

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond
Tarkvarateaduse instituut

IT40LT
Hendrik Laas 143064IAPB

TTÜ MOODLE'I KASUTAJALIIDES: ANALÜÜS JA ARENDUS

Bakalaureusetöö

Juhendaja: Ago Luberg
MSc

Tallinn 2017

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Hendrik Laas

22.05.2017

Annotatsioon

Selle bakalaureusetöö eesmärgiks on luua õpiplatvormile Moodle Tallinna Tehnikaülikooli temaatiline ja Moodle'i uuendustele vastuvõtlik kasutajaliides.

Töö tulemusena valmib Moodle'i kasutajaliides, mis võimaldab kasutada kogu Moodle'i funktsionaalsust, on kasutatav erinevate suurustega ekraanidel ning toetab populaarsemate veebibrauserite uusimaid versioone.

Töö analüüsi osas uuritakse üldtunnustatud tavasid kasutajaliidese loomiseks, analüüsitakse eelmise kasutajaliidese kohta tehtud küsitlust ning pannakse kirja nõuded. Baasliidese valiku peatükis selgitatakse, millist liidest laiendama hakatakse ning miks selline valik tehti. Töö arenduse osas antakse ülevaade kasutatud tehnoloogiatest, selgitatakse liidese lähtekoodi struktuuri ja kirjeldatakse lisatud funktsionaalsust.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 35 leheküljel, 5 peatükki, 13 joonist, 2 tabelit.

Abstract

User interface of TUT Moodle: analysis and development

The main aim of this thesis is to develop an user interface for Moodle that fits the style of Tallinn University of Technology and is responsive to Moodle updates.

The developed user interface of Moodle enables user to use the whole functionality of Moodle, is usable on devices with varying screen sizes and in newest versions of popular web browsers.

In the analysis different aspects of good user interface design and quiz about previous interface are analyzed and requirements are set. In the next part the selection of extendable theme is explained. In development part main used technologies are described, structure of the source code is explained and added functionality is presented.

The thesis is in Estonian and contains 35 pages of text, 5 chapters, 13 figures, 2 tables.

Lühendite ja mõistete sõnastik

CSS	<i>Cascading Style Sheets</i> , kaskaadlaadistik
HITSA	Hariduse Infotehnoloogia Sihtasutus
HTML	<i>Hypertext Markup Language</i> , hüperteksti märgistuskeel
Moodle	<i>Modular Object Oriented Dynamic Learning Environment</i> , Modulaarne Objektorienteeritud Dünaamiline Õpikeskkond
PHP	<i>PHP: Hypertext Preprocessor</i> , PHP: Hüperteksti Eeltöötleja
SASS	<i>Syntactically Awesome Stylesheets</i> , Sünaktiliselt Vinge Stiililehe keel
SCSS	<i>Syntax of SASS</i> , SASS-i süntaks
TTÜ	<i>Tallinn University of Technology</i> , Tallinna Tehnikaülikool
URL	<i>Uniform resource locator</i> , universaalne ressursilokaator
XML	<i>Extensible Markup Language</i> , laiendatav märgistuskeel

Sisukord

1 Sissejuhatus	10
1.1 Eesmärgid ja oodatav tulemus	10
1.2 Metoodika	10
2 Analüüs	12
2.1 Üldtunnustatud tavad kasutajaliidese disainile	12
2.2 Küsitluse analüüs	13
2.2.1 Välimus ja disain	14
2.2.2 Kasutusmugavus	14
2.2.3 Funktsionaalsus	14
2.3 Nõuded	15
2.3.1 Funktsionaalsed nõuded	15
2.3.2 Mittefunktsionaalsed nõuded	15
3 Baasteema valik	17
3.1 Kahe sobivaima baasteema valik	17
3.2 Boosti ja Snapi võrdlus	17
4 Arendus	19
4.1 Moodle	19
4.1.1 Moodle'i teema	19
4.1.2 Moodle'i teema loomine Boosti laiendades	20
4.1.3 Moodle teema config.php ja version.php failide sisud	21
4.2 Mirage	22
4.2.1 Kohanduv disain	22
4.2.2 SCSS	24
4.2.3 JavaScriptiga lisatud funktsionaalsus	25
4.2.4 PHP-ga lisatud funktsionaalsus	26
4.3 Testimine erinevatel brauseritel	31
4.4 Edasiarendused	32
5 Kokkuvõte	34
Kasutatud kirjandus	35

Lisa 1 – Pealeht	36
Lisa 2 – Kursuse vaade.....	37
Lisa 3 - Töölaua vaade.....	38
Lisa 4 – Seadete vaade.....	39
Lisa 5 – Mobiilivaated.....	40

Jooniste loetelu

Joonis 1. Boosti põhjal loodava teema struktuur.....	20
Joonis 2. ained.ttu.ee favicon.ico ja pealkiri.....	20
Joonis 3. Näide teema config.php failist.	21
Joonis 4. Näide teema version.php failist.	22
Joonis 5. Kursuse vaade akendes laiuslega 1920 pikslit ja 1440 pikslit.....	23
Joonis 6. Kursuse vaade nutitefonis laiuslega 360 pikslit.	24
Joonis 7. Näide renderer.php laiendamisest.	27
Joonis 8. Navigatsiooniribale lisatud hinnetelege ja kalendri nupud.	28
Joonis 9. Muutmise keelamise ja ploki lisamise nupp kursuse vaate päises.	28
Joonis 10. Administraatorimenüü lehe paremas servas.....	29
Joonis 11. Lähenevad tähtjad ja hiljutine tagasiside pealehel.	30
Joonis 12. Sektsiooni lohistamise ja muutmise nupud	30
Joonis 13. Kursuse õpetajate loetelu ühel real.....	31

Tabelite loetelu

Tabel 1. Boosti ja Snapi võrdlus.....	18
Tabel 2. Erinevate vaadete töötavus erinevatel veebilehitsejatel	31

1 Sissejuhatus

Tallinna Tehnikaülikool kasutab veebipõhise õpikeskkonnana peamiselt Moodle'it. Enamik ülikoolist kasutab küll välist Moodle'i pakkujat (HITSA Moodle), kuid Infotehnoloogia teaduskond on üle läinud oma Moodle'ile (ained.ttu.ee). Praegu on probleemiks see, et TTÜ oma Moodle'i kasutajaliides ei võimalda kasutada kogu Moodle'i funktsionaalsust (näiteks sõnumid). Antud töö ülesandeks on Moodle'ile töötava ning TTÜ-le kohandatud kasutajaliidese loomine.

1.1 Eesmärgid ja oodatav tulemus

Töö esimeseks eesmärgiks on analüüsida praegu kasutuses oleva liidese puudusi, uurida millised on põhiaspektid hea kasutajaliidese loomiseks ning kogutud informatsiooni põhjal uue kasutajaliidese loomine, millelt on likvideeritud olemasoleva liidese puudused.

Töö teiseks eesmärgiks on arenduse läbiviimine niiviisi, et valmiva kasutajaliidese lähtekood oleks arusaadav ning vajadusel oleks seda lihtne edasi arendada ning sinna funktsionaalsust lisada. Arendatav kasutajaliides peab olema vastuvõtlik Moodle'i uuendustele.

Oodatava tulemusena valmib kasutajaliides, mis on TTÜ temaatiline nagu praegu kasutatavgi. Tulemuse lähtekood on dokumenteeritud, et vajadusel oleks muutuste elluviimine võimalikult lihtne. Kasutajaliides on Moodle'i uuendustele vastuvõtlik ning järgib Moodle'i nõudeid kasutajaliidese arendamiseks.

1.2 Metoodika

Töö eesmärkideni jõudmiseks analüüsitakse kõigepealt 2016. aasta sügissemestri alguses läbiviidud küsitluse tulemusi praegu kasutatava kasutajaliidese kohta. Küsitlus võimaldab välja selgitada, millised on praeguse liidese puudused ning milliste osadega kasutajad

rahul on. Enne kasutajaliidese arendamist uuritakse üldtunnustatud nõudeid kasutajaliidese arendamisele ning rakendatakse uuritud uue kasutajaliidese arendamisel.

Kasutajaliides arendatakse Moodle'i teemana (Moodle *Theme*), laiendades olemasolevat Moodle'i teemat. Töös selgitatakse välja, milline teema võtta baasteemaks. Arendamise käigus kasutatakse nelja põhilist tehnoloogiat: HTML, JavaScript, SASS ja PHP. Arendamise lihtsustamiseks kasutatakse raamistikke jQuery¹ ja Bootstrap². HTML-i mallide loomisel kasutatakse Mustache³ mallisüsteemi.

¹ <https://jquery.com/>

² <http://getbootstrap.com/>

³ <https://mustache.github.io/mustache.5.html>

2 Analüüs

Selles peatükis uuritakse üldtunnustatud tavaid kasutajaliideste disainile, analüüsitakse küsitluse tulemusi praegu kasutuses oleva Moodle'i teema kohta ja pannakse paika nõuded arendatavale kasutajaliidesele.

2.1 Üldtunnustatud tavad kasutajaliidese disainile

Kasutajaliidese põhieesmärk on võimaldada kasutajal soovitud tegevusi võimalikult tõhusalt ja arusaadavalt ellu viia. Et see võimalik oleks, lähtutakse kuuest peamisest põhimõttest: kasutajat tunde ja järjekindel disain, arusaadavus, kõrge vastavus kasutaja tegevusele, hooldatavus ja uuendatavus, andestatavus ning atraktiivsus. [1], [3]

Kasutajat tunde ja järjekindel disain

Kasutajaliidese juures on kõige tähtsam, et see oleks lihtsasti kasutatav. Seega peavad kõik kasutajaliidese osad olema disainitud järjepidevalt ning kasutajale arusaadaval moel. Tänapäeval kasutatakse palju erinevaid liideseid ning kasutajad on nendega harjunud. Et kasutajad end mugavalt tunneksid, tuleb arvestada olemasolevate liidestega, mitte luua täiesti uut kujundust. [1]

Arusaadavus

Kõik kasutajaliidese komponendid peavad olema disainitud nii, et nende tähendus on üheti mõistetav [2]. Kuldreegel on, et mida kiiremini ja lihtsamini seletatav komponent on, ilma, et see kaotaks oma tähendust, seda paremini komponent disainitud on [1]. Kui kasutajad ei saa aru, kuidas kasutajaliides toimib, siis nad satuvad segadusse, ärrituvad ning ei pruugi enam kunagi rakendust kasutada [3].

Kõrge vastavus kasutaja tegevusele

Kasutajakogemus halveneb, kui kasutajaliides ei suuda sammu pidada kasutaja poolt antud sisenditega, nagu hiireklakk või klaviatuuri nupule vajutus. Kui vähegi võimalik, siis peab kasutajaliides püsima tempos kasutajaga, kuid kui see võimalik ei ole, siis on soovitatav kasutajale interaktiivsete komponentide (laadimismärk, aktiivselt uuendatav

tekst) abil teada anda, et lehte laaditakse. Aeglaselt töötav kasutajaliides võib jätta mulje puudulikkust või vigasest tarkvarast, kuigi tarkvara võib olla kvaliteetne. [1]

Kasutajaliides peab andma kasutajale tagasisidet selle kohta, mis rakenduses toimub. Näiteks kui vajutatakse nupule, siis teatatakse kasutajale, et vajutus oli edukas muutes nupu mitteaktiivseks ning kuvades kasutajale infot kasutajaliidese laadimise kohta. [3]

Hooldatavus ja uuendatavus

Üks tähtsamaid aspekte kvaliteetse kasutajaliidese loomise juures on selle vastuvõtlikkus uuendustele. Siin kohal on tähtis, et uue funktsionaalsuse lisamisel, ei tekiks olemasoleva funktsionaalsusega konflikte. Tuleb mõelda sellele, et ka uute komponentide jaoks oleks ruumi. [1]

Andestatavus

Keegi ei ole eksimatu ning kasutajaliidest kasutades tehakse tahes-tahtmata vigu. Seega peaks kasutajaliides olema andestav vigade suhtes, näiteks kui kustutatakse mõni tähtis osa informatsioonist, siis peab olema võimalik kustutamise tegevus tühistada. Kui navigeeritakse lehele, mis ei tööta, peab kasutajale kuvama abistavat lehte, kus on informatsioon selle kohta, mis võis minna valesti, mitte krüpteeritud veakoodi. [3]

Atraktiivsus

On soovituslik, et kasutajaliides oleks atraktiivne, kuid seejuures ei tohi kaotada teisi olulisi eelmistes punktides kirjeldatud aspekte [3]. Tuleb arvestada et atraktiivsus on suhteline ning sõltub sihtgrupist, kellele kasutajaliides orienteeritud on. Atraktiivsus muudab kasutamise meeldivaks, kuid siin kohal tuleb silmas pidada, et atraktiivsus ei muuda puudulikku kasutajaliidest paremaks, vaid annab ainult lisaväärtust [1] .

2.2 Küsitluse analüüs

Analüüsitav küsitlus viidi läbi 2016. aasta sügissemestri alguses vahemikus 29.09.2016 – 13.10.2016. Küsitluse koostasid ja viisid läbi TTÜ bakalaureuse tudengid Joosep Franz Moorits Alviste ja Brenda Uga. Küsitluses osales kokku 102 tudengit, kellest 99 õppisid küsitluse läbiviimise ajal Infotehnoloogia teaduskonda kuuluvat eriala (informaatika,

äriinfotehnoloogia, küberkaitse) ehk puutusid praegu kasutuses oleva kasutajaliidese igapäevaselt kokku keskkonnas ained.ttu.ee.

Vastanute käest küsiti, kuidas nad üldiselt keskkonnaga rahul on, kuidas nad on rahul välimusega, kasutusmugavusega ja funktsionaalsusega. Iga aspekti kohta küsiti rahuolu viie palli skaalal ning lisaks küsiti, mida võiks muuta paremaks, kus oli vastanutel võimalus anda ka vabatekstiline vastus.

2.2.1 Välimus ja disain

Vastanutest kokku 87.3% hindasid välimust hindegaga 4 või 5 (vastavalt 40.2% ja 47.1%), 9.8% vastanutest hindegaga 3 ning madalamalt hindasid välimust 3% vastanutest. Võib järeldada, et välimusega ollakse rahul, mistõttu võetakse töös aluseks praegu kasutuses oleva liidese välimus.

Kuna enamjaolt ollakse välimusega rahul, siis võetakse arvesse soovitusi, mida on võimalik täita ilma, et välimuse põhilised komponendid muutuksid. Töös arvestatakse vastanute sooviga muuta välimust mobiilisõbralikumaks ning samuti arvestatakse sooviga muuta mõne vaate välimust (näiteks failide üleslaadimine).

2.2.2 Kasutusmugavus

Vastanutest kokku 73.6% hindas kasutusmugavust hindegaga 4 või 5 (vastavalt 47.1% ja 26.5%), 16.3% hindegaga 3 ning kokku 8.9% hindegaga 1 või 2. Võib järeldada, et kasutajamugavust on võimalik parandada, sest 26,4% kasutajatest ei ole kasutusmugavusega rahul.

Küsitletutest 43.1% leidis, et vajalikud komponendid võiksid kergemini leitavad olla ning 41.2% leidis, et igapäevased toimingud, nagu hinnete vaatamine ja materjalide leidmine, võiks olla lihtsam. Suur osa soovitustest on seotud lehel navigeerimise lihtsustamisega. Võib järeldada, et kasutusmugavust on võimalik parandada lehel navigeerimist lihtsustades.

2.2.3 Funktsionaalsus

Vastanutest kokku 85.3% hindas funktsionaalsust hindegaga 4 või 5 (vastavalt 49% ja 36.3%), 10.8% hindegaga 3 ning 3% hindegaga 1 või 2. Võib järeldada, et üldiselt ollakse funktsionaalsusega rahul, kuid seda vaid arvutil kasutades.

Küsitluses osalenutest 43.1% leiab, et kasutajaliides võiks olla mobiilisõbralikum. Teine funktsionaalsuse puudumine, mis vastanutele ei meeldinud, on oma töölaua (dashboard) muutmise võimaluse puudumine. Ka funktsionaalsuse juures leidis 28.8% tudengitest, et navigeerimine võiks olla lihtsam ning võimalik väiksema arvu klikkidega.

2.3 Nõuded

Järgnevalt esitatakse loodavale teemale funktsionaalsed ja mittefunktsionaalsed nõuded.

2.3.1 Funktsionaalsed nõuded

1. Teema võimaldab kasutada kogu Moodle'i funktsionaalsust.
2. Pealehel kuvatakse kursuste otsingukasti.
3. Kui kasutaja on sisse logitud kuvatakse pealehel lähenevaid tähtaegasid ning hiljutist tagasisidet.
4. Kui kasutaja on sisse logitud, kuvatakse pealehel kursusi, kuhu kasutaja on registreeritud.
5. Kursuse lehel kuvatakse kursuse sisuga seotud info vasakul menüüs, kõik muu paremal.
6. Administraatorimenüü on kättesaadav iga lehe paremas ääres.
7. Teated, sõnumid, kalender ja hinded on kättesaadavad lehe navigatsiooniribal ühe klikiga.

2.3.2 Mittefunktsionaalsed nõuded

1. Teema kood on inglisekeelne ning kõik funktsioonid on kommenteeritud ning arusaadavad.
2. Teema on tõlgitud nii eesti kui ka inglise keelde, uue keelefaili lisamisel on teema võimalik tõlkida mistahes Moodle'i poolt võimaldatavasse keelde.
3. Teema disain peab olema veebibrauseri akna suurusega kohanduv (*responsive*).
4. Teema on kasutatav kõikidel platvormidel – nii arvutil, tahvelarvutil kui ka nutitelefonil.
5. Teema töötab kõikide enamkasutatavate veebilehitsejate (Google Chrome, Safari, Mozilla Firefox, Microsoft Edge, Opera) uutes versioonides.
6. Teema järgib Moodle'i standardeid.

7. Teema on vastuvõtlik Moodle'i uuendustele.
8. Arendatav teema laiendab mitte ei kirjuta üle baasteema faile.
9. Teema järgib peatükis 2.1 kirjeldatud põhimõtteid.
10. Teema on Tallinna Tehnikaülikooli temaatiline.
11. Teema loomisel arvestatakse peatükis 2.2 analüüsitud küsitluse tulemusi.

3 Baasteema valik

Baasteema (Moodle *theme*) on olemasolev Moodle'i kasutajaliidese teema, mida laiendatakse antud töös loodava kasutajaliidese teema arendamisel. Selles peatükis selgitatakse baasteema valikut. Selgitatakse, miks kitsendati valikut vaid kahele teemale ning võrreldakse neid.

3.1 Kahe sobivaima baasteema valik

Baasteema valikul võeti aluseks kaks teemat:

1. Boost – Moodle'i arendajate poolt loodud teema.
2. Snap – praegu kasutuses oleva teema alus.

Valiti just need teemad, sest Boost on Moodle'i arendajate poolt loodud teema, mis toetab nii Bootstrap'i kui ka jQuery't. Boost on Moodle'i uusimal versioonil *default* (vaikimisi) teemaks. Teised Moodle'i arendajate poolt loodud teemad toetavad Bootstrap'i vanemaid versioone ning neid ei ole seetõttu mõistlik aluseks võtta.

Lisaks soovivat Moodle kasutada just Boosti modernsete kasutajaliideste loomiseks [4]. Võrdluseks valiti Snap, sest praegu kasutuses olev teema (ained.ttu.ee-s kasutatav) on ülesehitatud Snap'i koodi muutes.

3.2 Boosti ja Snap'i võrdlus

Tabel 1 esitab Boosti ja Snap'i võrdluse. Tabelis võrreldakse erinevaid aspekte, mis võetakse aluseks baasteema valimisel.

Analüüsid selgus, et Boost on võrreldes Snap'iga parem või võrdne igas aspektis, välja arvatud visuaali läheduses praegu kasutatavale visuaalile (Tabel 1). Kuna visuaali on võimalik muuta, kuid teisi tabelis kajastatud aspekte mitte, siis otsustati valida baasteemaks Boost.

Tabel 1. Boosti ja Snapi võrdlus

Aspekt	Boost	Snap
Bootstrapi versioon	Bootstrap 4	Bootstrap 3
jQuery	Jah	Jah
Koodi arusaadavus ja eraldatus	Kujundused laaditakse Mustache failidest, mis muudab koodi arusaadavaks ja eraldatuks.	Kogu HTML genereeritakse PHP failides ning seega on kood vähem eraldatud ja arusaadav kui boostis.
Visuaali lähedus praegu kasutatavale visuaalile	Kauge	Lähedane
Kujunduskeel	SCSS – võrreldes CSS-iga arusaadavam, sest võimaldab valijaid (<i>selector</i> 'eid) kirjutada üksteise sisse, võimaldab defineerida muutujaid, mugavam kirjutada.	CSS3
Laiendatavus	Arusaadavalt laiendatav, internetis mitmeid juhendeid. Ülekirjutatavate failide hulk väiksem kui Snapi puhul.	Laiendamiseks tuleb suur osa koodist ümber kopeerida, mis tähendab sisuliselt otse snapi muutmist.
Vastuvõtlikkus Moodle'i uuendustele	Kui Moodle'it uuendatakse ning tekib vajadus uuendada kasutajaliidest, siis kasutajaliidest uuendatakse samuti samas versioonis.	Kui Moodle'it uuendatakse ning tekib vajadus uuendada kasutajaliidest, siis läheb aega enne kui Snapis viga parandatakse.

4 Arendus

Selles peatükis kirjeldatakse arendamiseks kasutatavaid tehnoloogiaid ning selgitatakse Boostile lisatud funktsionaalsust.

4.1 Moodle

Moodle (*Modular Object Oriented Dynamic Learning Environment*) on õpiplatvorm, mis pakub õpetajatele, õpilastele ja administraatoritele süsteemi isikupärastatud õpikeskkonna loomiseks [5]. Moodle'it kasutab üle saja miljoni inimese, mis teeb sellest maailma populaarseima õpiplatvormi [10]. Õpiplatvorm on tõlgitud rohkem kui 100-sse keelde, mis muudab Moodle'i kasutamise ülemaailmselt lihtsamaks [11].

Moodle on avatud lähtekoodiga tarkvara. Kõigil on lubatud kohandada, laiendada või muuta Moodle'i lähtekoodi nii ärilistel kui ka mitteärilistel eesmärkidel ilma litsentsimaksudeta. Moodle'i avatud lähtekood, modulaarne ülesehitus ning koostalitusvõimeline disain võimaldavad arendajatel luua lisamoduleid ning integreerida väliseid rakendusi, mis võimaldab Moodle'isse lisada mistahes funktsionaalsust. [5] Töös loodavaks Moodle'i lisamooduliks on teema (Moodle *theme*).

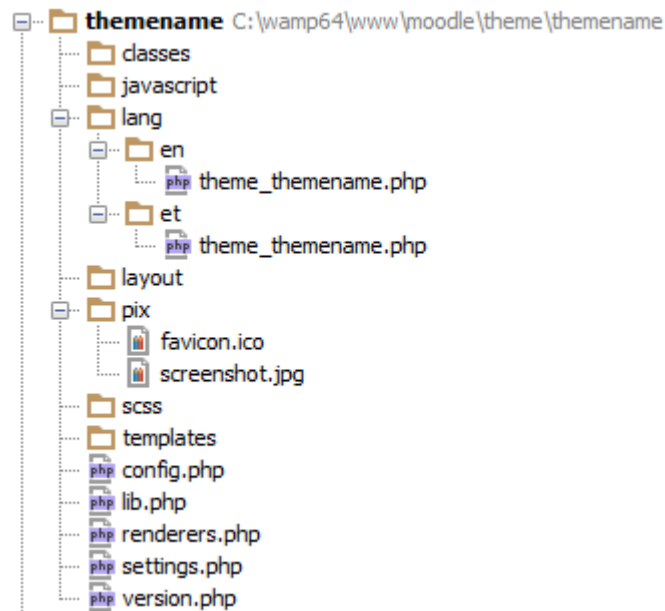
4.1.1 Moodle'i teema

Moodle'i teema on Moodle'i lisamoodul, mis võimaldab arendajal muuta õpikeskkonna välimust ja välimusega seotud funktsionaalsust. Moodulit saab rakendada tervele veebilehele, kategooriale, kursusele või tegevusele (*activity*). Teema saab eraldi luua arvutile, nutitelefonile ja tahvelarvutile. [6] Antud töös luuakse teema arvutile, mis töötab ka teistel nimetatud seadmetel.

Kõik Moodle'i teema failid asuvad Moodle'i juurkausta alamkaustas `/theme/<themenam>`, kus `<themenam>` on loodud teema nimi, ja selle alamkaustades. Teema sisaldab nelja kohustuslikku faili: `config.php`, `settings.php`, `version.php`, mille sisusid kirjeldatakse järgmises alapeatükis. [6] Töös luuakse uus teema laiendades olemasolevat Boosti.

4.1.2 Moodle'i teema loomine Boosti laiendades

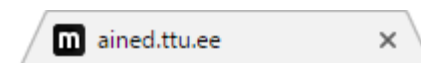
Laiendades Boosti tuleb kasutada järgnevat struktuuri. Nagu iga Moodle'i lisamoodul, peab ka Boosti põhjal loodava teema struktuur järgima Moodle'i lisamoodulite struktuuri standardeid. Boosti põhjal loodava teema struktuuri iseloomustab joonis 1.



Joonis 1. Boosti põhjal loodava teema struktuur.

Joonisel 1 näidatud teema struktuuri seletused:

- /classes – sisaldab funktsionaalsuse saavutamiseks vajaminevaid PHP klasside faile.
- /javascript – sisaldab funktsionaalsuse saavutamiseks vajaminevaid JavaScripti faile.
- /lang – sisaldab keelefaile. /lang/en/theme_themename.php ja /lang/et/theme_themename.php vastavalt inglise- ja eestikeelsed.
- /layout – sisaldab vaadete kujunduste (*layouts*) faile.
- /pix – sisaldab teema poolt kasutatavaid pildifaile.
 - favicon.ico – veebilehe tunnusikoon, mis kuvatakse veebilehe pealkirja kõrval (joonis 2).



Joonis 2. ained.ttu.ee favicon.ico ja pealkiri

- `screenshot.jpg` - teema kaanefoto.
- `/scss` – sisaldab SCSS-i faile.
- `/templates` – sisaldab Mustache malli faile, kus defineeritakse lehe HTML struktuur.
- `lib.php` – kohustuslik fail, milles defineeritakse lisaomadused, näiteks SCSS failide laadimine.
- `renderers.php` – fail, milles kirjutatakse üle vajaminevad *renderer*'id.
- `settings.php` – fail, kus defineeritakse lisasätteid teema jaoks. Loodavas teemas kasutatakse samu seadeid, mis on Boostil (näiteks brand color ja lisa SCSS-i lisamise võimalus koodi muutmata).

Failide `config.php` ja `version.php` sisud seletatakse järgnevas alapeatükis.

4.1.3 Moodle teema `config.php` ja `version.php` failide sisud

Fail `config.php` sisaldab kogu teemaga seotud konfiguratsiooni. Joonisel 3 on näide teema `config.php` faili sisust. Real number 1 defineeritakse teema nimi ning real number 2 öeldakse, millist/milliseid teemasid laiendatakse. Rida number 15 on vajalik selleks, et üle kirjutada mistahes Moodle'i *renderer*'e (komponendi HTML-i genereerimiseks mõeldud PHP fail). Muutujas `$THEME->layouts` (read 3 - 14) defineeritakse, millist layouti, kindlatel lehtedel kasutatakse, näiteks kursuse leht võetakse failist `course.php`. [4]

```

1 $THEME->name = 'themename';
2 $THEME->parents = ['boost'];
3 $THEME->layouts = [
4     'base' => array(
5         'file' => 'columns2.php',
6         'regions' => array(),
7     ),
8     'course' => array(
9         'file' => 'course.php',
10        'regions' => array('side-pre'),
11        'defaultregion' => 'side-pre',
12        'options' => array('langmenu' => true)
13    )
14 ];
15 $THEME->rendererfactory = 'theme_overridden_renderer_factory'
```

Fail `version.php` sisaldab teemaga seotud versiooniandmeid. Joonisel 4 on näide teema `version.php` sisust. Rida 1 defineerib teema versiooni. Rida 2 ütleb, mis on vanim Moodle'i versioon, mida teema toetab. Muutuja `$plugin->dependencies` (read 4-6) sisaldab teisi mooduleid ja nende versioone, millest teema sõltub. Kõik versiooninumbrid sisaldavad aastat, kuud, kuupäeva ning kaks viimast numbrit on versiooni täpsustamiseks. [12]

```
1 $plugin->version = 2017051105;
2 $plugin->requires = '2016070700';
3 $plugin->component = 'theme_myboost';
4 $plugin->dependencies = [
5     'theme_boost' => '2016102100'
6 ];
```

Joonis 4. Näide teema `version.php` failist.

4.2 Mirage

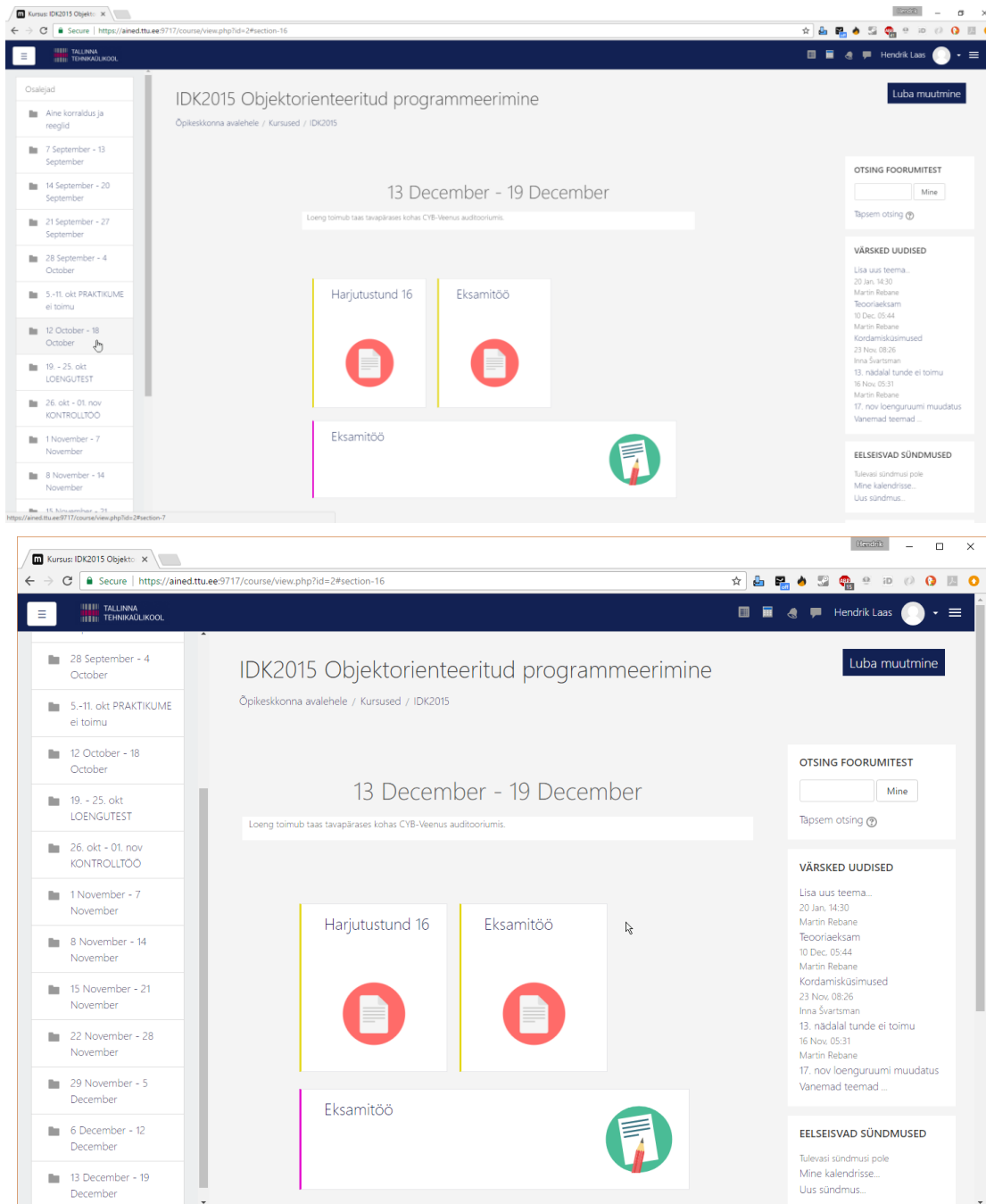
Loodud teema nimeks valiti Mirage. Mirage valiti loodusnähtuse järgi, kuid ei oma sisulist tähendust. Selles peatükis kirjeldatakse Mirage'i loomise protsessi ning põhiaspekte – selgitatakse kohanduva (*responsive*) disaini põhimõtteid, kirjeldatakse SCSS-i failide ülesehitus ning tuuakse välja, milline funktsionaalsus lisati JavaScriptis ja milline PHP-s.

4.2.1 Kohanduv disain

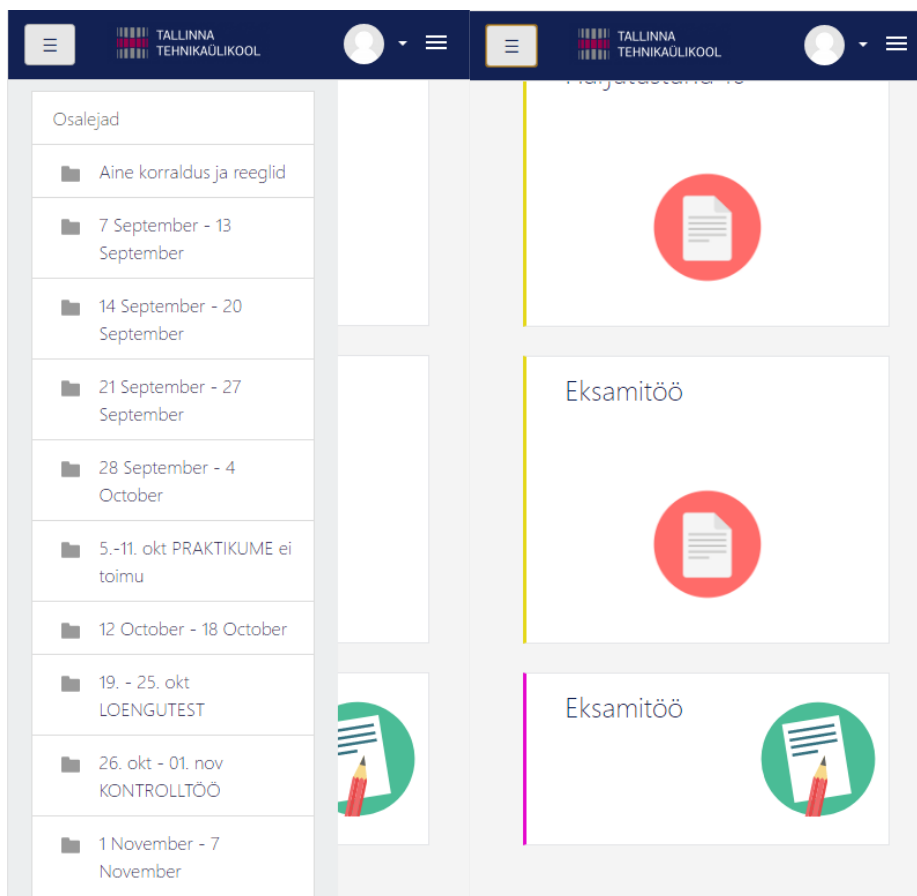
Kohanduv (*responsive*) veebidisain on disainivorm, mis arvestab seadmes kasutatava brauseriakna suurusega ning kohandub vastavalt sellele [7]. Lihtsaim näide kohanduvast disainist on veebilehe arendamine nii, et see töötaks korrektselt ka nutitelefoni. Töös saavutatakse kohanduv disain peamiselt kasutades CSS3-e sisseehitatud `@media` päringut, mis võimaldab rakendada CSS-i reegleid ainult kindla laiuse või kõrgusega brauseriakna või seadme suuruse korral (näiteks päringu `@media (max-width: 377px) {}` loogiliste sulgude vahele kirjutatavad reeglid rakenduvad ainult siis, kui akna laius on 377 pikslit või vähem).

Joonisel 5 on toodud kursuse vaade aknas, mille laius on 1920 pikslit (monitor) ning sama kursuse vaade aknas laiusega 1440 pikslit (tavaline sülearvuti). Joonisel 6 on toodud sama

kursuse vaade aknas laiuusega 360 pikslit (nutitefon). Joonised on tehtud administraatorina ning seega kuvatakse nuppu „Luba muutmine“, mida õpilastele ei näidata. Joonisel 5 on näha, kuidas lehe põhiosa (osa, mis sisaldab nädala sisu, näiteks „Eksamitöö“) protsentuaalne laius kogu veebilehe laiuse suhtes suureneb, kui kogu veebilehe laius väheneb. Lisaks on näha, et nutitefonil ei kuvata lehe navigatsiooniribal hinnete, kalendri, teavituste ega sõnumite ikoone, vaid nendele pääseb ligi rippmenüüst.



Joonis 5. Kursuse vaade akendes laiuusega 1920 pikslit ja 1440 pikslit.



Joonis 6. Kursuse vaade nutitelefonis laiusega 360 pikslit.

4.2.2 SCSS

Kuna Boost kasutab HTML-i elementide kujundamiseks tavaliselt kasutatava CSS3-e asemel SCSS-i, siis kasutakse arendatava teema Mirage loomisel kujundamiseks samuti SCSS-i. SCSS on kujunduskeele SASS-i üks süntaksitest. SASS on kujunduskeel, mis on mõeldud CSS-i reeglite kirjutamise lihtsustamiseks. SASS-is kirjutatud reeglid teisendatakse edasi CSS-i [9]. Kogu HTML-i elementide kujundus on töös kirjutatud SCSS-is.

SCSS-i eelised CSS-i süntaksi ees

SCSS-il on kaks suurt eelist CSS-i süntaksi ees. Esimene eelis on võimalus luua muutujaid, mis muudab koodi kirjutamise mugavamaks ja koodi kergemini muudetavaks, sest muutujat on võimalik ära kasutada kogu koodi ulatuses ning muutes muutuja väärtust on võimalik mitut elementi korraga muuta (näiteks taustavärv). SCSS-i teiseks eeliseks on võimalus kirjutada *selector*'eid üksteise sisse, mis teeb koodi arusaadavamaks ja koodi kirjutamise lihtsamaks. [9]

SCSS-i ülesehitus Mirage's

Kõik SCSS-i failid on paigutatud alamkausta `mirage/scss/moodle`. Kõik põhireeglid asuvad failis `default.scss` ning rakenduvad kõikidele vaadetele, kui teistes failides ei ole defineeritud teisiti. Failis `default.scss` on näiteks reeglid taustavärvide ja tabelite üldise välimuse kohta.

Põhilised reeglid (näiteks pealkirjade suurus), mis sõltuvad ekraani suuruselt, on kirjutatud faili `mobile.scss`. See fail on erinevate ekraani laiuste järgi organiseeritud ning sisaldab reegleid, mis rakenduvad nutitelefonidele, tahvelarvutitele ning erinevate suurusel arvutiekraanidele. Ekraani suuruselt sõltuvad reeglid, mis rakenduvad üksikutele vaadetele on kirjutatud nende vaadete SCSS-i failide lõppu.

Iga vaate spetsiifilisemad reeglid on paigutatud eraldi failidesse, millele on püütud anda võimalikult palju vaadet kirjeldav nimi. Näiteks asub kõik kalendriga seonduv failis `calendar.scss` ja kõik sõnumitega seonduv failis `messages.scss`.

4.2.3 JavaScriptiga lisatud funktsionaalsus

Töös kasutati JavaScriptiga tehtava töö hõlbustamiseks jQuery raamistikku, mis muudab JavaScripti funktsionaalsuse (näiteks animatsioonide kasutamine, sündmuste käsitlemine) kasutamise lihtsamaks ja mugavamaks [7].

JavaScriptiga lisati Boostile kaks peamist funktsionaalsust:

- kursuse lehele navigeerimisel aktiivse sektsiooni kuvamine
- administraatorimenüü lülitamine

Aktiivse nädala/teema kuvamine

Praeguse sektsiooni kuvamise eelduseks on, et kursus on nädalate ning teemade (*topics*) formaadis ning kursus on seadistatud näitama kõiki sektsioone korraga, mis tähendab, et kursuse lehele navigeerides laaditakse kogu kursuse sisu korraga. SCSS-iga lisati reegel, mis peidab lehe laadimisel kõik sektsioonid ning seejärel, kui leht laaditakse otsitakse aktiivne sektsioon üles ning tehakse see nähtavaks.

Teistele nädalatele navigeerimiseks on kursuselehe vasakus servas navigeerimismenüü, mis sisaldab kõikide sektsioonide pealkirju. Vajutades sektsiooni pealkirja peale peidetakse eelmine kuvatud sektsioon ning näidatakse sektsiooni, millele vajutati. Selline funktsionaalsus saavutatakse JavaScripti võimalusega reageerida URL-i (*Uniform Resource Locator*) muutustele. Kui navigeeritakse teisele sektsioonile, muudab Moodle veebilehe URL-i lisades sinna sektsiooni numbri. Kui URL muutub, kutsutakse JavaScriptis välja vastav funktsioon, mis peidab ühe sektsiooni ning teeb teise sektsiooni nähtavaks.

Administraatorimenüü lülitamine

Kui kasutajal on piisavalt õigusi, et näha administraatorimenüüd, siis paigutatakse menüü vaate laadimisel lehe paremasse serva ning peidetakse SCSS-is. Samuti paigutatakse lehe navigatsiooniribale ka navigeerimisnupp administraatorimenüü nähtavaks/nähtamatuks tegemiseks. Kui nupule vajutatakse, siis tehakse menüü nähtavaks ning uuesti vajutades peidetakse menüü.

4.2.4 PHP-ga lisatud funktsionaalsus

Enamik Moodle'i vaadetes kuvatavaid komponente genereeritakse PHP-s, mis tähendab, et mingi komponendi muutmiseks tuleb üle kirjutada Moodle'i funktsioon, mis muudetava komponendi HTML-i genereerib. Arenduse üks raskematest osadest oligi Moodle'i koodis orienteerumine ning õige funktsiooni üles leidmine. Moodle on ülesehitatud nii, et iga mooduli (näiteks kalender, kursus, sõnumid jne) jaoks on oma `renderer.php` fail, kus genereeritakse selle mooduli jaoks vajalik HTML.

Lisaks Boosti funktsionaalsusele lisati PHP-ga:

- hinnete lehe ja kalendri nupud lehe navigatsiooniribale

- muutmise lubamise nupp ja ploki lisamise nupp kursusevaate päisesse
- administraatorimenüü lehe paremasse serva
- lähenevad tähtjad ning hiljutine tagasiside pealehele, kui kasutaja on sisselogitud

Lisaks muudeti sektsiooni lohistamise nupu ja sektsiooni muutmise nupu HTML-i elementide asukohta, mis kuvatakse, kui on aktiveeritud kursuse lehe muutmise. Muutus oli vajalik selleks, et nimetatud nupud oleksid paigutatavad üksteise kõrvale. Muudeti kursuse kastis kuvatavat õpetajate loetelu.

Renderer .php laiendamine

Komponendi muutmiseks tuleb kõigepealt leida Moodle'i lähtekoodist üles `renderer.php` fail, mis antud mooduli näitamise tegeleb ning seejärel funktsioon, mis soovitud komponendi HTML-i genereerib. Järgmiseks tuleb loodavas teemas, failis `renderers.php`, leitud `renderer.php`-d laiendada. Näiteks, kui on vaja midagi kursuse lehel muuta, siis tuleb leida üles `/course/renderer.php` ning see üle kirjutada teema `renderers.php` failis. Näide `renderer.php` laiendamisest on toodud joonisel 7.

```
require_once($CFG->dirroot.'/course/format/weeks/renderer.php');

class theme_mirage_format_weeks_renderer extends format_weeks_renderer {}
```

Joonis 7. Näide `renderer.php` laiendamisest.

Töös on parema eraldatuse saavutamiseks siiski kõik laiendatavad `renderer.php` failid erinevates failides, mis on `renderers.php` failis välja kutsutud PHP `require_once` funktsiooniga. Töös on failid nimetatud:

- `core_renderer.php` – Alus HTML-i elementide genereerimine (näiteks vaadete päised).
- `course_format_topics_renderer.php` - teemade formaadiga kursuste HTML-i elementide genereerimine.
- `course_format_weeks_renderer.php` – nädalate formaadiga kursuste HTML-i elementide genereerimine.

- `course_renderer` – kursusega seotud HTML-i elementide genereerimine (näiteks õpetajate loetelu).

Kõik loetletud failid on `renderers.php`-s välja kutsutud funktsiooniga `require_once`, näiteks `require_once(__DIR__ . '/renderers/core_renderer.php')`.

Hinnetelege ja kalendri nuppude lisamine lehe navigatsiooniribale

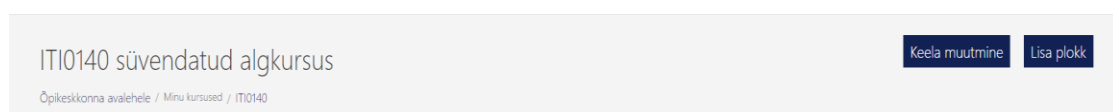
Hinnetelege ja kalendri nuppude lisamiseks kirjutati üle funktsioon, mis võimaldab päisesse lisada kohandatud menüüsid. Funktsioonis lisatakse päisesse kaks uut menüüd, mis on tegelikult lihtsalt lingid. Linkide suurust muudekase ning linkidele lisatakse taustapildina logo SCSS-is. Kui lingid on lisatud kutsutakse välja kasutatava funktsiooni *parent* (funktsioon, mis üle kirjutati), et tagada üle kirjutatud funktsiooni ülejäänud funktsionaalsus. Joonisel 8 on näidatud lisatud nupud.



Joonis 8. Navigatsiooniribale lisatud hinnetelege ja kalendri nupud.

Muutmise lubamise/keelamise ja ploki lisamise nupud kursusevaate päises

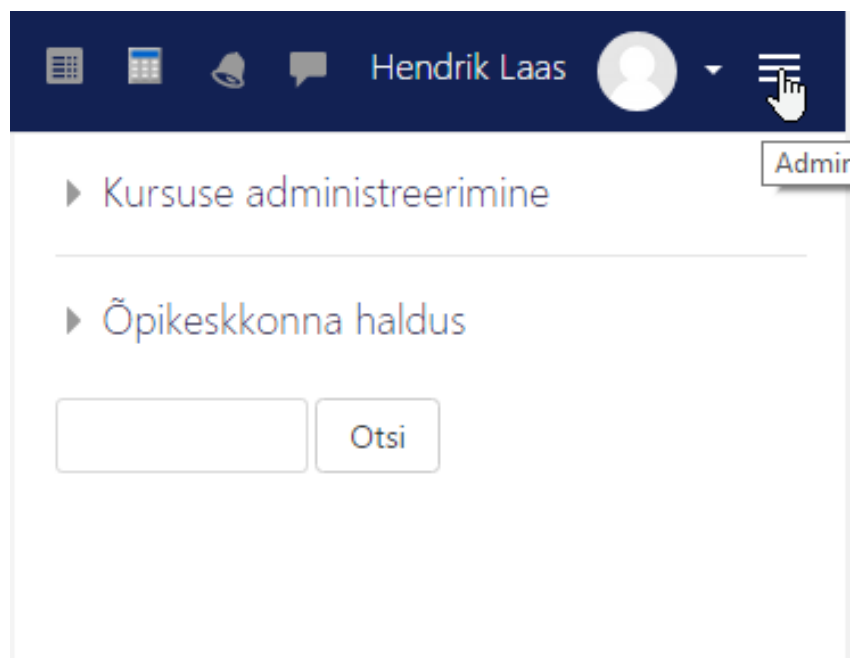
Muutmise lubamise/keelamise ja ploki lisamise nupud kursusevaate päisesse lisamiseks kirjutati üle funktsioon, mis genereerib kõikide vaadete päised, ning lisati sinna funktsionaalsus, mis kõigepealt kontrollib, kas ollakse kursuse lehel. Kui ollakse kursuse lehel, siis kutsutakse välja funktsioon, mis tagastab muutmise nupu HTML-i. Plokkide lisamise nupp lisatakse igale lehele, kus plokkid on lubatud, eeldusel, et kasutajal on õigus antud lehele plokkide lisada. Seega kontrollitakse, kas lehel on plokkid lubatud ning kas kasutajal on õigus plokkide lisada ning kutsutakse välja loodud funktsioon, mis tagastab plokkide nupu lisamise HTML-i. Joonisel 9 on näidatud kursuse lehe päisesse lisatud nupud.



Joonis 9. Muutmise keelamise ja ploki lisamise nupud kursuse vaate päises.

Administraatorimenüü lisamine lehe paremasse serva

Administraatorimenüü lisatakse teemale reaga `$THEME->requiredblocks = 'settings'` `config.php` failis, mis lisab igale lehele administraatorimenüü, kui kasutajal on õigus seda näha. Järgmiseks paigutati administraatorimenüü lehe paremasse äärde ning peideti see SCSS-ga. Joonisel 10 on näidatud administraatorimenüü lehe paremas servas.



Joonis 10. Administraatorimenüü lehe paremas servas.


Lähenevate tähtaegade ja hiljutise tagasiside lisamine pealehele

Lähenevate tähtaegade ja hiljutise tagasiside lisamiseks kasutati suures osas praegu kasutuses oleva teema (Snap) lähtekoodi ning kohandati seda arendatavale teemale (Mirage) ja muudeti koodi arusaadavamaks. Lähenevad tähtajad saadakse kasutades Moodle'i funktsionaalsust, mis tagastab kasutaja unikaalse identifikaatori järgi kõik saabuvad tähtajad. Hiljutise tagasiside saamiseks tuli aga läbi käia kõik kursused, kuhu kasutaja registreeritud on ning nendes olevad ülesanded. Joonisel 11 on näidatud saabuvad tähtajad ja hiljutine tagasiside, kui kasutaja on sisselogitud.

Deadlines

You have no upcoming deadlines.

Recent Feedback

 Test course 1 / The most important
Released: Wednesday, 26 April, 3:58 PM

Joonis 11. Lähenevad tähtajad ja hiljutine tagasiside pealehel.

Sektsiooni muutmise ja liigutamise nupu HTML-i elementide asukoha muutmine kursuse lehel, kui kursuse muutmine on lubatud

HTML-i elementide järjekorra muutmiseks kursuse lehel tuli laiendada kahte erinevat `renderer.php` faili: `/course/format/weeks/renderer.php` ja `/course/format/topics/renderer.php`. Kuna nendest mõlemad sisaldavad sama funktsiooni, mis genereerib sektsiooni päise HTML-i, siis kasutati koodihulga vähendamiseks PHP funktsiooni `trait` (funktsioon, mis mõeldud koodi taaskasutamiseks). `trait` asub eraldi failis ning seda kutsutakse välja mõlemas laiendatud failis. Joonisel 12 on näidatud kõrvuti paiknevad sektsiooni lohistamise ja muutmise nupud.



Joonis 12. Sektsiooni lohistamise ja muutmise nupud

Kursuse kastis kuvatavate õpetajate loetelu muutmine

Õpetajate loetelu muudeti eesmärgiga kuvada kõik õpetajad ühel (kui ühele reale ei mahu, siis ka mitmel) real, mitte nummerdatuna üksteise all. Sellise kuvamise kasuks otsustati, sest see võimaldab ruumi kokku hoida. Muutuse tegemiseks tuli laiendada faili `/course/renderer.php` ning muuta funktsiooni, mis genereerib kursuse kasti sisu. Joonisel 13 on näidatud saavutatud tulemus.

Capella Test

Teachers: Joosep Franz Moorits Alviste, Teach Her, Artur Luik, Brenda Uga, Admin User

Joonis 13. Kursuse õpetajate loetelu ühel real.

4.3 Testimine erinevatel brauseritel

Arendatud teemat testiti käsitsi erinevatel veebilehitsejatel. Selgus, et teema töötab kõikide testitud lehitsejate uusimatel versioonidel. Küll aga avastati testimise käigus, et mõnede lehitsejate vanemates versioonides (näiteks Safari 7) kõik vaated ei tööta. Kuna nõue oli, et teema peab töötama brauserite uusimatel versioonidel, siis vanemaid versioone põhjalikult läbi ei testitud, kuid leitud vead parandati SCSS-i reeglite lisamisega. Näiteks parandati kursuste loetelu avalehel ning tegevuste (*activity*) taustapildid. Tabel 2 esitab testitud vaadete töötavuse erinevatel veebilehitsejatel.

Tabel 2. Erinevate vaadete töötavus erinevatel veebilehitsejatel

Vaade	Google Chrome (versioon 57)	Mozilla Firefox (versioon 53)	Microsoft Edge (versioon 3)	Safari (versioon 10)	Opera (versioon 45)
Pealeht	✓	✓	✓	✓	✓
Kursuse vaade	✓	✓	✓	✓	✓
Kursuse vaade, kui kursust muudetakse	✓	✓	✓	✓	✓
Sõnumite vaade	✓	✓	✓	✓	✓
Teadete vaade	✓	✓	✓	✓	✓

Hinnete vaade	✓	✓	✓	✓	✓
Kalendri vaade	✓	✓	✓	✓	✓
Seadete vaade	✓	✓	✓	✓	✓
Töölaua vaade	✓	✓	✓	✓	✓
Kasutaja andmete vaade	✓	✓	✓	✓	✓
Kursuste otsingu vaade	✓	✓	✓	✓	✓
Sisselogimise vaade	✓	✓	✓	✓	✓

4.4 Edasiarendused

Kuna lõputöö sooritamise aeg oli piiratud, siis ei jõutud kasutajaliidesele lisada funktsionaalsust, mille lisamise plaan tekkis töö lõpuosas. Selles peatükis esitatakse arendatud kasutajaliidese suuremad puudused ning pakutakse välja lahendusi, kuidas puudused likvideerida ning kasutajaliidese kasutamist mugavamaks muuta.

Pealehe sisu

Kuna kasutajaliides läheb kasutusse kogu Tallinna Tehnikaülikooli ulatuses, siis Moodle'is olevate õppeainete hulk on suur ning ei ole otstarbekas neid pealehel kuvada. Küll aga on pealehel kasutajale lisaks lähenevatele tähtaegadele ja hiljutisele tagasisidele mõistlik kuvada aineid, milles ta osaleb ning otsingukasti kõikide kursuste leidmiseks. Seda ka valminud kasutajaliides võimaldab, kuid probleemiks on pealeht, kui kasutaja ei ole sisse logitud.

Kui kasutaja on välja logitud, siis võiks talle kuvada otsingukasti ning ülejäänud osa lehest täita informatsiooniga, kuidas keskkonda kasutada. Lisaks võiks pealehel kuvada populaarseid kursusi, kuigi nende kuvamise lisamine on keerukas. Lisaks võiks pealehel kuvada menüüna loetelu kursuse kategooriatest, millele vajutades viiakse kasutaja antud kategooria ainete juurde.

Kursuste otsing

Kuna aineid on palju ning neid pealehel ei kuvata, siis oleks otstarbekas lisada kursuse otsingu kastidele funktsionaalsus, mis juba esimese tähe sisestamisel reaajas otsingukasti all sobivaid kursusi kuvab. Näiteks on võimalik sellist otsingut luua kasutades AJAX-i (*asynchronous* JavaScript and XML) päringut, mis võimaldab taustal serverist informatsiooni pärida. Selline lahendus parandaks oluliselt kasutusmugavust, sest otsitav kursus oleks võimalik üles leida juba otsingukasti all ning ei peaks navigeerima vaatesse, mis sobivaid kursusi kuvab.

5 Kokkuvõte

Töö eesmärk oli luua õpiplatvormile Moodle Tallinna Tehnikaülikooli temaatiline ja Moodle'i uuendustele vastuvõtlik kasutajaliides, mis võimaldab kasutada kogu Moodle'i poolt pakutavat funktsionaalsust.

Üldtunnustatud tavade analüüs kasutajaliidese loomisel ning eelmise kasutatava kasutajaliidese kohta tehtud küsitluse analüüs andis aimu, milline arendatav kasutajaliides olla võiks ning aitas kaasa nõuete kirjutamisel. Töö lugeja peaks saama ülevaate kasutajaliidese lähtekoodi struktuurist, kasutatud tehnoloogiatest ning lisatud funktsionaalsusest.

Kokku sisaldab loodud teema 4299 rida koodi (koos kommentaaridega), millest 1972 on kirjutatud PHP-s, 1852 SCSS-is, 77 JavaScriptis ja 578 Mustache's (HTML). Koodi kõik funktsioonid on dokumenteeritud ning keerulisemad kohad kommenteeritud. Töö lähtekood on saadaval aadressil <https://gitlab.cs.ttu.ee/Hendrik.Laas/MoodleTheme>.

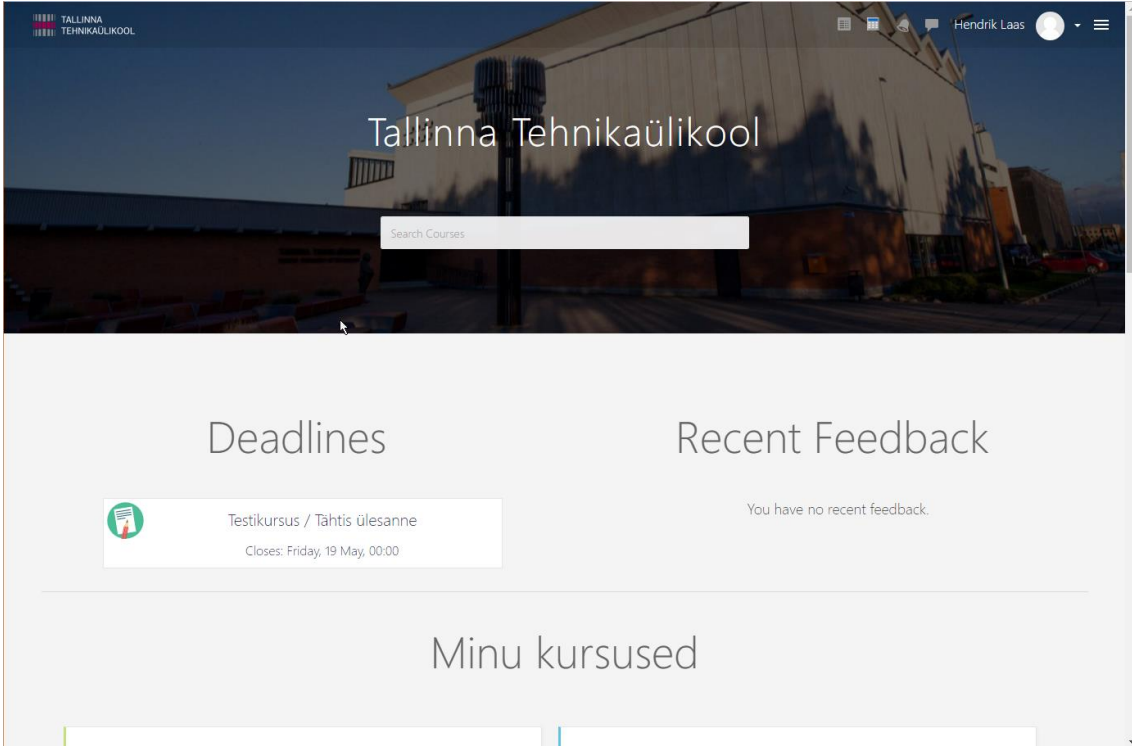
Töö tulemusena valmis Tallinna Tehnikaülikooli temaatiline kasutajaliidese teema (Moodle *theme*) nimega Mirage. Mirage on lihtsasti muudetav ning uue funktsionaalsusega täiendatav. Loodud teema on vastuvõtlik Moodle'i uuendustele ning toetab kogu Moodle'i funktsionaalsust.

Planeeritud funktsionaalsus sai loodud, kuid töö lõpuosas tekkis juurde mõtteid, kuidas liidese kasutusmugavust tõsta. Kasutusmugavuse tõstmiseks võiks lisada kursuste otsingu taustal ning pealehele kursuste otsimise peakategooriate järgi. Kui kasutaja on välja logitud võiks pealeht olla informatiivsem.

Kasutatud kirjandus

- [1] „5 aspects of a good user interface,“ [Võrgumaterjal]. Available: <http://www.argondesign.com/news/2014/feb/5/5-aspects-good-user-interface/>
[Kasutatud 03.05.2017]
- [2] „Effective Visual Communication for Graphical User Interfaces“ [Võrgumaterjal]. Available: https://web.cs.wpi.edu/~matt/courses/cs563/talks/smartin/int_design.html
[Kasutatud 03.05.2017]
- [3] „8 Characteristics Of Successful User Interfaces“ [Võrgumaterjal]. Available: <http://usabilitypost.com/2009/04/15/8-characteristics-of-successful-user-interfaces/>
[Kasutatud 25.02.2017]
- [4] „Creating a theme based on boost“ [Võrgumaterjal]. Available: https://docs.moodle.org/dev/Creating_a_theme_based_on_boost [Kasutatud 27.02.2017]
- [5] „About Moodle“ [Võrgumaterjal]. Available: https://docs.moodle.org/32/en/About_Moodle [Kasutatud 10.05.2017]
- [6] „Themes overview“ [Võrgumaterjal]. Available: https://docs.moodle.org/dev/Themes_overview [Kasutatud 25.02.2017]
- [7] „Responsive Web Design: What It Is And How To Use It“ [Võrgumaterjal]. Available: <https://www.smashingmagazine.com/2011/01/guidelines-for-responsive-web-design/> [Kasutatud 20.04.2017]
- [8] „jQuery documentation“ [Võrgumaterjal]. Available: <http://api.jquery.com/>
[Kasutatud 05.05.2017]
- [9] „Sass Basics“ [Võrgumaterjal]. Available: <http://sass-lang.com/guide> [Kasutatud 02.03.2017]
- [10] „Moodle Statistics“ [Võrgumaterjal]. Available: <https://moodle.net/stats/>
[Kasutatud 12.05.2017]
- [11] „Language packs“ [Võrgumaterjal]. Available: https://docs.moodle.org/32/en/Language_packs [Kasutatud 12.05.2017]
- [12] „version.php“ [Võrgumaterjal]. Available: <https://docs.moodle.org/dev/version.php>
[Kasutatud 12.05.2017]

Lisa 1 – Pealeht




TALLINNA
TEHNIKAÜLIKOO

Hendrik Laas

Tallinna Tehnikaülikool

Search Courses

Deadlines

 Testikursus / Tähtis ülesanne
Closes: Friday, 19 May, 00:00

Recent Feedback

You have no recent feedback.

Minu kursused

Lisa 2 – Kursuse vaade

The screenshot shows the course overview for ITI0140. On the left is a sidebar with a calendar of dates from September to November. The main content area is titled "ITI0140 süvendatud algkursus" and lists several items: "Announcements", "TEST01", "ASS1", and "ainult custom". Each item has a corresponding icon. On the right, there are three informational boxes: "OTSING FOORUMITEST" with a search bar, "VIIMATINE TEGEVUS" showing the last activity on May 2017, and "VÄRSKED UUDISED" with a list of recent news items. The user's name "Hendrik Laas" is visible in the top right corner.

This screenshot shows the calendar view of the course. The left sidebar is identical to the first screenshot. The main area displays a calendar with three visible dates: "10 September - 16 September", "17 September - 23 September", and "24 September - 30 September". Each date range has a "Muuda" (Edit) button. Below the calendar, there are three event cards: "ainult custom" (with a plus icon and edit button), "Konsult test?" (with a plus icon and edit button), and "EX02 Test Errors in Form" (with a plus icon and edit button). Each card also has a "Lisa tegevus või vahend" (Add activity or resource) button. The right sidebar is partially visible, showing the "VIIMATINE TEGEVUS" section.

Lisa 3 - Töölaua vaade

The screenshot displays a user interface for Hendrik Laas. At the top left, the logo for Tallinn University of Applied Sciences (TALLINNA TEHNIKAÜLIKOOL) is visible. The user's name, Hendrik Laas, is shown in the top right corner along with a profile picture and a 'Uuenda lehte' (Refresh page) button. The main content area is titled 'KURSUSE ÜLEVAADE' (Course Overview) and contains two course cards: 'Testikursus' and 'ITI0140 süvendatud algkursus'. On the right side, there are four sidebar panels: 'PRIVAATSED FAILID' (Private Files) showing no files; 'VÖRGUS OLEVAD KASUTAJAD' (Active Users) showing one user, Hendrik Laas; 'UUSIMAD ÕPIMÄRGID' (Latest Grades) showing no grades; and 'KALENDER' (Calendar).

Lisa 4 – Seadete vaade

ained.ttu.ee

Õpikeskkonna avalehele / Õpikeskkonna haldus / Valimus / Kujundused / Kujunduse seaded

Luba plokkide muutmine

Kujunduse seaded

Kujunduste loend
themeist

Vaikimisi:

Kui soovite, et kasutada saaks kõiki kujundusi, jätke see väli tühjaks. Kui soovite, et kujundusemenüü oleks lühike, saate siin määrata kujunduste komaerialdusega loendi (ärge kasutage tühikuid). Naide: standard,orangewhite.

Malli kujundaja režiim
themedesignermode

Vaikimisi:

Tavaliselt kõik kujundus pildid ja stiilid on puhverdatud vahemällu veebilehitsejates ja serveris pikaks ajaks, jõudluse nimel. Kui lood uut kujundust või programmeerid koodi, siis tasuks see välja lülitada, et sa ei näeks keskkonda puhverdatud kujul. Hoiatust: See teeb lehe aeglasemaks kõikide kasutajate jaoks! Alternatiivina, saad samas ka käsitsi vahemälu tühjendada Kujunduse valimise lehelt.

Luba kasutaja kujundused
allowuserthemes

Vaikimisi:

Kui sa lubad selle, siis kasutajatel on võimalik määrata ise kujundust. Kasutaja kujundus on ülimalikud õpikeskkonna kujunduste suhtes (aga mitte kursuse kujunduste suhtes).

Luba kursuse kujundused
allowcoursethemes

Vaikimisi:

Kui sa lubad selle, siis saavad kursused seada oma kujundust. Kursuste kujundused on ülimalikud kõigi teiste kujunduste vaikute (õpikeskkond, kasutaja, sessioon) suhtes.

ADMINISTRAATORI JÄRJEHOIDJAD

Lisa see leht järjehoidjasse

Lisa 5 – Mobiilivaated

