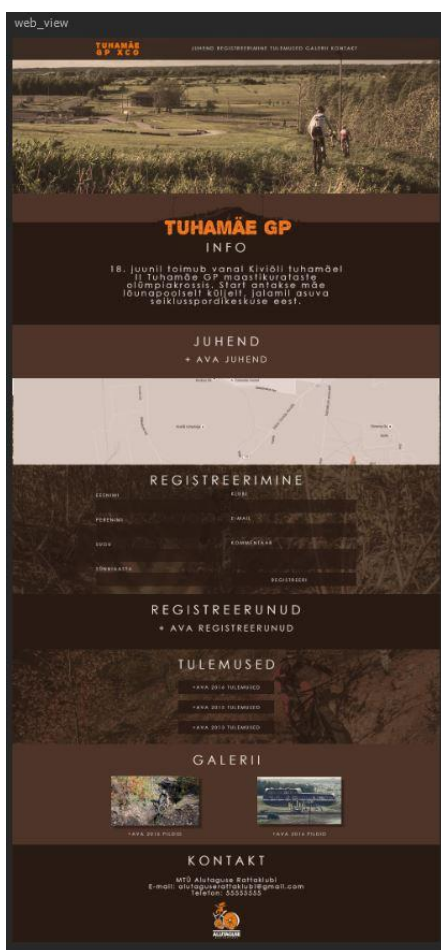


Joonis 4. Komponentide lehe näide

Photoshopile on saadaval suurel hulgal erinevaid pistikprogramme, mis võimaldavad täita täiesti erinevaid ülesandeid. Leidub ka mitmeid disainerite tööd produktiivsemaks muutvaid pistikprogramme. Kirjatüüpide otsimiseks ja projekti sisestamiseks on olemas Layerhero poolt

loodud Font Hero³⁹ pistikprogramm. Olenevalt loodavast disainist, võib disaineril vaja minna ikoonide kasutamist, selle lihtsustamiseks on olemas FlatIcon⁴⁰ pistikprogramm. Samuti on olemas rohkesti programme erinevate graafiliste tööde lihtsustamiseks, nagu värvipalettide genereerijad ja pintsli (ingl *brush*) kogumikke.

Komponentide lehe loomise järel võetakse sõrestikmudel ning vastavalt sellele hakatakse looma lehekomponente. Esimese sammuna paigutatakse lehe päise (ingl header) ja jaluse (ingl footer) osad, kuna neid kasutatakse igal järgnevalt loodaval lehel. Seejärel loodakse sisuleht, mis on toorikuks igale järgnevalt loodavale lehele, ehk siin kuvatakse baaselemendid, nagu taustapilt/värv, päis, jalus või menüüriba. Kui sisuleht on valminud, asutakse looma vastavalt sõrestikmudelitele erinevaid lehti, esialgu avaleht ning siis alamlehed. Vastavalt vajadusele näidatakse disainifaili vaheversioone kliendile ja kooskõlastatakse need.

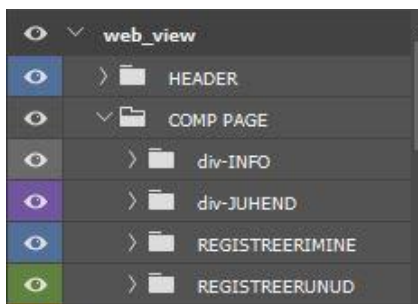


Joonis 5. Valminud veebidisaini näide

³⁹ Font Hero on kättesaadav aadressil: <http://www.layerhero.com/fonthero/>

⁴⁰ Flaticon pistikprogramm on kättesaadav aadressil: <http://www.flaticon.com/apps/adobe-extension>

Väga oluline on kogu loomingulise tegevuse juures hoida tekitatavad elemendid struktureeritult vastavatel kihtidel ja gruppides ning vajadusel kihtidele värvid määrata (vt Joonis 6).



Joonis 6. Kihtide struktureerimise ja värvimise näide

See aitab hiljem disaineril endal, aga ka teistel töövoos osalevatel osapooltel loodud disainifailis navigeerida ning kihte hallata. Seda eriti eeskomponendi arendajal, kellel on vaja täpselt teada, millised elemendid omavahel koos käivad ja millised mitte. Kihtidega töötamist aitab lihtsustada Layers Control ²⁴¹ pistikprogramm, mis sobib suuremahulistes projektides kihte haldamiseks. Programm laseb mugavalt kihte ümber nimetada ja liita (ingl *flatten*) ning optimeerida projekti kustutades tühjad kihid ja kasutamata efektid (20 Best Photoshop Plugins For Designers, 2016).

Disainer võib oma töös kasutada ka endale meelepärast ja projektiga sobivat disainimalli (ingl *template*), ehk ettevalmistatud ja stiliseeritud komponentidest koosnevat disainifaili. Sellist võimalust pakub näiteks UI8⁴² keskkond, kust on võimalik paljude erinevate mallide seast osta Photoshopi failivormis sobilik disainipakett ja kasutada sealt vajalikke komponente ja vajaduse korral neid muuta. Samuti on olemas pistikprogramme, mis võimaldavad Photoshopi siseselt valida erinevaid disainimalle. Erinevate komponentide ja mallide disainikavandile sisestamiseks pakub mugavat võimalust WebZap⁴³ pistikprogramm.

Photoshopi kasutamisega käib kaasas ka puudusi, millega disainerid arvestama peavad. Näiteks pole võimalik Photoshopi abil edasi anda elementide animeerimist. Samuti on piiratud vektorgraafika loomine ja muutmine, selleks on soovitatav pigem kasutada Adobe Illustratori programmi. Photoshop võimaldab küll SVG faili rastergraafikast eksportida, kuid see on üsna tülikas ning väheste valikuvõimalustega.

⁴¹ Layers Control 2 on kättesaadav aadressil: <http://madebyvadim.com/layers/>

⁴² UI8 keskkond aadressil: <https://ui8.net/>

⁴³ WebZap on kättesaadav aadressil: <http://webzap.uiparade.com/>

3.2 Veebi eeskomponendi arendamine

Eeskomponendi arendus hõlmab endas enamasti koodiridade kirjutamist ja rakenduse testimist. Juhul, kui veebidisainer on disainifaili loonud Photoshopi tarkvara kasutades, siis on ka eeskomponendi arendajal tööprotsessi algusfaasis vaja nimetatud programmi käsitseda.

Sarnaselt disainerile, on ka eeskomponendi arendajal kasulik seadistada töölaud. Kui seda juba enne tehtud ei ole, siis tuleks kindlasti mõttesüsteemiks määrata pikselmõõdustik. Paneeliakendest peaks kindlasti nähtaval olema Layers (kihtide) aken, olulisel kohal on ka Character (fondi) aken ja Info (info) aken, vajalikuks võib osutada Navigation (navigatsiooni) aken, kui on tarvis suuremahulises dokumendis tegutseda.

Seejärel võib arendaja asuda disaini väljalõikamise kallale. Lõiketööriista (ingl *slice tool*) kasutades lõigatakse välja pildid, ning eksporditakse need. Pilte saab genereerida ka käsuga *image assets*. Disaifailist mõõdetakse elementide suurused ja värvid, lehele paigutatakse HTML elemendid, mida paralleelselt ka stiliseeritakse. CSS koodis määratakse ära üldised elemendiklasside stiilid. Juhul kui kasutatakse muutujaid võimaldavat CSS eelprotsessorit, näiteks Sassi, siis võetakse disainifaili komponentide lehelt kõik värvid ja suurused ning pannakse muutujatesse, et neid hiljem mugav kasutada oleks ja vajadusel kõike korruga muuta saaks. Icoonidest genereeritakse *sprite*'id (vt Joonis 7), mida tänu CSS3 tehnoloogiale veebilehel kasutada saab. See meetod kiirendab oluliselt ikoonide kuvamist lehel, kuna ikoonileht loetakse ühe laadimisega sisse ja üksikud ikoonid lõigatakse CSS koodi poolt välja.



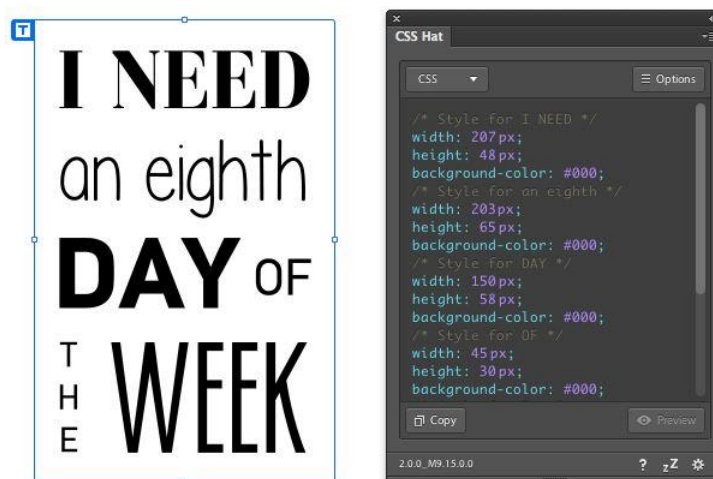
Joonis 7. Sprite sheet pildifail erinevate sotsiaalmeedia ikoonide näitel⁴⁴

Seejärel asutakse erinevate komponentigruppide loomise juurde. Esimese sammuna luuakse tavaliselt lehekülje päis ja jalus, kuna enamjaolt on need veebisaidil läbivalt samad. Seejärel luuakse sisuleht, ehk baaskomponentidega leht, kuhu seatakse taustavärv või taustapilt. Lehele

⁴⁴ Autor wysiwyg, joonise pilt võetud aadressilt: <http://www.wysiwyg.co.il/Social-Media-CSS-social-icons-sprite.asp>

paigutatakse ka päis ja jalus ning olemasolul ka menüüd. Sisuleht võetakse toorikuks iga järgneva lehe koostamisel. Tihti luuakse sisulehele ka disainifailis oleva komponentide lehe elemendid (nt lingid, nupud, tabelid), et neid edaspidiselt projekti alamlehtede loomisel läbivalt kasutada saaks. Kui sisuleht on valminud, asub eeskomponendi arendaja looma järgimiseid lehti, millest tavaliselt esimesena avaleht ning siis järgmised vaated. Arendustöö vahetulemusi testitakse ja lastakse ka kliendil testida. Testimiseks on erinevaid võimalusi, näiteks laetakse projektifailid kliendi veebimajutusse üles, või on töö teostajal oma keskkond kasutajapoolse hinnangu saamiseks. Kui kasutatakse sisuhaldussüsteemi, siis toimub ka kohe liidestamine ning projekt võidakse kliendile osadena edastada, et klient saaks hakata sisu haldama.

Photoshopi CC 2015 versioonis on mitmeid tööriistu, mis lihtsustavad eeskomponendi arendaja tööd. Näiteks saab küsida valitud elemendi CSS'i koodi. See tuleb eriti kasuks näiteks mitmikastmete (ingl *gradient*), varjude ja läbipaistvuse genereerimisel. Siiski pole antud tööriist kõige täpsem ning enamasti kasutatakse ka selleks tööks pistikprogrammi. Tuntuimate seas on CSS Hat ⁴⁵ (vt Joonis 8), mis on CSS koodi genereerimisel oluliselt täpsem.



Joonis 8. CSS Hat 2 näide: teksti elementidest loodud CSS⁴⁶

HTML koodi genereerimiseks Photoshopis sisseehitatud töövahendit pole, kuid Creative Cloud keskkonnast on võimalik leida pistikprogramm Webbsy⁴⁷, mis võimaldab nii HTML kui ka CSS koodi genereerimist.

⁴⁵ CSS Hat 2 on kättesaadav aadressil: <https://csshat.com/>

⁴⁶ Autor CSS Hat 2, joonise pilt võetud aadressilt: <https://csshat.com>

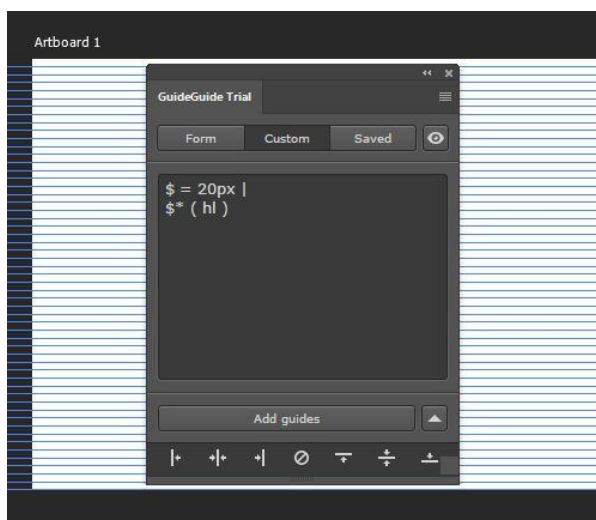
⁴⁷ Webbsy on pistikprogramm kättesaadav aadressil: <http://www.webbsy.com/>

Töörühmaga disainifaili jagamiseks on loodud mugav pistikprogramm Avocode⁴⁸, mis võimaldab töös olevat disainifaili üles laadida Avocode'i keskkonda. Seal saab määrata töörühma osalised ning seejärel muudetud disainifaili jooksvalt sünkroniseerida keskkonnas oleva failiga, et teised osapooled saaksid lihtsamalt vaadata loodavat disainifaili ning jätta kommentaare.

3.3 Hinnang tulemustele

Töö raames proovis autor läbi erinevad töövahendid ning pistikprogrammid. Luues näidiseena veebidisaini, sai autor katsetada erinevaid tööriistu. Peaesmärgiks võttis autor uurida võimalusi Photoshopi disainifailist koostada ise lähtekoodi kirjutamata töötav veebileht.

Esimese pistikprogrammina uuris autor GuideGuide vahendit, mis osutus üsna mugavaks. Programm võimaldab sisestada vajalikud parameetrid ning vastavalt nendele loob töölauale abijooned. Samuti võimaldab programm salvestada parameetrid mallina ning hiljem uuesti sama seadistusega abijooni sisestada. Ühtlasi on võimalik sisse anda parameetrid koodina ja vastavalt sellele abijooned joonistada (vt Joonis 9). Samas on Photoshop CC 2015 versioonis olemas New Guide Layout tööriist, mis on küll väiksemate võimalustega, kuid üsna sarnane vahend.



Joonis 9. GuideGuide pistikprogrammi käskude näide

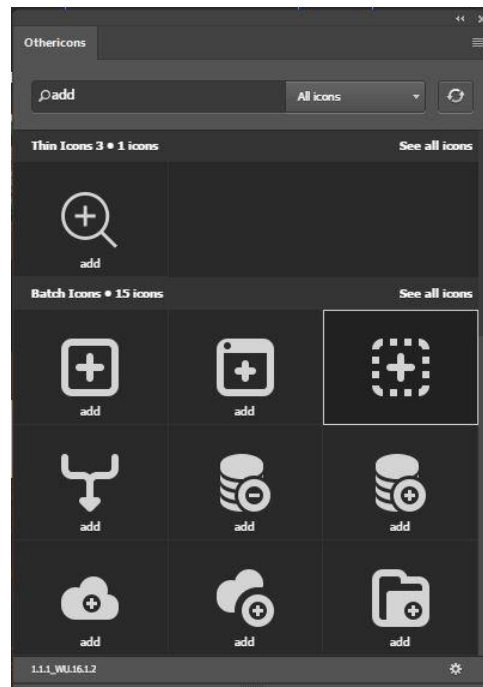
Järgmisena testis autor disainiprotsessi käigus Pictura⁴⁹ ja Othericons⁵⁰ pistikprogramme, millest esimene, Pictura, võimaldab projekti sisestada pilte, mida saab sisestatava otsisõnaga

⁴⁸ Avocode Photosop Plugin on kättesaadav aadressil: <https://madebysource.com/download/#/all>

⁴⁹ Pictura pistikprogramm on kättesaadav aadressil: <http://pictura.madebysource.com/>

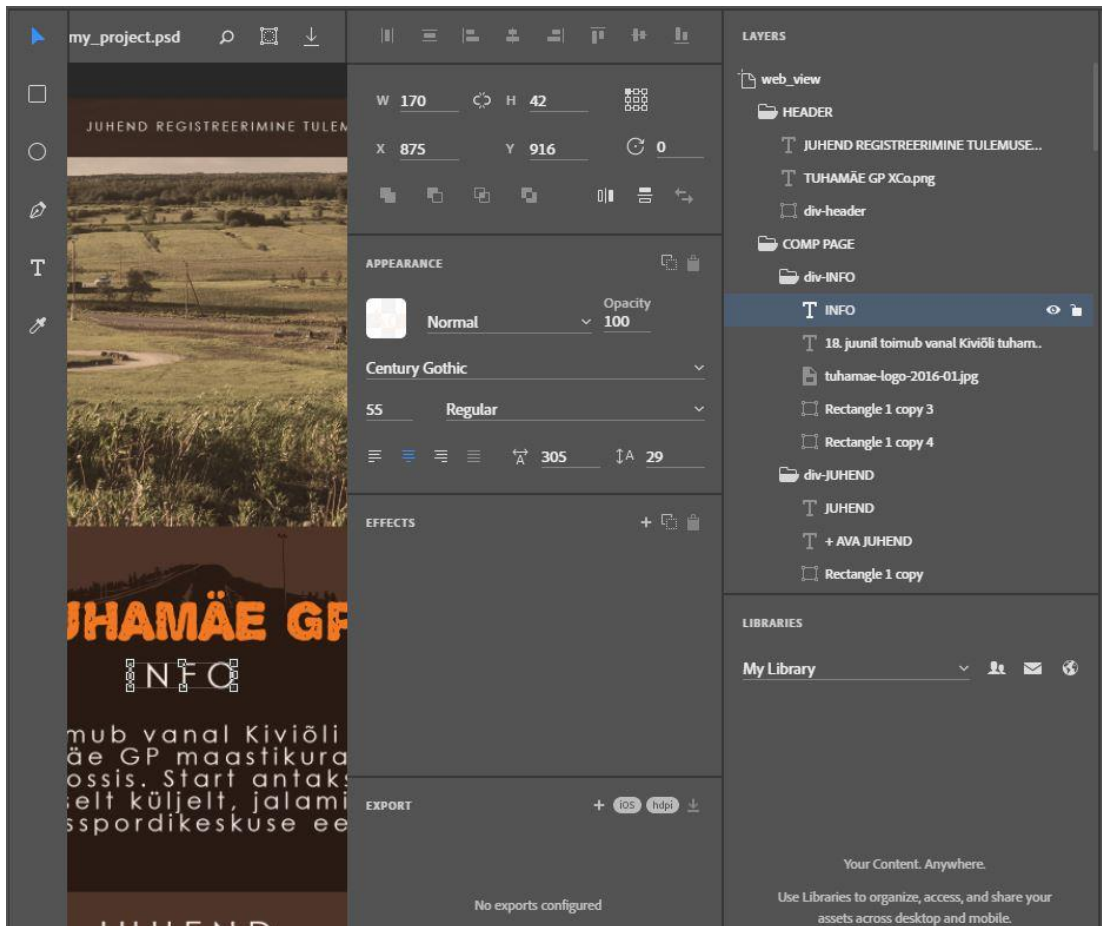
⁵⁰ Othericons pistikprogramm on kättesaadav aadressil: <http://othericons.madebysource.com/>

otsida. Sarnase funktsionaalsusega programm Othericons (vt Joonis 10) võimaldab vastavalt otsisõnale leida sobiliku ikooni. Mõlemad pistikprogrammid osutusid üsna mugavaks, kuna andsid kiireid ja vajadustele vastavaid tulemusi. Pictura laseb pilte filtreerida vastavalt litsentsile, Othericons ikoonid on tasuta kättesaadaval.



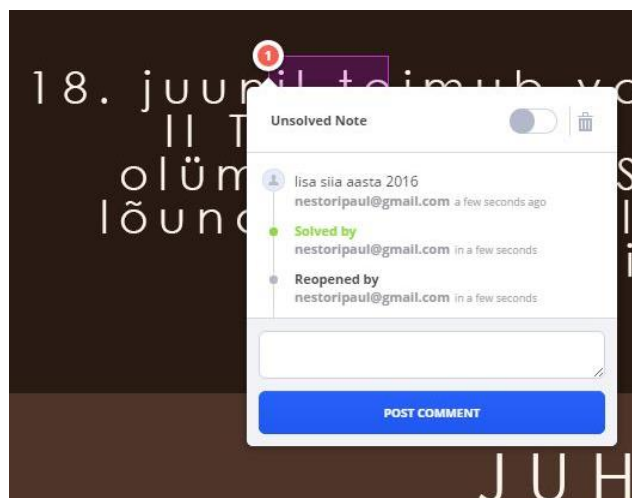
Joonis 10. Othericons pistikprogrammi aken

Design Space disainikeskkonnas (vt Joonis 11) töötamine osutus autori jaoks üllatavalt mugavaks. Kuna keskkond on loodud spetsiaalselt disainitööks, siis on tööriistaribale jäetud alles ainult vajaminevad tööriistad nagu Move tool, Rectangle ja Ellipse tool, Pen tool, Type tool ning Sample tool. Kohandatud on ka paneeliakende riba, kus välja toodud kõik vajaminev. Lisaks sellele on kõik kasutajaliidese elemendid suurelt ja selgelt välja toodud. Autori seisukohalt võiks Design Space keskkonda kasutada nii veebidisainerid, kui ka eeskomponendi arendajad.



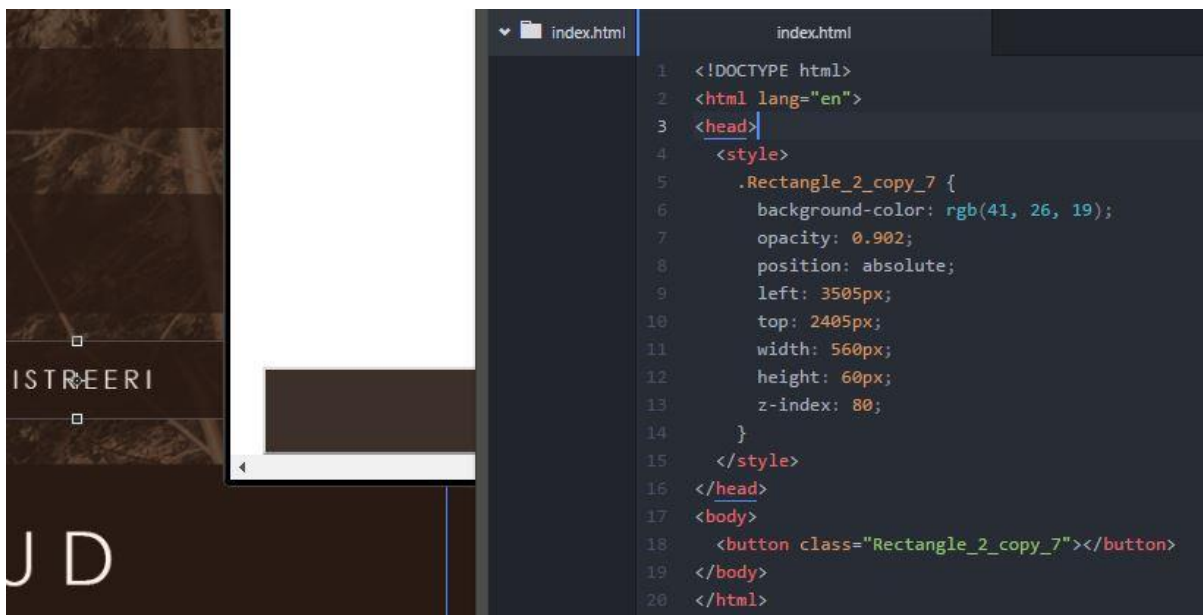
Joonis 11. Design Space keskkonna vaade

Avocode pistikprogramm võimaldab otse Photoshopi programmiaknast Avocode keskkonda üles laadida loodavat disainifaili ning seda jagada. Avocode keskkonnas on võimalik kavandile märkmeid teha, kommenteerida neid ja määrata märkmeid lahendatuks (vt Joonis 12). Autori hinnangul on tegemist kasuliku vahendiga, kui disaini juures töötab mitu töörühma liiget.



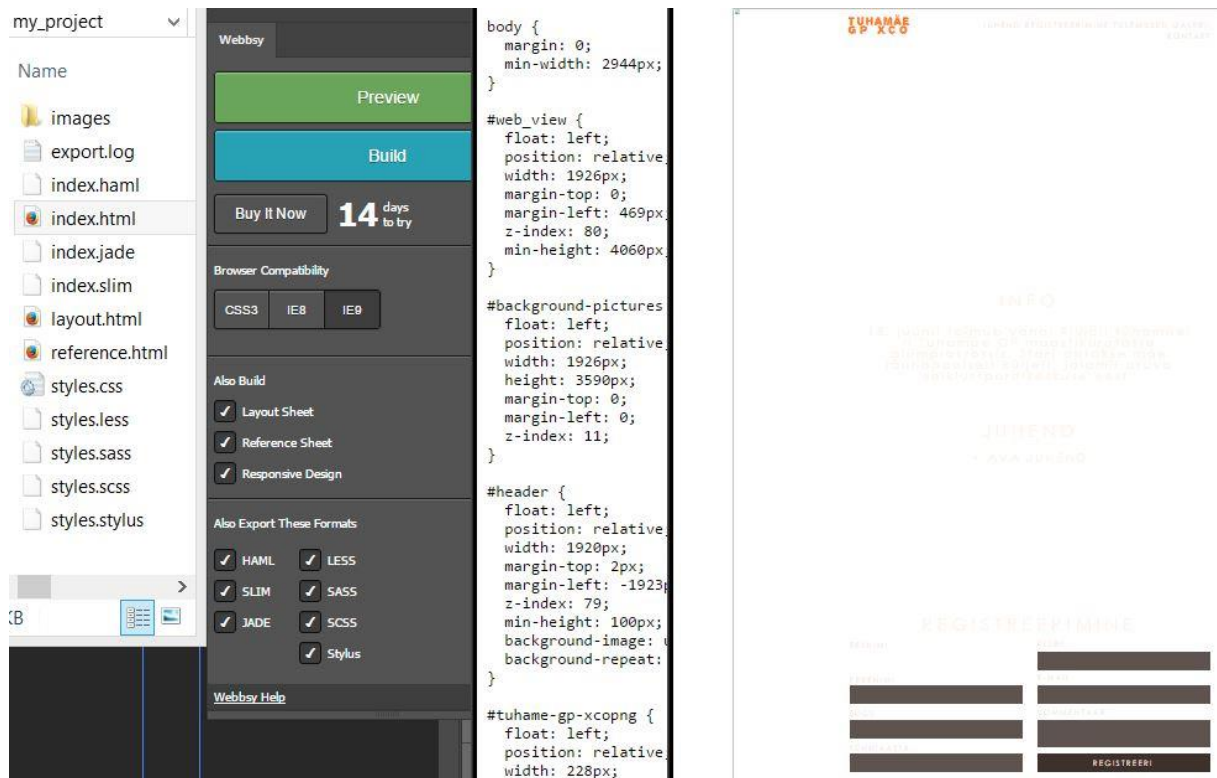
Joonis 12. Avocode keskkonnas disainikavandile märkme lisamine

Järgmisena proovis autor CSS koodi kopeerimist. Selleks tuli valida element ning Layer menüü alt valida käsk Copy CSS või teha Layers paneelil valitud elemendil vajutada paremat hiireklahvi ning sealt valida käsk Copy CSS. Autor leidis, et tegemist on küllaltki kasuliku vahendiga. Kuigi element asetati pikslite järgi lehel sinna, kus ta ka disainifailis oli, siis värvid ja suurused tundusid olevat vastavad Photoshopis loodud elemendi omadega (vt Joonis 13). Küll aga soovitatakse täpsuse huvides kasutada selleks tööks pistikprogrammi CSS Hat 2, mida autoril polnud kahjuks võimalik testida, kuna tegemist on tasulise tarkvaraga millel puudub ka prooviversioon.



Joonis 13. Copy CSS käsu näide disainifailis oleva nupu põhjal. Joonisel näha Photoshopis loodud ristkülikut (vasakul), brauseris kuvatavat ristkülikut (keskel) ning genereeritud koodi (paremal, style elemendi sees)

Webbsy pistikprogramm on mõeldud Photoshopi veebidisaini failist HTML ja CSS koodi konverteerimiseks. Programmis on olemas Preview ja Build käsud, millest esimene loob lihtsama ülevaate lehest, ning teine käsk genereerib põhjalikuma veebilehe projekti. Samuti loob erinevate HTML (Haml, Slim, Jade) ja CSS (Less, Sass, Scss ja Stylus) eelprotsessorite failid (vt Joonis 14).

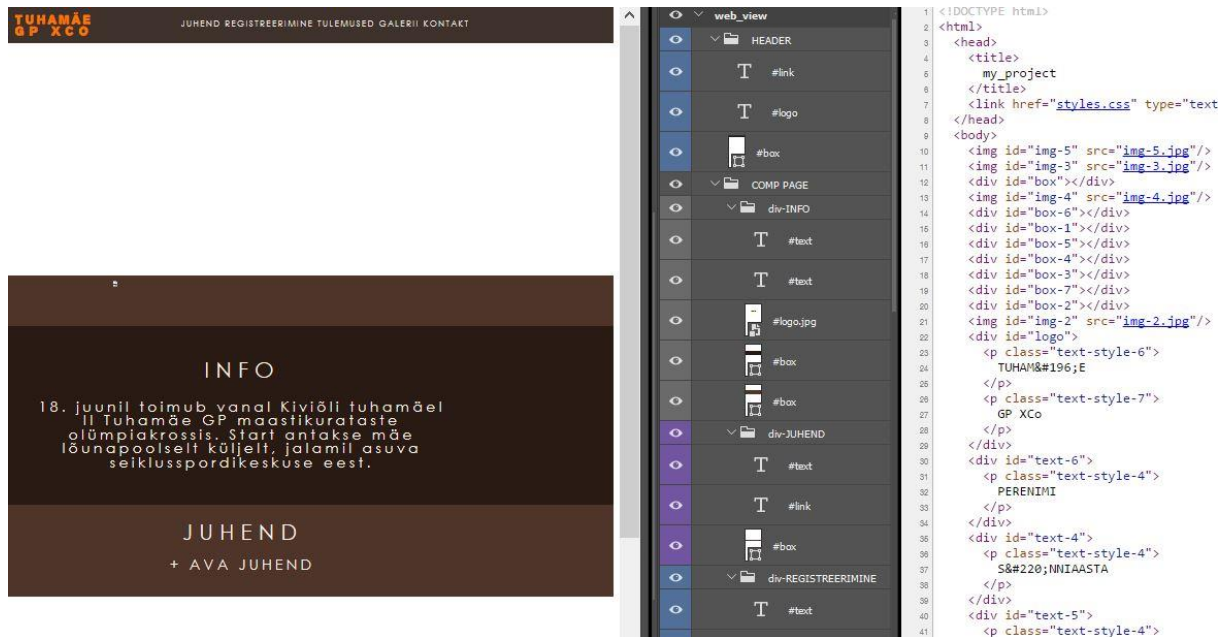


Joonis 14. Webbsy pistikprogrammi Preview käsu näide. Joonisel näha alates vasakult genereeritud faile, Webbsy programmi kasutajaliidest, genereeritud CSS koodi ja genereeritud veebilehte brauseris

Testides Preview käsku, ei tulnud selleks teha disainifailis ühtegi muudatust, kuna kavand lõigati piltideks ning määrati asukoht CSS koodiga. Siiski nägi tulemus välja väga ebatäpne ning puudusid paljud elemendid.

Build käsku kasutades tuli eelnevalt elementikihtide nimedele ette kirjutada „#“ märk. Lisaks määras autor ära elementide tüübid (nt tekst, link, pilt). Genereeritud leht oli visuaalselt üsna sarnane disainifailiga, kuid puudusid pildid. Samuti oli tulemus väga suuremõõtmeline ning puudus lehe kohandatavus, mida programmiaknas sai määrata genereerimiseks. W3C validaatorist⁵¹ HTML koodi kontrollides näidati veateatena *img* märgendil *alt* atribuudi puudumist. Samas oli HTML kood hästi loetav, kuigi elementide loomiseks kasutati ainult *div* ja *p* märgendeid (vt Joonis 15).

⁵¹ W3C HTML validaator kättesaadav aadressil: <https://validator.w3.org>



Joonis 15. Webbsy pistikprogrammi Build käsu näide. Joonisel näha genereeritud veebilehte (vasakul), kihtide nimetusi (keskel) ja genereeritud HTML koodi (paremal)

Testitud vahendid osutusid üsna kasulikuks. Autorile meeldisid enim Photoshopis olev Design Space keskkond ning Avocode pistikprogramm, mida ta ilmselt ka edaspidi kasutama hakkab. Suurimat huvi pakkus Webbsy pistikprogramm, kuid kahjuks oodatavaid tulemusi see vahend ei andnud. Samuti avastas autor, et pistikprogrammide paigaldamine võib olla keerukas. Tihti tuli programmi installeerimisele erinevalt läheneda, mõningatel juhtudel tuli end keskkonda kasutajaks registreerida. Kohati lakkas pistikprogramm töötamast või ei käivitunud üldse. Paljud huvipakkuvad pistikprogrammid on tasulised ning neil puudub prooviversioon, mistõttu ei saanud autor neid käsitleda.

Kokkuvõte

Veebilehe loomise protsess on tänapäeval võrdlemisi keerukas. Kuigi leidub töövahendeid, mis oluliselt protsessi lihtsustavad, siis tihti ei võimalda need luua rätseplahendusi vastavalt kliendi soovidele. Mahukamate tööde puhul on soovitatav protsess jagada mitmeks osaks ning määrata nendega tegelema erinevad töörühma liikmed, kes on vastavalt spetsialiseerunud. Seetõttu on veebilehe visuaalse poole loomiseks enamasti eraldi nii veebidisainer kui ka eeskomponendi arendaja. Olulise lülina kahe arendaja vahel on graafikaprogramm, mis võimaldab disaineril oma loomingut edastada eeskomponendi arendajale, kes loob disainifailist veebilehe kliendipoolse rakenduse. Laialdaselt kasutatavaks vahendiks selles töös on tõusnud Adobe Photoshop graafikaprogramm.

Käesoleva bakalaureusetöö eesmärgiks oli anda ülevaade veebilehe disaini ja eeskomponendi loomise töövoost ning põhjalikumalt kirjeldada, kuidas seda tehakse Adobe Photoshop tarkvara abil.

Bakalaureusetöö käigus tutvustati tänapäeva veebilehe loomise protsessi ning uuriti lähemalt veebidisaini ja eeskomponendi arendusmeetodeid. Ühtlasi tutvustati erinevaid tehnoloogiaid ja töövahendeid. Seejärel anti ülevaade Adobe Photoshop CC tarkvarast. Töö põhiosaks kujunes veebilehe loomise uurimine Photoshop tarkvara abil, mis hõlmas endas disainiprotsessi ja eeskomponendi arendamise tööriistu ning meetodikaid. Lisaks anti hinnang olulisematele töövahenditele ja lihtsamini kättesaadavatele pistikprogrammidele.

Bakalaureusetöö käigus selgus, et Adobe Photoshop CC 2015 on väga mahukas programm, kuhu on liidestatud ka mitmed veebidisaineri ja eeskomponendi arendaja tööd lihtsustavaid vahendeid. Lisaks on saadaval väga suur hulk erinevaid malle, kogumikke ja pistikprogramme loome- ja arendusprotsessi produktiivsemaks muutmiseks.

Tehtud uurimustöö andis töö autorile uusi teadmisi ja oskusi Photoshopist veebilehe loomise protsessis. Saadud teadmised on kasulikud tööturul, kus disaineritele ja eeskomponendi arendajatele on tihti määratud kohustuslikuks oskuseks Photoshopi kasutamine. Lisaks sai autor hulgaliselt teadmisi veebi eeskomponendi arendamise tehnoloogiast tänapäeval.

Kasutatud kirjandus

20 Best Photoshop Plugins For Designers. (6. aprill 2016. a.). Loetud aadressil

<http://beebom.com/best-photoshop-plugins/>

Adobe. (kuupäev puudub). *Discover the Creative Cloud experience*. Loetud aadressil

https://creative.adobe.com/plans?single_app=photoshop&promoid=5JRH8K43&mv=other

Arel, E. (18. 04 2016. a.). *Requirements Analysis – a Critical First Step in Web Site Design*.

Loetud aadressil <http://www.ayantek.com/requirements-analysis-critical-first-step-web-site-design/>

Arguin, M., Heald, G., & Team, A. C. (2013). Starting up. rmt: M. Arguin, G. Heald, & A. C. Team, *Dreamweaver CC Digital Classroom* (lk 1). Hoboken: John Wiley & Sons.

awwwards-team. (20. veebruar 2016. a.). *What are Frameworks? 22 Best Responsive CSS*

Frameworks for Web Design. Loetud aadressil <http://www.awwwards.com/what-are-frameworks-22-best-responsive-css-frameworks-for-web-design.html>

Bauer, P. (2013). *Photoshop CC For Dummies*. Somerset: For Dummies.

Clark, P. (15. juuni 2015. a.). *Adobe Launches the 2015 Release of Photoshop CC – with Major New Features for Designers (including Artboards)*. Loetud aadressil

<http://blogs.adobe.com/photoshop/2015/06/adobe-launches-the-2015-release-of-photoshop-cc.html>

Google. (kuupäev puudub). *Chrome DevTools Overview*. Loetud aadressil

<https://developer.chrome.com/devtools>

Requirements analysis. (kuupäev puudub). Loetud aadressil

<https://www.aoe.com/en/services/requirements-analysis.html>

Rinde, A. (kuupäev puudub). Loetud aadressil

http://www.cs.tlu.ee/~rinde/www_materjal/veebi_elutsykkel.pdf

Schewe, J. (veebruar 2000. a.). Photoshop Profile Thomas John Knoll 10. *PEI Magazine*.

Shustek, L. (2013). *Adobe Photoshop Source Code*. Loetud aadressil

<http://www.computerhistory.org/atcm/adobe-photoshop-source-code/>

Vallaste, H. (kuupäev puudub). *CSS (1) (Cascading Style Sheets)*. Loetud aadressil e-teatmik: <http://www.vallaste.ee/index.asp>

Vallaste, H. (kuupäev puudub). *HTML (HyperText Markup Language)*. Loetud aadressil e-teatmik: <http://www.vallaste.ee/index.asp>

Vallaste, H. (kuupäev puudub). *JavaScript*. Loetud aadressil e-teatmik: <http://www.vallaste.ee/index.asp>

Wimberly, C. (6. Juuli 2015. a.). *What Photoshop's CC 2015 update means for screen designers*. Loetud aadressil <http://www.creativebloq.com/photoshop/what-cc2015-means-screen-designers-71515605>

Summary

The process of creating a webpage these days is rather difficult. Although there are tools that make this process easier, they don't necessarily provide tailor-made solutions according to the client's wishes. It is reasonable to divide bigger works into smaller tasks and assign them to different suitably qualified members of the workgroup. For this reason both a web designer and a front-end developer are involved in creating the visual part of a webpage. An important link between the two developers is a graphics program that gives the designer the opportunity to forward his creations to the front-end developer, who turns the design file into the webpage's client-side application. The graphics program Adobe Photoshop has become a widely used tool in this part of the process.

The aim of this thesis is to give an overview of the workflow of creating a webpage's design and the front-end component, also to describe in more detail how it is done using the Adobe Photoshop software.

This thesis described the creating of a modern webpage and examined the development methods of web design and the front-end component. Different technologies and tools were described as well. After that an overview of the software Adobe Photoshop CC was given. The main part of the thesis was the study of how webpages are created using the Photoshop software. This included the tools and methods of the design process and the front-end development. In addition some of the more important tools and easily accessible plug-in programs were evaluated.

In writing this thesis it was revealed that Adobe Photoshop CC 2015 is a very substantial program, into which a lot of tools are interfaced that make the work of a web designer and the front-end developer easier. There is also a large amount of templates, sets and plug-in programs available to make the creative development process more productive.

This thesis provided the author with a lot of new knowledge and skills on how Photoshop is used in the process of creating a web page. The acquired knowledge is very useful in the job market, where designers and front-end developers are required to know how to use Photoshop. In addition the author of this thesis gained a lot of knowledge on the technologies of the development of the front-end component.