

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Informaatikainstituut

Infosüsteemide õppetool

**Õppeinfosüsteemi õpingukavade
allsüsteemi üliõpilase töökoha
veebiliidese prototüüpimine**

Bakalaureusetöö

Üliõpilane: Kaspar Kask

Üliõpilaskood: 123997IAPB

Juhendaja: dotsent Erki Eessaar

Tallinn
2015

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.

(kuupäev)

(allkiri)

Annotatsioon

Õppeinfosüsteemi õpingukavade allsüsteemi üliõpilase töökoha veebiliidese prototüüpimine

Käesoleva bakalaureusetöö eesmärk on luua näiterakendus andmebaaside õppeaine näiteprojektile, kasutades programmeerimiskeelt PHP ja andmebaasisüsteemi Oracle. Kuna näiterakenduse aluseks on Erki Eessaare (ühtlasi käesoleva töö juhendaja) poolt loodud õppeinfosüsteemi õpingukavade allsüsteemi näiteprojekt, siis otsustati loodud rakenduse kasutajaliidest ja kasutajakogemust võrrelda praeguse Tallinna Tehnikaülikooli õppeinfosüsteemi õpingukavade koostamise kasutajaliideselega.

Töö alguses kirjeldatakse loodud näiterakenduse ülesehitust ja struktuuri. Seejärel kirjeldatakse loodud veebiliidest ja liidese loomisel aluseks olnud kasutajaliidese loomise häid tavasid ning disainimustreid. Töö lõpus võrreldakse loodud kasutajaliidest Tallinna Tehnikaülikooli õppeinfosüsteemiga kasutajaliidese loomisel aluseks olnud eksperthinnangute ja disainimustrite kogumike alusel.

Töö tulemusena on valminud programmeerimiskeelt PHP ja andmebaasisüsteemi Oracle kasutav näiterakendus andmebaaside õppeainet sooritavatele tudengitele. Samuti on toodud välja murekohad praeguse õppeinfosüsteemi õpingukavade koostamise kasutajaliidesele.

Loodud rakendus on leitav aadressil <http://apex.ttu.ee/opingukava>.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 49 leheküljel, 5 peatükki, 26 joonist ja 2 tabelit.

Abstract

Prototyping the Web Interface for Students in case of the Study Plan Subsystem of a Study Information System

The main goal of this thesis is to create an example of a web application using the PHP programming language and Oracle database management system for students who study database subjects. This application is based on an example project by Erki Eessaar (who is also the supervisor of the thesis) describing an information system where students can declare study plans in a Study Information System. Thus, it was decided to compare the user interface and user experience of the created application with the existing Study Information System in use at Tallinn University of Technology.

The beginning of this thesis describes the structure of the created application. After that, the created user interface is described using the design patterns and good practices used to create the interface. The thesis is finished with a comparison of the created user interface and the existing user interface of TUT Study Information System based on user interface and experience design patterns.

The main result of this thesis is the example application using PHP and Oracle that can be used by future students studying databases. Another result is the description of problems in the design of the current study plan declaration user interface.

The thesis is in Estonian and contains 49 pages of text, 5 chapters, 26 figures and 2 tables.

Lühendite ja mõistete sõnastik

AJAX

Asynchronous JavaScript And XML

asünkroonne JavaScript ja XML

„Lühend AJAX võeti kasutusele 2005.a. ja sellega tähistatakse selliste interaktiivsete veebirakenduste loomise meetodit, kus toimub jooksev kulissidetagune andmevahetus brauseri ja veebiserveri vahel, nii et kasutaja iga liigutuse peale pole vaja kogu veebilehte uuesti alla laadida.“ (Vallaste 2015)

CSS

Cascading Style Sheets

kaskaadlaadistik

„Veebilehtede valmistajatele ja kasutajatele mõeldud laadistik. Laadilehed (style sheets) kirjeldavad, kuidas HTML dokumente esitada kuvaril, printeril või kõnesüntesaatorist kostva kõnena.“ (Vallaste 2015)

DOM

Document Object Model

dokumendiobjektide mudel

„Eeskiri selle kohta, kuidas objekte (tekst, pildid, pealkirjad, lingid jne.) veebilehel esitada. DOM määrab ära, millised atribuudid kuuluvad millise objekti juurde ning kuidas objekte ja atribuute käsitleda.“ (Vallaste 2015)

EAP

Euroopa ainepunkt

„Euroopa ainepunkt (EAP) on õppe mahu arvestamise ühik. 1 EAP vastab 26 tunnile üliõpilase poolt õppeks kulutatud tööle.“ (TTÜ õppekorralduse eeskiri, 2014)

eesrakendus

front-end application

„Veebilehele ilmuv kasutajaliides, mis võimaldab veebisaidi külastajal kahepoolselt suhelda saidi dünaamiliste osadega [---].“ (Vallaste 2015)

HTML

HyperText Markup Language

hüpertext-märgistuskeel

„Enimlevinud kodeerimissüsteem (tekstivorming) veebidokumentide

loomiseks. HTML koodid ehk märgendid määravad ära selle, kuidas veebileht arvutiekraanil välja näeb“ (Vallaste 2015)

HTTP

HyperText Transfer Protocol

hüpertexti edastusprotokoll

„TCP/IP klient-server protokoll HTML-dokumentide vahetamiseks veebis ehk lihtsamalt öeldes andmevahetusprotokoll, mida kasutatakse Internetis dokumentide vahetamiseks“ (Vallaste 2015)

JS

JavaScript

„[---] [S]kriptikeel, mis võimaldab veebiautoritel luua interaktiivseid veebisaite. [---] JavaScript suudab suhelda HTML-keeles kirjutatud lähtekoodiga ja võimaldab muuta veebilehed dünaamiliseks.“ (Vallaste 2015)

Less

CSS keele eeltötlusvahend

MIT ehk X11

MIT license, X11 license

litsents

Nõrkade piirangutega, seaduse poolt seatud autoriõigustest loobuv vaba tarkvara litsents. (Free Software Foundation, Inc., 2015)

MVC

Model-View-Controller

Rakenduse disainimuster, kus rakendus jaotatakse struktureerimise ja koodi korduvkasutatavuse huvides osadeks.

PHP

PHP: Hypertext Preprocessor

PHP ehk PHP: Hypertext Preprocessor on skriptimiskeel, mida kasutatakse peamiselt serveripoolsetes lahendustes dünaamiliste veebilehtede loomisel. (Vikipeedia 2015)

raamistik

framework

Loodavat platvormi, tarkvara, riistvara, protokollide vms toetav struktuur (Vallaste 2015)

reageeriv disain

responsive design

Veebilehe kujundust nimetatakse „reageerivaks“ juhul, kui see kohandub

seadmele, millega veebilehte külastatakse. Reageeriv disain erineb sellest, kui server kuvab mobiiliga külastajatele erinevat lehte.

SQL

Structured Query Language

struktuurpääringukeel

„Enimkasutatav pääringukeel, mida toetavad kõik klient-server keskkonnale projekteeritud relatsioonandmebaasid. Pääringukeeled kujutavad endast reeglite kogumit, mille alusel konstrueeritakse pääringuid andmete otsimiseks andmebaasist.“ (Vallaste 2015) Lisaks pääringutele võimaldab see keel muuta andmebaasis andmeid, juhtida tehinguid ning hallata andmebaasiobjekte ning nendega seotud õiguseid.

SQL süstimine

SQL injection

„Rakenduse andmebaasikihi turvanõrkus, mida põhjustab SQL-lausesse manustatud dünaamiliselt genereeritud stringiliteraali ebakorrektnete pagu.“ (Vallaste 2015)

tagarakendus

back-end application

„Tagarakenduse" (back-end application) või –teenuse ülesandeks on kaudselt toetada eesteenusi ning nad on tavaliselt lähemal vajalikule ressursile ja/või suudavad sellega suhelda.“ (Vallaste 2015)

TTÜ

Tallinna Tehnikaülikool

URL

Uniform Resource Locator

internetiaadress

„Igale dokumendile või muule ressursile Internetis vastab oma unikaalne internetiaadress. Kui soovitakse pöörduda mingi veebilehe poole, siis kirjutatakse URL veebilehitseja aadressilahtrisse.“ (Vallaste 2015)

ÕIS

Õppeinfosüsteem

„TTÜ ametlik õppekorralduse infovahetuskeskkond, kus muu hulgas toimub õppetööd puudutavate andmete kogumine, töötlemine ja säilitamine ning informatsiooni edastamine“ (TTÜ õppekorralduse eeskiri, 2014)

Jooniste nimekiri

Joonis 2.2.1. Rakenduse vooskeem. Application Flow Chart : CodeIgniter User Guide.....	14
Joonis 3.3.1. Realiseeritud kasutajaliidese avaleht.....	22
Joonis 3.3.2. Sisselogimisvaade	23
Joonis 3.3.3. Veateade, kui kasutaja sisestatud andmetele ei leitud andmebaasist vastet.	24
Joonis 3.3.4. Kasutaja avakuva.....	25
Joonis 3.3.5. Sisselogitud kasutaja navigatsiooniriba.	26
Joonis 3.3.6. Õpingukavade üldvaade	27
Joonis 3.3.7. Teade, kui kasutajal on juba avatud õpingukava, ent ta üritab uut luua.....	28
Joonis 3.3.8. Õpingukava kustutamise kinnitust küsiv modaalaken.	28
Joonis 3.3.9. Tüüpõpingukavade vaade.....	29
Joonis 3.3.10. Moodulite vaade.	30
Joonis 3.3.11. Õpingukava detailvaade peale tüüpõpingukava alustamist.....	31
Joonis 3.3.12. Õppeainete rippmenüü vaikimisi ja täidetud otsingulahtriga	33
Joonis 3.3.13. Teade, kui tudeng valib õppeaine, mille kuulajaskond on täis.....	33
Joonis 3.3.14. Õpingukava esitamise kinnitust küsiv modaalaken.	34
Joonis 3.3.15. Kasutaja andmete vaade.	35
Joonis 4.1.1. Praeguse õppeinfosüsteemi esileht sisselogitud tudengile.	36
Joonis 4.1.2. Veateade mitmes raamis õppeinfosüsteemis.....	37
Joonis 4.2.1. Õpingukavade üldvaade õppeinfosüsteemis.	39
Joonis 4.2.2. Õpingukavade otsingu vorm.	39
Joonis 4.2.3. Dialoogiaken, mis küsib, kas tudeng soovib uut õpingukava lisada.	40
Joonis 4.3.1. Õpingukava detailvaade	41
Joonis 4.3.2. Õpingukava detailvaade peale tüüpõpingukava järgi genereerimist.....	42
Joonis 4.3.3. Õppeinfosüsteemi õpingukavasse aine lisamine.	43
Joonis 4.3.4. Aine-õppejõu paaride otsingu tulemus.....	44
Joonis 4.3.5. Õpingukavas olevate ainete tabel.	44

Tabelite nimekiri

Tabel 2.2.1. Rakenduses olevad mudelid.	17
Tabel 2.2.2. Rakenduses olevad kontrollid.	19

Sisukord

1. Sissejuhatus	11
1.1 Taust ja probleem	11
1.2 Ülesande püstitus	11
1.3 Metoodika.....	12
1.4 Ülevaade tööst	12
2. Veebiliidese tagarakendus	13
2.1 Rakenduse loomise vahendid	13
2.2 Veebirakenduse ülesehitus ja arhitektuur	14
2.2.1 CodeIgniteri seadistamine	14
2.2.2 CodeIgniteris rakenduse kirjutamine.....	15
3. Veebiliides	20
3.1 Veebiliidese loomise vahendid.....	20
3.2 Veebiliidese kavandamine	21
3.3 Veebiliidese funktsionaalsus	21
3.3.1 Veebiliidese avamine.....	22
3.3.2 Sisselogimisvaade.....	22
3.3.3 Kasutaja avakuva.....	24
3.3.4 Navigatsioon.....	26
3.3.5 Õpingukavade üldvaade	27
3.3.6 Tüüpõpingukavade vaade	29
3.3.7 Moodulite vaade	30
3.3.8 Õpingukava detailvaade	31
3.3.9 Kasutaja andmete vaade	35
4. Loodud veebiliidese võrdlus praeguse TTÜ Õppeinfosüsteemi veebiliideselega.....	36
4.1 Üleüldine kasutajaliidese ülesehitus	36
4.2 Õpingukavade vaade.....	38
4.3 Õpingukava detailvaade	41
5. Kokkuvõte	46
Summary.....	47
Kasutatud kirjandus	48

1. Sissejuhatus

Autor valis käesoleva bakalaureusetöö teema lähtudes soovist luua rakendus, mille loomisest on tulevikus kasu. Antud juhul ei ole tegu rakendusega, mida hakataks tulevikus funktsionaalselt kasutatama, vaid pigem on sellest abi TTÜ õppeaineid „Andmebaasid I“ ja „Andmebaasid II“ läbivatele tudengitele, kel seni puudus kasutatud tehnoloogiate abil loodud näiterakendus, mis demonstreeriks andmebaasi veebiliidest.

Kuna PHP võimaldab luua rakendusi paindlikumalt ja rohkem vastavalt programmeerija soovidele võrreldes Oracle poolt pakutava Application Express veebirakenduste kiirprogrammeerimise tarkvaraga, loodab autor, et sellise näiterakenduse olemasolu on positiivseks näiteks tulevastele tudengitele rakenduste loomisel ja andmebaaside ainete raames loodavate projektide teema valikul.

1.1 Taust ja probleem

Bakalaureusetöö tulemusena valmib seni puudunud programmeerimiskeelt PHP ja andmebaasisüsteemi Oracle kasutatav näiterakendus Tallinna Tehnikaülikooli õppeaineid „Andmebaasid I“ ja „Andmebaasid II“ läbivatele tudengitele. Tegemist on illustratsiooniga, milline võiks välja näha registri tüüpi infosüsteemi tarkvara ning see peaks aitama tudengeid paremini mõista, et nende kavandatavaid andmebaase hakkavad kasutama programmid ja läbi nende lõppkasutajad. Lisaks näiterakenduse loomisele võrreldakse loodud kasutajaliidest praegu Tallinna Tehnikaülikoolis kasutusel oleva õppeinfosüsteemi kasutajaliidese, toomaks välja murekohti ja vastuolu kasutajaliidese ja -kogemuse ekspertide sõnastatud heade disainimustritega.

Töö valmis 2015. aasta kevadel autori õpingute ajal Tallinna Tehnikaülikoolis.

1.2 Ülesande püstitus

Töö aluseks on juhendaja Erki Eessaare koostatud andmebaaside õppeainete näiteprojekt „Ülikooli infosüsteemi õpingukavade allsüsteem“ (Eessaar 2014). Töö tulemusena luuakse selle projekti põhjal näiterakendus andmebaaside õppeaineid läbivatele tudengitele. Samuti valmib loodud kasutajaliidese ja praeguse õppeinfosüsteemi kasutajaliidese võrdlus.

1.3 Metoodika

Näiterakenduse loomiseks kasutatakse programmeerimiskeelt PHP, sellel põhinevat rakendusraamistikku CodeIgniter ja andmebaasisüsteemi Oracle. Veebiliidese loomiseks kasutatakse HTML kujundusraamistikku UIKit ning JavaScripti rakendusraamistikku AngularJS.

Kasutajaliidese kavandamisel tutvutakse kasutajaliidese ja kasutajakogemuse ekspertide koostatud materjalide ja artiklitega hea kasutajaliidese koostamise teemadel. Samuti võttis autor liidese koostamisel arvesse mitmete kaastudengite arvamusi ja soovitusi. Kokku konsulteeris autor sellel teemal kaheksa tudengiga.

Õpingukava koostamisel registreeritavate andmete hulk on piiratud käesolevale tööle sisendiks oleva (lihtsustatud) õpingukavade allsüsteemi projektiga, mis näiteks ei näe ette võimalust õppekeele valmiseks (mis samas TTÜ õppeinfosüsteemi tarkvaras on võimalik).

Loodud ja olemasoleva kasutajaliidese võrdlemisel on lähtunud samadest alustest kui kasutajaliidese kavandamisel.

Käesolev töö on näiteks disainiteaduse (*design science*) metoodika rakendamisest, mille tulemuseks on uus ja vajalik artefakt (antud juhul näiterakendus) teatud probleemide lahendamiseks (antud juhul andmebaasi veebiliidese ja selle hea kasutatavuse illustreerimine) (Design Science 2015) ning mis loodi kasutades nii parimaid praktikaid kui kaastudengitelt saadud tagasisidet.

1.4 Ülevaade tööst

Töö alguses kirjeldatakse näiterakenduse ülesehitust ja struktuuri. Seejärel kirjeldatakse loodud veebiliidest ja selle kavandamisel kasutatud põhimõtteid ja lahendusi. Töö lõpus võrreldakse loodud veebiliidest praeguse õppeinfosüsteemi kasutajaliideselega.

2. Veebiliidese tagarakendus

Käesolev peatükk kirjeldab veebiliidese tagarakenduse loomiseks kasutatud vahendeid ning rakenduse tehnilist arhitektuuri.

2.1 Rakenduse loomise vahendid

Üheks töö eesmärgiks on luua andmebaaside projekteerimist ning programmeerimist õppivatele tudengitele näiterakendus kasutades programmeerimiskeelt PHP ja andmebaasisüsteemi Oracle. Tegemist on näitega registri tüüpi infosüsteemi tarkvarast. See tarkvara pakub vorme/lehekülgi andmebaasis olevate andmete kasutamiseks ning andmebaasi komponent on sellises tarkvaras väga oluline.

PHP on üks populaarsemaid veebiarenduseks kasutatavaid programmeerimiskeeli (BuiltWith Pty Ltd 2015). Selle paigaldamine on lihtne ja kasutamiseks piisab sellest, kui faile majutavas arvutis on installeeritud lisaks PHP-le ka mõni serveritarkvara (nt Apache). PHP on interpreteeritav, s.t kirjutatud kood ei nõua eelnevat kompileerimist, vaid seda tehakse igal käivitamisel uuesti. See võimaldab näiteks kiiret tagasisidet koodis tekkinud vigade osas, samas võib mõne programmi kompileerimine võtta rohkem aega kui selle käivitamine (Hudson 2015).

Rakenduse loomiseks kasutati programmeerimiskeelel PHP põhinevat rakendusraamistikku CodeIgniter. Raamistikku otsustati kasutada prototüüpimise kiirendamiseks, seega sobiva raamistiku valimisel oli oluline, et see oleks kiiresti omandatav ning sisaldaks arendamist lihtsustavaid lahendusi nagu andmebaasiühenduste loomine. CodeIgniteri valimisel mängis rolli asjaolu, et selle kasutamiseks pole tarvis õppida raamistikule spetsiifilist süntaksi ning töö autoril on sellega varasem kokkupuude.

Antud töös on kasutatud CodeIgniteri versiooni 2.2.1. Töö kirjutamise ajal avaldati ka raamistiku 3. versioon, kuid selles esinevad põhimõttelised muudatused nõudnuks juba valminud rakenduse osa mahukat ümberkirjutamist, mistõttu otsustati jätkata eelmise versiooniga. Kasutatav raamistiku versioon nõuab, et serveris oleks minimaalselt PHP versioon 5.1.6.

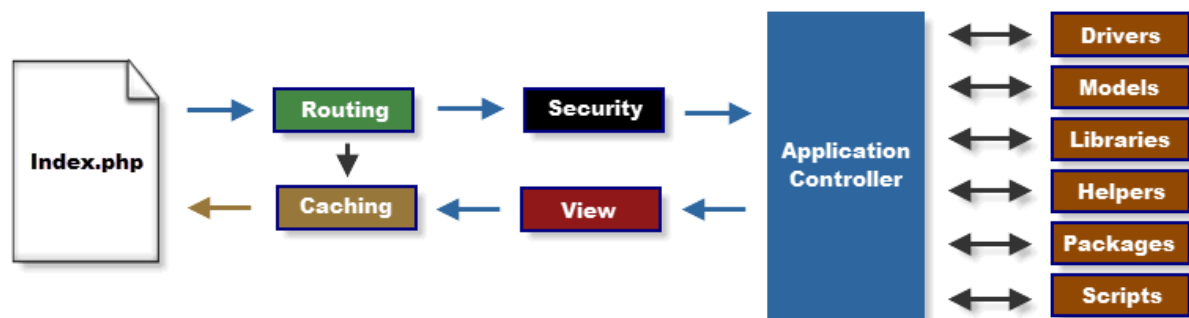
Rakenduse vahendusel hallatavaid andmeid hoitakse Tallinna Tehnikaülikooli serverisse paigaldatud andmebaasisüsteemis Oracle Database 12c Enterprise Edition. PHP kaudu Oracle andmebaasiga suhtlemiseks on vajalik andmebaasidraiver OCI8, mis on serveris kasutusel

olevas PHP versioonis 5.5.3 juba installeeritud. Draiveri kasutamiseks tuleb PHP vastavalt konfigureerida (The PHP Group, 2015). Rakendust töö kirjutamise ajal majutatav apex.ttu.ee server on juba eelnevalt seadistatud.

2.2 Veebirakenduse ülesehitus ja arhitektuur

CodeIgniteri ülesehitus põhineb kaustade ja failide struktuurile. Rakenduse failid on jaotatud kahte peamisesse kausta – *application* (edaspidi rakenduskaust) ja *system* (edaspidi süsteemikaust). Kaustade nimesid ja asukohtasid on võimalik turvalisuse huvides muuta, ent käesolevas rakenduses ei ole seda tehtud.

Rakenduse poole pöördumine käib läbi *index.php* faili, milles on võimalik seadistada rakenduse muutujaid ning mis haldab kogu ülejäänud rakenduse töölepanemist (vt Joonis 2.2.1).



Joonis 2.2.1. Rakenduse vookeem. Application Flow Chart : CodeIgniter User Guide.

index.php faili käivitamise tulemusena laetakse CodeIgniteri tööks vajalikud failid ning sissetulnud päring suunatakse *Router* klassi, mis otsustab, mida sissetulnud HTTP päringuga ette võtta. Kui vastavale päringule leitakse puhvrists sobiv vaste, tagastatakse see, vastasel juhul filtreeritakse päring turvalisuse huvides ning saadetakse *Router* klassis tuvastatud kontrollerrisse. Kontrolleris hallatakse draiverite, mudelite, teekide jm abifunktsioonide laadimist, sejärel tagastatakse kontrolleri töö tulemus vaatesse, mis vastava seadistuse korral puhverdatakse ja saadetakse brauserisse.

2.2.1 CodeIgniteri seadistamine

Rakendusepõhised seadistused tehakse rakenduskaustas asuvas *config* kaustas olevates failides.

Andmebaasi kasutamise seotud muutujaid hoitakse *database.php* failis, kus seadistatakse andmebaasi asukoht, kasutajanimi, parool, kasutatav andmebaasidraiver jne.

CodeIgniter kasutab dünaamilist laadimist – rakenduse komponendid laetakse nende küsimisel. Vähendamaks vajadust tihti kasutatavaid komponente igas kontrollis uuesti laadida, on võimalik *config* kaustas asuvas *autoload.php* failis seadistada, millised komponendid laetakse igal päringul automaatselt. Käesolevas rakenduses laetakse automaatselt andmebaasi- ja sessioonihaldusega tegelevad teegid ning internetiaadresside ja vormide kuvamisega seotud sisseehitatud abifailid, lisaks ka isekirjutatud abifunktsioon kontrollimaks, kas kasutaja on sisse logitud. Erinevalt ülejäänud CodeIgniterist ei ole abifunktsioonid kirjutatud objektorienteeritult, vaid on tavalised PHP funktsioonid.

2.2.2 CodeIgniteris rakenduse kirjutamine

CodeIgniter kasutab Model-View-Controller (MVC) disainimustrit. MVC mustri rakendamisel jaotatakse rakendus kolmeks osas – mudelid, vaated ja kontrollid – ning iga osa tegeleb omaette ülesannetega. Selline jaotus aitab vähendada korduva koodi kirjutamist ning aitab rakendust paremini struktureerida.

2.2.2.1 Mudelid

Mudelid tegelevad andmete haldamisega ning kõik andmebaasipäringud tehakse mudeliklassides. CodeIgniteri mudeleid hoitakse rakenduskaustas asuvas *models* kaustas ja nimetatakse *<mudeli_nimi>_model.php* skeemi järgi. Siin ja edaspidi tähistab nurksulgude (< >) vahel olev tekst kohahoidjat vastavalt tekstile – nt antud juhul mudel nimega *studyplan* asub failis *studyplan_model.php*.

Rakenduses kasutatakse andmete pärimiseks peamiselt CodeIgniteris sisalduvat Active Record klassi. Antud juhul pole siiski tegu objekt-relatsioonilise vastendamise puhul kasutatava lähenemisega, kus andmebaasi tabelid ja tabelite read on vastavuses klasside ja objektidega, vaid abiklassiga, mis lihtsustab andmebaasipäringute tegemist.

Active Recordi kasutamine muudab andmebaasipäringute tegemise sõltumatuks kasutatavast andmebaasisüsteemist, kuna päringu koostab CodeIgniter ise vastavalt kasutuses olevale andmebaasidraiverile. Samuti on Active Recordi abil koostatud päringud ohutumad, kuna nende sisu kontrollitakse enne täitmist SQL süstimise takistamiseks.

Active Recordi abil kirjutatud päringud on SQL klauslite järgi juppideks jaotatud. Järgnevalt analüüsitakse päringut, millega küsitakse andmebaasist nende õppeainete nimekiri, mida tudeng sel semestril endale deklareerida saab.

```
$this->db->select('ainekood, aine_nimetus, ainepunkte,  
aine_kirjeldus, kokku_kohti, reg_arv');
```

Selle lausega defineeritakse SQL SELECT klausli sisu, ehk veerud, mida andmebaasist päritakse.

```
$this->db->distinct('ainekood');
```

Selle lausega eemaldatakse kordused veerust *ainekood*. See on tarvilik, kuna kasutatavas vaates on iga aine ja õppejõu kombinatsiooni kohta oma rida.

```
if ($exclude_list) {  
    $this->db->where_not_in('ainekood', $exclude_list);  
}
```

Päringut teostavale funktsioonile on võimalik kaasa anda nimekiri ainetest, mille tudeng on juba õpingukavasse lisanud. Selle koodiploki juures tuleb välja üks Active Recordi kasutamise eelis – ainult juhul, kui see nimekiri on kaasa antud, lisatakse päringule WHERE NOT IN klausel, millega välistatakse neile ainetele vastavad read päringu tulemusest. Selline lähenemine on programmiline ja hoiab koodi puhtama ning arusaadavamana, vältides SQL lausete koostamist sõnede liitmise teel.

```
$query = $this->db->get('AINED_AKT_SEMESTRIL');
```

Päringu tulemus küsitakse vaatest AINED_AKT_SEMESTRIL muutujasse \$query. Päringule võib FROM klausli ka eraldi lisada, ent selle kasutamine on samaväärne ülaltoodud süntaksiga.

Kui välja jäetavate ainete nimekirja ei ole kaasa antud, moodustatakse selline päring:

```
SELECT DISTINCT ainekood, aine_nimetus, ainepunkte, aine_kirjeldus,  
kokku_kohti, reg_arv FROM AINED_AKT_SEMESTRIL;
```

Kui aga kaasaantavas nimekirjas sisalduvad näiteks ained koodidega IDU0080 ja IDU0220, on päring selline:

```
SELECT DISTINCT ainekood, aine_nimetus, ainepunkte, aine_kirjeldus,  
kokku_kohti, reg_arv FROM AINED_AKT_SEMESTRIL WHERE ainekood NOT IN  
( 'IDU0080', 'IDU0220' );
```

Keerulisemate päringute loomiseks on võimalik SQLi ka ise koostada. Päringusõne koostamisel tuleb parameetrid asendada küsimärkidega, ning päringu teostamisel lisaks sõnele kaasa anda

massiiv väärtustega, mis sõne algusest vastavalt järjekorrale massiivi elementidega asendatakse. Ka neid väärtuseid kontrollitakse SQL süstimise eest.

Kõik andmemuudatused on realiseeritud andmebaasirutiinidega. CodeIgniteri Active Recordil ei ole otsest tuge Oracle andmebaasirutiinide väljakutsumiseks, mistõttu kasutatakse andmebaasirutiinide käivitamiseks PHP-le lisatud OCI8 draiverit. Draiveris sisalduvad funktsioonid võimaldavad rutiinide väljakutsel parameetrid asendada kohahoidjatega, mis hiljem seotakse PHP muutujatega.

Rakenduses on viis mudelit. Mudelid on koostatud nii, et sarnaste andmetega tegelevad funktsioonid on ühes klassis (vt Tabel 2.2.1).

Tabel 2.2.1. Rakenduses olevad mudelid.

Mudeli nimi	Mudeli valdkond
Membership	Tegeleb kasutajate autentimisega.
Studyplan	Tegeleb õpingukavade andmete pärimise, muutmise ja kustutamisega.
Subject	Tegeleb õppeainete ja õppekavamoodulite andmete pärimisega.
Teacher	Tegeleb õppejõudude andmete pärimisega.
User	Tegeleb tudengi andmete pärimisega.

2.2.2.2 Vaated

Vaated tegelevad andmete kuvamisega. Vaateid hoitakse rakenduskaustas asuvas *views* kaustas ja paremaks haldamiseks loodud alamkaustades. Vaadete puhul ei ole nende nimetamisel ja failipuu struktuuril raamistikust tulenevaid piiranguid. Vaateid laetakse kontrollierist kasutades suhtelist rada rakenduskaustas asuva *views* kausta suhtes.

Selles rakenduses on kasutatud CodeIgniteris sisalduvat võimalust vaateid üksteise sisse põimida, et luua korduvkasutatavaid malle. Kõik lehtede kuvamised tehakse läbi *includes* kaustas asuva *template.php* mallivaate, mis laeb esmalt lehe päise. Päises avatakse HTML ning laetakse omakorda eraldi vaatest kõik rakenduse toimimiseks vajalikud CSS ja JavaScripti

failid. Seejärel laeb mall menüü vastavalt sellele, kas kasutaja on sisse logitud või mitte, lehe sisu ning jaluse (Stevens 2012).

Vaadetele saab kontrolleriist kaasa anda vaates kuvatavad andmed. Kuigi CodeIgniteril on ka sisseehitatud mallimootor, mille abil saab kaasa antud muutujaid lihtsalt kuvada, on võimalik kasutada ka tavalist PHP süntaksi muutujate trükkimiseks. Käesolevas rakenduses on valitud viimane variant, kuna see pakub muutujate töötlemiseks vaates suuremat vabadust.

2.2.2.3 Kontrollerid

Kontrollerid haldavad rakenduse tööd. CodeIgniteri rakenduse poole pöördudes tehakse URLi alusel kindlaks, millise kontrolleri poole pöördatakse – vaikimisi seadistuses on aadressi ülesehitus järgnev *index.php/<kontroller>/<meetod>/<parameetrid>*. Aadresse haldav *Router* klass kutsub rakenduskaustas asuvast *controllers* kaustast vastavalt aadressis sisalduvatele kontrolleri ja meetodi nimedele välja vastava nimega klassis sisalduva meetodi. Selline lähenemine teeb kontrolleri kasutamise lihtsaks ja puudub vajadus vastendada aadresse meetoditega. Pöördudes kontrolleri poole ilma meetodit ja parameetreid täpsustamata laetakse automaatselt kontrolleri *index*-nimeline meetod.

Sisseehitatud klassidel on kasutusel eesliide *CI_*. Nendele funktsionaalsuse lisamiseks on võimalik baasklasse laiendada, lisades rakenduskausta *core* kausta faile, millel on kasutusel laiendatud klasside eesliide, vaikimisi *MY_*.

Eraldamaks sisselogitud kasutajatele kättesaadavat funktsionaalsust kogu rakendusest, on rakenduses laiendatud baaskontrollerit selliselt, et kontrolleri laadimisel kontrollitakse, kas kasutaja on sisse logitud või mitte. Seda kontrollerit kasutatakse omakorda baaskontrollerina kõikide vaadete jaoks, millel kuvatakse kasutajakohast andmebaasist pärit infot.

Kõik rakenduse tööks vajalikud komponendid, mida ei ole eelnevalt *autoload.php* abil laetud, laetakse kontrolleri meetodites. Antud rakenduses on tegevused jaotatud kontrolleritesse sisu järgi ja iga kontroller tegeleb omaette valdkonnaga.

Rakenduses on seitse kontrollerit (vt Tabel 2.2.2):

Tabel 2.2.2. Rakenduses olevad kontrollid.

Kontrolleri nimi	Kontrolleri valdkond
Home	Tegeleb sisseloginud kasutajate andmete kuvamisega.
Front	Tegeleb avalehe kuvamisega.
Login	Tegeleb kasutajate autentimisega.
Logout	Tegeleb kasutajate välja logimise ja sessioonimuutujate kustutamisega.
Studyplan	Tegeleb õpingukavadega.
Subjects	Tegeleb ainetega.
Teachers	Tegeleb õppejõududega.

Kuigi veebiliides kasutab AJAX tehnoloogiat, ei ole rakenduses eraldi kontrollit AJAX pöördumiste haldamiseks – seda selleks, et üks kontrollid moodustaks loogilise terviku ja kõik antud objektiga seotud meetodid oleksid koos. AJAX pöördumiste tõstmine eraldi kontrollidesse muudaks tegevuste haldamise keerulisemaks.

3. Veebiliides

Käesolev peatükk kirjeldab veebiliidese loomiseks kasutatud vahendeid, veebiliidese kavandamise protsessi ning selle realiseeritavat funktsionaalsust.

3.1 Veebiliidese loomise vahendid

Rakenduse kasutajaliides on üles ehitatud kasutajaliidese raamistiku UIkit abil. UIkit on MIT litsentsi alusel kasutatav kogu HTML, CSS ja JS komponentidest, mis lihtsustab veebiliidese loomist. Raamistikus on kujundatud valdav enamus HTMLi elementidest, ühtlustamaks veebilehitsejate vahelisi vaikumisi kujunduse erisusi.

UIkiti kujundusfailid on kirjutatud Lessi abil. Less on CSS-keele eeltöötlusvahend, mis lisab CSSile täiendavaid võimalusi, nt muutujate ja funktsioonide toe ning võimaldab kujundusreegleid hierarhiliselt üksteise sisse kirjutada, muutes koodi kirjutamise ja lugemise selgemaks ja kiiremaks. Less failid tuleb eelnevalt kompileerida, mille tulemusel genereeritakse automaatselt vajalikud CSS failid. Rakenduse tarbeks kirjutatud kujundusfailid on samuti kirjutatud Lessis.

UIkitis on sisseehitatud reageeriva disaini tugi lihtsustamaks selliste veebilehtede loomist, mis kohanduvad külastaja seadmele. Käesolevas projektis ei ole seda tuge kasutatud, kuna autori hinnangul kasutab valdav enamus tudengeid õpingukavade koostamiseks ja esitamiseks tavalist laua- või sülearvutit, mitte mõnd nutiseadet.

Kasutajaliidese funktsionaalsuse annab JavaScripti rakendusraamistik AngularJS. Tegu on MIT litsentsi alusel avaldatud raamistikuga, mille arendamist haldab Google. Raamistiku toimimise aluseks on tavalise HTML lehe sisse lisatud „direktiivid“, mida AngularJS oskab tõlgendada ja siduda JavaScriptis olevate muutujate ning funktsioonidega. AngularJS on mõeldud veebirakenduste kirjutamise lihtsustamiseks, peamiselt kahepoolse andmesidumise (*two-way data binding*) kasutuselevõtu abil. Üks peamisi AngularJSi eesmärke on eraldada rakendusloogika dokumendiobjektide mudeli (DOMi) muutmisest – see väljendub näiteks selles, et muutes mõne AngularJS abil DOM elemendiga seotud muutuja väärtust, ei tule DOM elementi omaette muuta (nagu näiteks jQuery puhul), vaid muutus toimub automaatselt. Sõltuvalt kuvatavate andmete keerukusest ja struktuurist võib see märgatavalt vähendada kirjutatava koodi hulka.

Realiseeritud kasutajaliideses kasutatakse andmevahetuseks AJAX tehnoloogiat. AJAX ehk *Asynchronous Javascript and XML* on lähenemine, milles kasutatakse JavaScriptis sisalduvat XMLHttpRequest klassi pöördumaks veebilehitsejast serveri poole nii-öelda „taustal“, ehk ilma tervet veebilehte taaslaadimata. AJAXi kasutamine vähendab kasutajate ooteaega, kuna serverist päritakse terve veebilehe asemel ainult vajaminevaid andmeid. Kombineerides AJAXi kasutamise AngularJS raamistikuga, saab päritud andmeid kuvada pannes need HTMLis kahepoolselt seotud JavaScripti muutujasse ning AngularJS muudab DOMi automaatselt.

Veebiliidest on testitud järgnevate veebilehitsejatega: Google Chrome 43.0.2357.81 m, Mozilla Firefox 38.0.1, Microsoft Internet Explorer 11.0.9600.17801 ja Opera 29.0.1795.60. Liides käitub kõikides katsetatud veebilehitsejates vastavalt ootustele.

3.2 Veebiliidese kavandamine

Veebipõhise kasutajaliidese kavandamisel lähtusin asjaolust, et tegu on infosüsteemi allsüsteemiga, mille kasutajad on tudengid, sh esmakursuslased, kellel pole varasemat kokkupuudet ülikooli infosüsteemide ja tööprotsessidega. Infosüsteemi kasutamine on ülikoolis õppimiseks kohustuslik, kuna allsüsteem on ainus võimalus õpingukava koostamiseks ja õpitavate ainete deklareerimiseks. Seega on kasutajaliidese kujundamisel võetud eesmärgiks luua võimalikult lihtne ja kergesti kasutatav keskkond, mis muudaks kasutamise kiireks ja efektiivseks.

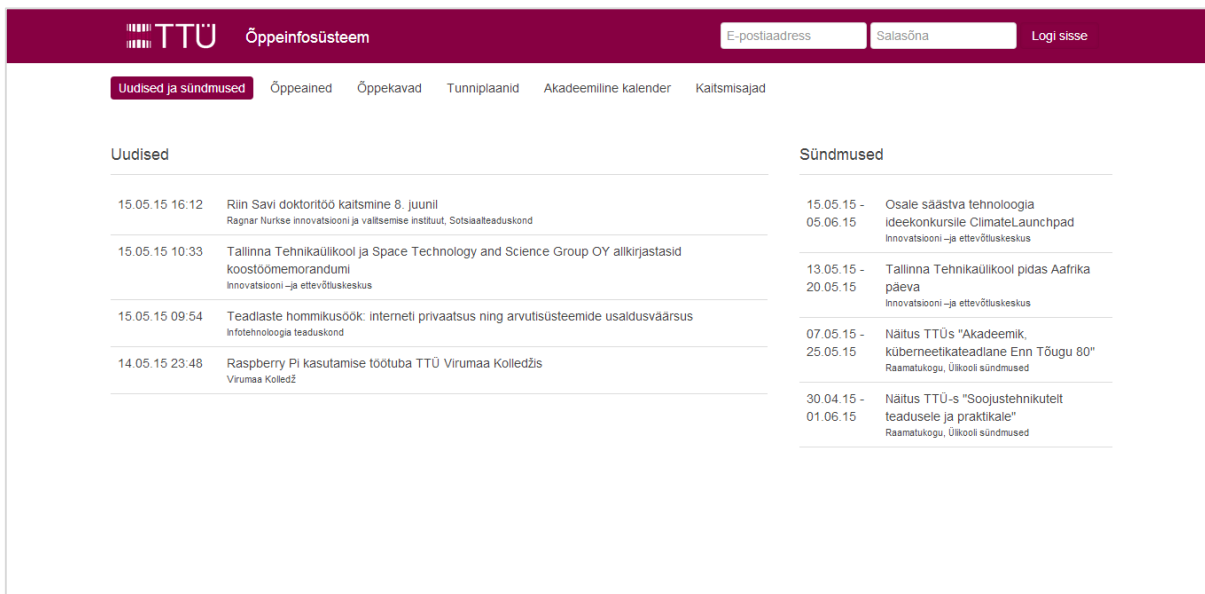
Eesmärgi täitmiseks tutvus autor mitmete kasutajaliidese ja kasutajakogemuse ekspertide koostatud materjalidega. Valdavalt on veebiliidese kujundamisel lähtutud Ian Grahami (2002) juhendamisel koostatud disainimustrite keelest *wu*. Lisaks on kasutatud põhimõtteid Bruce Tognazzini (2014) artiklist ning Majandus- ja Kommunikatsiooniministeriumi Riigi Infosüsteemide Osakonna tellitud uuringust „Kasutajakeskse veebi lehekülgede disain“ (edaspidi RISO uuring), mille teostas Trinidad Consulting OÜ (2009).

3.3 Veebiliidese funktsionaalsus

Järgnevalt on kirjeldatud realiseeritud veebiliidest ja selle funktsionaalsust vaadete kaupa.

3.3.1 Veebiliidese avamine

Rakendus on kättesaadav aadressil <http://apex.ttu.ee/opingukava>. Kasutajale avaneb õppeinfosüsteemi avaleht, kus on kuvatud ülikooli värsked uudised ja sündmuste nimekiri (vt Joonis 3.3.1). Ülaloleval navigatsiooniribal on võimalik sisse logida. Navigatsiooniriba all on alammenüü, milles on viited jaotistele, mis peaksid olema kättesaadavad õppeinfosüsteemi sisse logimata.



Uudised	Sündmused
15.05.15 16:12 Riin Savi doktoritöö kaitsmine 8. juunil Ragnar Nurkse innovatsiooni ja valitsemise instituut, Sotsiaalteaduskond	15.05.15 - Osale säästva tehnoloogia 05.06.15 ideekonkursile ClimateLaunchpad Innovatsiooni- ja ettevõtluskeskus
15.05.15 10:33 Tallinna Tehnikaülikool ja Space Technology and Science Group OY allkirjastasid koostöömemorandumi Innovatsiooni- ja ettevõtluskeskus	13.05.15 - Tallinna Tehnikaülikool pidas Aafrika 20.05.15 päeva Innovatsiooni- ja ettevõtluskeskus
15.05.15 09:54 Teadlaste hommikusöök: interneti privaatsus ning arvutisüsteemide usaldusväärsus Infotehnoloogia teaduskond	07.05.15 - Näitus TTÜs "Akadeemik 25.05.15 kübermeetikateadlane Enn Tõugu 80" Raamatukogu, Ülikooli sündmused
14.05.15 23:48 Raspberry Pi kasutamise töötuba TTÜ Virumaa Kolledžis Virumaa Kolledž	30.04.15 - Näitus TTÜ-s "Soojustehnikuteilt 01.06.15 teadusele ja praktikale" Raamatukogu, Ülikooli sündmused

Joonis 3.3.1. Realiseeritud kasutajaliidese avaleht.

3.3.2 Sisselogimisvaade

Juhul, kui kasutaja eksib sisselogimisvormi täitmisel, suunatakse ta eraldi sisselogimisvaatesse (vt Joonis 3.3.2). Rakenduses on loodud testkasutaja kasutajanimelga `t990999@ttu.ee` ja parooliga `1a2b3c`

TTU Õppeinfosüsteem

E-postiaadress Salasõna Logi sisse

Palun taida kõik väljad korrektselt!

E-postiaadress

Salasõna

Logi sisse

Joonis 3.3.2. Sisselogimisvaade

Sõltuvalt esinenud veast kuvatakse ka vastav veateade. Kasutajale tagasiside andmine on oluline aitamaks mõista, miks süsteem ei käitunud vastavalt kasutaja ootustele (antud juhul – ei loginud kasutajat süsteemi) (*wu* muster Tagasiside (Feedback, 41), Graham 2002). Kohandatud veateate kasutamine aitab kasutajal tekkinud viga mõista ning seda edaspidi vältida (*wu* muster Kontekstitundlikud abiteated (Context-sensitive help, 17), Graham 2002).

Veateade kuvatakse vormi kohal (vt Joonis 3.3.3), kuna tegu ei ole veateatega konkreetse välja kohta, vaid vormi täitmise kohta üldiselt. Teate kujundamisel on kasutatud punast kui ohtu või viga tähistavat värvi ning märgatavuse tõstmiseks on lisatud väike animatsioon (RISO uuringu põhimõte 6.1.14 Veateade esita kasutajale nähtavalt, Trinidad Consulting OÜ 2009).

The image shows a login form with a red error message at the top: "Sisestatud kasutajanimi ja parool ei klapi. Palun kontrolli üle!". Below the message are two input fields: the first contains the email address "t990999@ttu.ee" and the second is labeled "Salasõna". At the bottom of the form is a dark red button with the text "Logi sisse".

Joonis 3.3.3. Veateade, kui kasutaja sisestatud andmetele ei leitud andmebaasist vastet.

Juhul, kui kasutaja täitis mõlemad väljad korrektselt, ent sisestatud väljadele vastavat kasutajat süsteemist ei leitud, kuvatakse vastavasisuline teade. Veateates ei täpsustata, kas süsteemis eksisteerib sisestatud e-postiaadressiga kasutaja või mitte, vältimaks turvariski, kus katsetamise teel on võimalik kasutaja olemasolu teada saada. Kasutajakogemuse huvides kuvatakse sisestatud e-postiaadressi, et selles esinenud võimalikud vead oleks visuaalselt tuvastatavad. (põhimõtte „Ensure that users never lose their work“, Tognazzini 2014). Salasõna puhul ei oleks sellest kasu, kuna salasõnaväljale sisestatud ei ole võimalik näha.

Kasutaja suunatakse sisselogimisvaatesse ka juhul, kui ta üritab sisse logimata ligi pääseda mõnele lehele, mis nõuab sisselogimist ehk antud realisatsioonis mõnele MY_Controlleril põhinevale lehele. Sellisel juhul jätab rakendus meelde, millisele lehele kasutaja üritas ligi pääseda ning peale edukat sisselogimist suunatakse kasutaja soovitud lehele edasi.

CodeIgniteril on sisseehitatud sessiooniklass, mille abil on võimalik talletada infot jooksva sessiooni kohta kasutaja küpsistesse. Edukal sisselogimisel salvestatakse sessioonimuutujatena tudengi isikukood, matriklinumber, õppekava tähis, nimi ja märg, et kasutaja on sisse logitud.

3.3.3 Kasutaja avakuva

Juhul, kui kasutaja ei üritanud süsteemile ligi pääseda mõne konkreetse sisselogimist nõudva vaate kaudu, suunatakse ta peale edukat autentimist kasutaja avakuvale. Avakuva on

ülesehitatud n-ö. *dashboard* mudelile, mis võimaldab kiiret ülevaadet peamisest õppetööga seotud informatsioonist (Nielsen 2008) (vt Joonis 3.3.4).

The screenshot shows a student dashboard for TTU. At the top, there is a navigation bar with the TTU logo, 'Õpingukavad' (Courses), 'Sooritud' (Completed), and user information for 'Andres Mets' (ID: 990999IABB) with a 'Logi välja' (Logout) button. A red warning banner at the top states: 'Sul puudub selleks semestriks õpingukava!' (You are missing a course plan for this semester!) with an 'Alusta' (Start) button.

The dashboard is organized into six main widgets:

- Teated (3):** A list of notifications. The first one is from 'Jaagup Irve' dated 01.04.2015 17:14, regarding a course plan. The second is from 'Merike Siidoja' dated 03.03.2015 12:00, about a seminar. The third is from 'Merike Siidoja' dated 05.02.2015 11:52, about a candidate selection process. A link 'Vaata vanemaid teateid' is at the bottom.
- Viimati saadud hinded:** A message stating 'Sa pole sel semestril veel hindeid saanud.' (You have not received grades yet this semester).
- Õpingukava:** A message stating 'Sa pole selle semestri õpingukava alustanud.' (You have not started your course plan for this semester.) with a link 'Vaata õpingukavasid'.
- Eelseisvad teadmiste kontrollid:** A message stating 'Sa pole veel ühelegi teadmiste kontrollile registreerunud.' (You have not registered for any knowledge tests yet).
- Ülesanded:** A message stating 'Sul pole veel ühtegi ülesannet.' (You have no assignments yet).
- Andres Mets:** A profile card showing 'Teaduskond: Infotehnoloogia teaduskond', 'Õppekava: Informaatika', 'Matrikinumber: 990999', and 'Õpperühm: IAPB62'. A link 'Vaata täpsemalt' is at the bottom.

Joonis 3.3.4. Kasutaja avakuva

Avakuval on kuus kategooriat – teated, viimati saadud hinded, õppeinfo, eelseisvad teadmiste kontrollid, ülesanded ja tudengi info. Kategooriad on valinud töö autor koostöös kaastudengitega selliselt, et oluline informatsioon oleks õppeinfosüsteemist võimalikult kiiresti leitav.

Kuna iga semestri alguses tuleb tudengil koostada õpingukava, kuvatakse enne punase joone päeva avakuval ka tudengi selle semestri õpingukava staatust. Teate eesmärk on juhtida tudengi tähelepanu tegutsemist vajavale ülesandele (põhimõte „Bring to the user all the information and tools needed for each step of the process“, Tognazzini 2014). Teates on kasutatud õpingukava seisundi tähistamiseks punast, kollast ja rohelist värvi vastavalt alustamata, esitamata ja esitatud õpingukava tarvis. Samuti on teates link õpingukavade vaatesse liikumiseks, mis aitab ülesandega seotud töövoogu lihtsustada.

Avakuval on sarnaselt praeguse õppeinfosüsteemi esilehega kuvatud teated, viimati saadud hinded ja eelseisvad teadmiste kontrollid.

Praeguse õppeinfosüsteemi esilehe parempoolses tulbas kuvatav õpingukava/tunniplaani info on tõstetud õppeinfo jaotisesse. Kuni õpingukava esitamiseni kuvatakse seal õpingukavasse valitud aineid, peale esitamist aga kuvatakse tudengi rühma alusel tema tunniplaani. Kuna käesoleva allsüsteemi raames andmebaasis sellekohaseid andmeid ei ole, kuvatakse seal näidisinfort.

Uue jaotisena on avakuval ülesanded. Selle jaotise lisamise taga on autori isiklik lootus, et ühel päeval on õppeinfosüsteem TTÜ ainus õpinguid puudutav infosüsteem, mis asendaks kõiki ülejäänud õppekeskkondi (nt erinevad Moodle keskkonnad, Maurus). Selline lähenemine aitaks tudengi hinnangul vähendada andmete haldamisega seotud ajakulu ning koondaks kõik õppetöök vajaliku ühte keskkonda.

Avakuval on ka jaotis tudengi peamise infoga (vt Joonis 3.3.5).

3.3.4 Navigatsioon



Joonis 3.3.5. Sisselogitud kasutaja navigatsiooniriba.

Sisselogitud kasutaja saab navigeerida lehe ülaservas oleva navigatsiooniriba abil. Lehe ülal vasakus servas on TTÜ logo, mille abil saab naasta kasutaja avakuvale (*wu* muster Saidi logo ülal vasakul (Site logo at top left, 24), Graham 2002). Sellele järgnevad peamised kategooriad – käesoleva prototüübi puhul on kasutatav vaid Õpingukavade link. Navigatsiooniriba all kuvatakse alammenüüd, mille abil saab valitud peakategooria raames liikuda.

Navigatsiooniriba paremas servas (vt Joonis 3.3.5) on kuvatud sisselogitud kasutaja nimi ja üliõpilaskood, millele klõpsates liigutakse kasutaja andmete vaatesse. Neile järgneb lugemata teadete nupp, millele hiirt viies avaneb rippmenüü teadete eelvaatega. Kõige paremas servas on nupp süsteemist välja logimiseks.

3.3.5 Õpingukavade üldvaade

TTU Õpingukavad Soortused Andres Mets 990999/ABB 3 Logi välja

Minu õpingukavad Tüüpõpingukavad Moodulid

Minu õpingukavad

2014/2015 sügis mitteaktuaalne 0 ainet Vaata

Sul puudub selleks semestriks õpingukava.

- Kui soovid õpingukava algusest peale ise koostada, loo uus tühi õpingukava.
- Kui soovid aluseks võtta mõne tüüpõpingukava, leiad selle võimaluse tüüpõpingukavade loetelust.

+ Loo tühi õpingukava + Vaata tüüpõpingukavasid

Joonis 3.3.6. Õpingukavade üldvaade

Õpingukavade üldvaatesse saab ülemise navigatsiooniriba kaudu või klõpsates alustamata õpingukava teatel olevas lingis avakuval. Üldvaate laadimisel sooritatakse AJAXi päring küsimaks kõiki tudengi õpingukavasid.

Kui tudengil pole käesolevaks semestriks õpingukava, kuvatakse vastavasisulist teadet, milles on ka juhend, kuidas jätkata (*wu* muster Kontekstitundlikud abiteated (Context-sensitive help, 17), Graham 2002; põhimõtte „Bring to the user all the information and tools needed for each step of the process“, Tognazzini 2014). Veateade on sõnastatud kasutajale arusaadavalt ja konteksti sobivalt (*wu* muster Sobiv sõnastus (Acceptable wording, 50), Graham 2002).

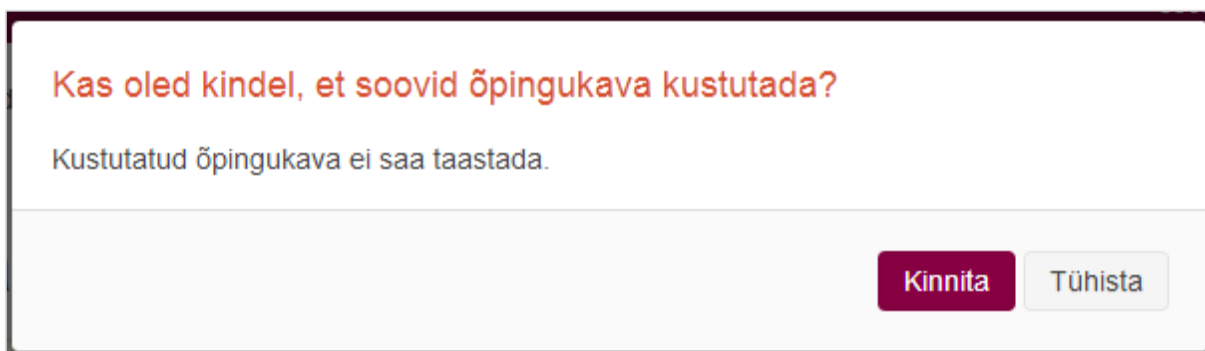
Õpingukava alustamiseks on kaks võimalust: luua uus tühi õpingukava või liikuda tüüpõpingukavade vaatesse, kus on võimalus soovitud semestri tüüpõpingukava aluseks võtta. Nende tegevuste sooritamiseks on teate all kaks nuppu, mida kuvatakse ainult siis, kui tudengil ei ole selleks semestriks õpingukava – vastasel juhul need ainult risustaksid kasutajaliidest (*wu* muster Mis on nähtaval, on kasutamiseks (What You See Is What You Can Use, 70), Graham 2002). Tüüpõpingukavadele on võimalik ligi pääseda ka lehe ülaosas oleva alammenüü abil. Kui tudengil juba on avatud õpingukava, aga ta siiski üritab ligi pääseda õpingukava alustamiseks mõeldud aadressile, suunatakse tudeng oma õpingukava juurde ning kuvatakse vastavasisuline hoiatusteade (vt Joonis 3.3.7).

Üritasid alustada uut õpingukava, ent Sul juba on avatud õpingukava!
Nullist alustamiseks kustuta õpingukavast kõik ained või kustuta õpingukava ja alusta uut.

Joonis 3.3.7. Teade, kui kasutajal on juba avatud õpingukava, ent ta üritab uut luua.

Nimekirjas oleva õpingukava avamiseks saab klõpsata ümbritseval kastil tervikuna, vähendamaks vajadust sihtida täpselt väikese nupu pihta. Selline lähenemine kiirendab kasutamist (Fitts'i seadus, Tognazzini 2014), ühtlasi on väiksem tõenäosus klõpsata kogemata õpingukava juures oleval kustutamise nupul.

Koostamisel olevat õpingukava on võimalik õpingukavade üldvaatest kustutada „Kustuta“ nupu abil. Nupule klõpsates avaneb modaalaken ning ülejäänud leht tumeneb tõmbamaks kasutaja tähelepanu tekkinud aknale (Nielsen 2008). Kuna kustutatud õpingukava taastada ei saa, küsitakse modaalaknas kustutamisele kinnitust, kuna kasutaja võis nupule kogemata klõpsata.

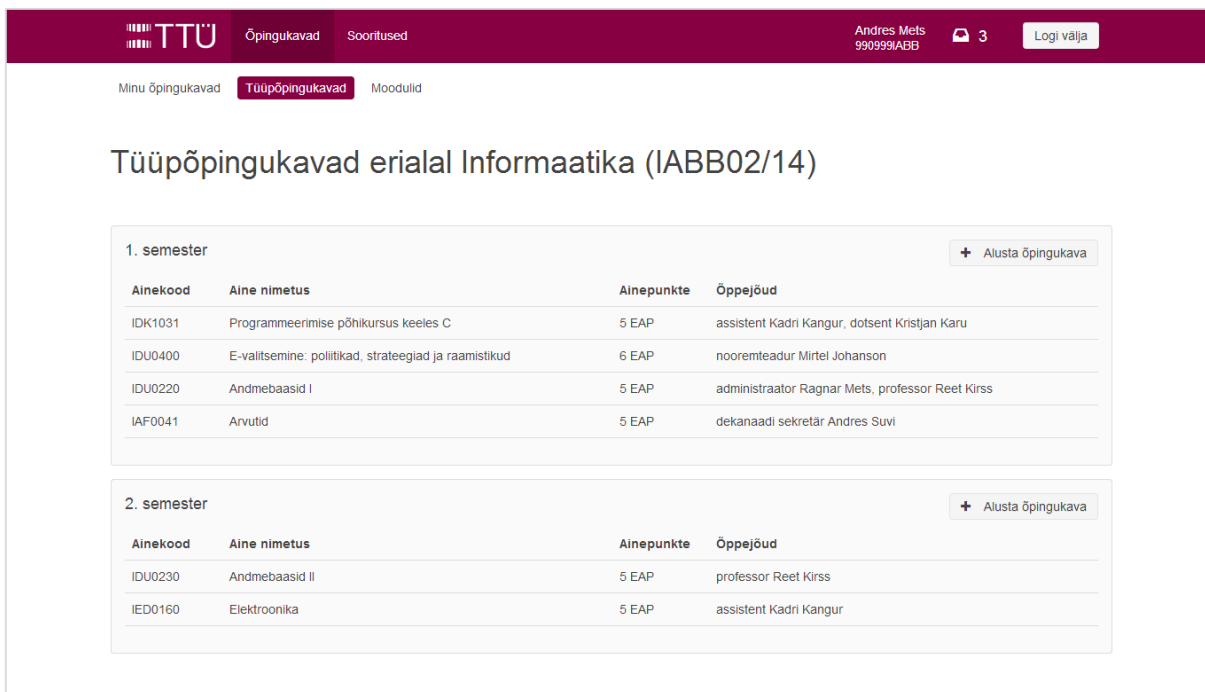


Joonis 3.3.8. Õpingukava kustutamise kinnitust küsiv modaalaken.

Kui tudeng otsustab oma õpingukava kustutada, sooritatakse DELETE-meetodiga AJAXi päring kontrolleri Studyplan vastava funktsiooni pihta. Kuna kontrolleri funktsioonid on kättesaadavad tavaliste internetiaadresside kaudu – antud funktsioon aadressil *index.php/studyplan/delete_studyplan/<õpingukava_id>* – kontrollitakse turvalisuse eesmärgil, kas päring tehakse DELETE meetodiga. Tavalise veebilehitsejaga sellele aadressile minnes (ehk põhimõtteliselt sooritades GET-meetodiga päringu) saab veateate, et kasutajal pole õiguseid antud päringu sooritamiseks – seega ei ole juhuslikult ligipääsul võimalik süsteemis muudatusi teha. Vältimaks aga olukorda, kus päringut kutsutakse välja väljaspool süsteemi, antakse andmebaasirutiinile lisaks kustutatava õpingukava identifikaatorile kaasa ka sisse logitud tudengi matriklinumber, mis päritakse sessiooniandmetest vahetult enne rutiini

käivitamist. Sel viisil on võimalik tagada, et autenditud tudeng saab kustutada vaid endaga seotud õpingukava.

3.3.6 Tüüpõpingukavade vaade



TTU Õpingukavad Sooritused Andres Mets 990999/IABB 3 Logi välja

Minu õpingukavad Tüüpõpingukavad Moodulid

Tüüpõpingukavad erialal Informaatika (IABB02/14)

1. semester [+ Alusta õpingukava](#)

Ainekood	Aine nimetus	Ainepunkte	Õppejõud
IDK1031	Programmeerimise põhikursus keeles C	5 EAP	assistent Kadri Kangur, dotsent Kristjan Karu
IDU0400	E-valitsemine: poliitika, strateegiad ja raamistikud	6 EAP	nooremlehtur Mirtel Johanson
IDU0220	Andmebaasid I	5 EAP	administraator Ragnar Mets, professor Reet Kirss
IAF0041	Arvutid	5 EAP	dekanaadi sekretär Andres Suvi

2. semester [+ Alusta õpingukava](#)

Ainekood	Aine nimetus	Ainepunkte	Õppejõud
IDU0230	Andmebaasid II	5 EAP	professor Reet Kirss
IED0160	Elektronika	5 EAP	assistent Kadri Kangur

Joonis 3.3.9. Tüüpõpingukavade vaade

Tüüpõpingukavade vaates (vt Joonis 3.3.9) kuvatakse infot tudengi õppekavaga seotud tüüpõpingukavade kohta. Juhul, kui tudeng ei ole õpingukava alustanud, kuvatakse iga tüüpõpingukava juures ka nuppu selle alusel uue õpingukava koostamiseks.

Tüüpõpingukavasse kuuluvate ainete kohta näidatakse ainekoodi, aine nimetust, aine sooritamisel saadavate EAPde arvu ning nimekirja õppejõududest, kes käesoleval semestril ainet õpetavad.

3.3.7 Moodulite vaade

TTU Õpingukavad Soortused Andres Mets 990999/IABB 3 Logi välja

Minu õpingukavad Tüüpõpingukavad **Moodulid**

Õppekava: Informaatika (IABB02/14)

31 EAP • 6 ainet

Moodulid

Üldõpe 0 ainet • 0 EAP

Selles moodulis ei ole aineid.

Alusõpe 0 ainet • 0 EAP

Selles moodulis ei ole aineid.

Põhiõpe 5 ainet • 25 EAP

Ainekood	Aine nimetus	Ainepunkte	Semester
IAF0041	Arvutid	5 EAP	1. semester
IDK1031	Programmeerimise põhikursus keeles C	5 EAP	1. semester
IDU0220	Andmebaasid I	5 EAP	1. semester
IDU0230	Andmebaasid II	5 EAP	2. semester
IED0160	Elektronika	5 EAP	2. semester

Eriõpe 1 aine • 6 EAP

Ainekood	Aine nimetus	Ainepunkte	Semester
IDU0400	E-valitsemine: poliitika, strateegiad ja raamistikud	6 EAP	1. semester

Praktika 0 ainet • 0 EAP

Selles moodulis ei ole aineid.

Vabaõpe 0 ainet • 0 EAP

Selles moodulis ei ole aineid.

Lõputöö 0 ainet • 0 EAP

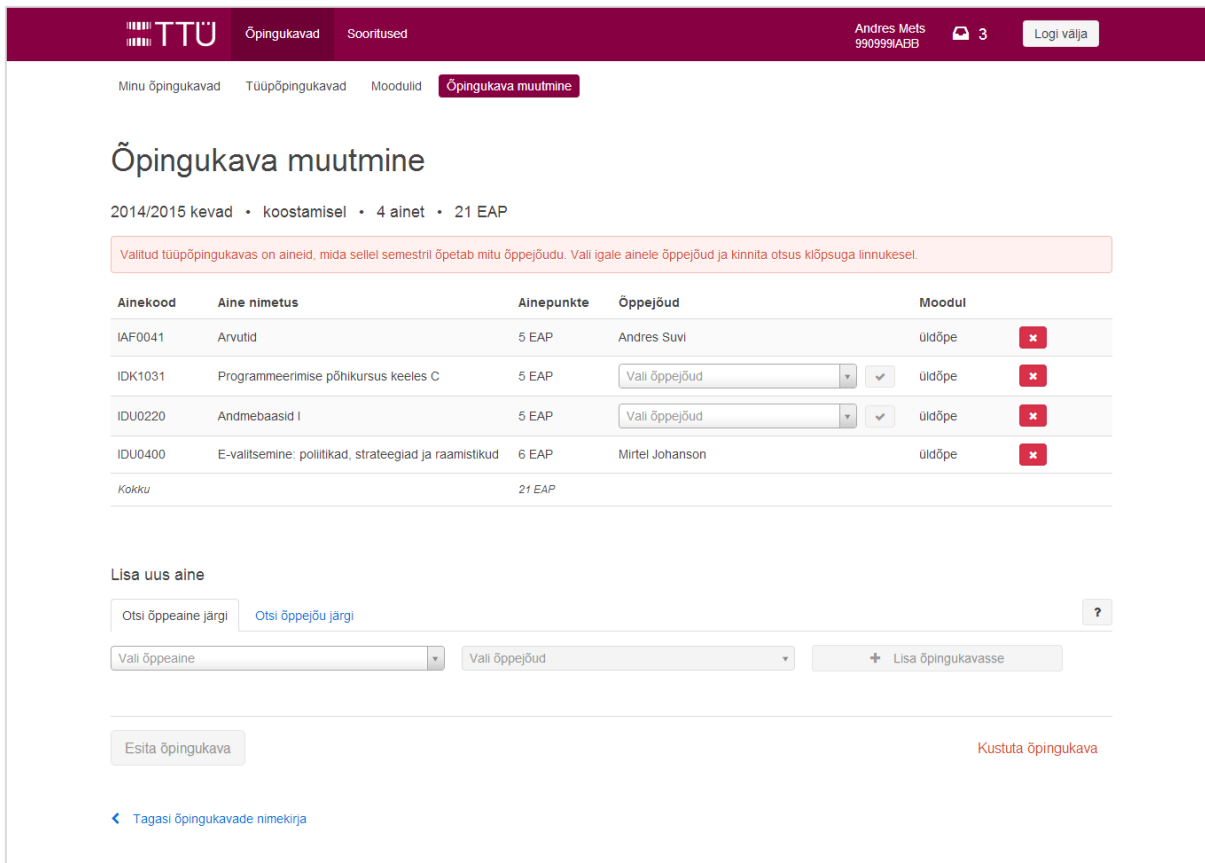
Selles moodulis ei ole aineid.

Joonis 3.3.10. Moodulite vaade.

Moodulite vaates kuvatakse tudengi õppekavas olevaid aineid moodulite kaupa. Lehe alguses kuvatakse kogu õppekava mahtu EAPdes ning õppekavasse kuuluvate ainete arvu, sejärel kuvatakse sama infot iga mooduli kohta eraldi koos nimekirjaga ainetest, mis sellesse moodulisse kuuluvad. Iga aine juures on ära toodud ainekood, aine nimetus, ainepunktide arv ning mitmenda semestri tüüpõpingukavasse aine kuulub.

3.3.8 Õpingukava detailvaade

Kui tudeng otsustab uut õpingukava alustada, suunatakse ta õpingukava detailvaatesse (vt Joonis 3.3.11).



The screenshot shows the 'Õpingukava muutmine' (Course Change) page on the TTU website. The page header includes the TTU logo, navigation tabs for 'Õpingukavad', 'Sooritused', and 'Moodulid', and a user profile for 'Andres Mets' with a 'Logi välja' button. The main content area is titled 'Õpingukava muutmine' and shows the current semester '2014/2015 kevad' with '4 ainet' and '21 EAP'. A red warning box states: 'Valitud tüüpõpingukavas on aineid, mida sellel semestril õpetab mitu õppejõudu. Vali igale ainele õppejõud ja kinnita otsus klõpsuga linnukesel.' Below this is a table of selected courses:

Ainekood	Aine nimetus	Ainepunkte	Õppejõud	Moodul
IAF0041	Arvutid	5 EAP	Andres Suvi	Üldõpe
IDK1031	Programmeerimise põhikursus keeles C	5 EAP	Vali õppejõud	Üldõpe
IDU0220	Andmebaasid I	5 EAP	Vali õppejõud	Üldõpe
IDU0400	E-valitsemine: poliitika, strateegiad ja raamistikud	6 EAP	Mirtei Johanson	Üldõpe
<i>Kokku</i>		<i>21 EAP</i>		

Below the table, there is a section 'Lisa uus aine' with search options 'Otsi õppeaine järgi' and 'Otsi õppejõu järgi'. Below that are dropdown menus for 'Vali õppeaine' and 'Vali õppejõud', and a '+ Lisa õpingukavasse' button. At the bottom, there are buttons for 'Esita õpingukava' and 'Kustuta õpingukava', and a link 'Tagasi õpingukavade nimekirja'.

Joonis 3.3.11. Õpingukava detailvaade peale tüüpõpingukava alustamist.

Õpingukava informatsioon laetakse AJAX päringuga. Esmalt kuvatakse õpingukava semestrit, seisundit, selles olevate ainete hulka ja mitme EAP väärtuses aineid õpingukavas on.

Seejärel kuvatakse õpingukavas olevad ained tabeli kujul. Iga aine kohta näidatakse ainekoodi, nimetust, EAPde hulka, õppejõudu, kellele tudeng aine deklareeris ning millises moodulis see aine tudengi õppekavas on. Tabeli read on loetavuse huvides vahelduvate värvidega.

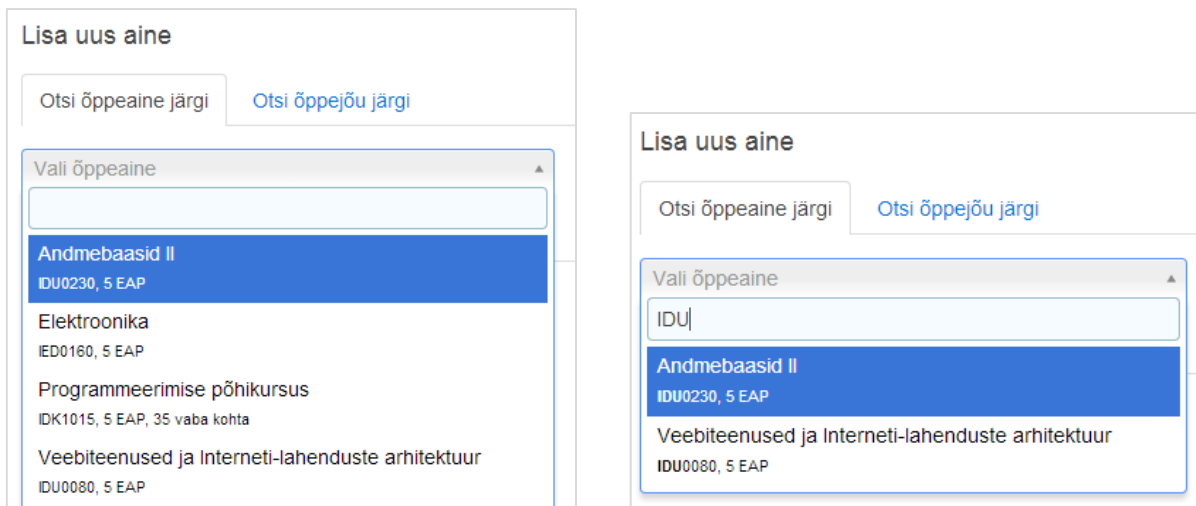
Õpingukavas olevaid aineid on võimalik sorteerida tulba päisel klõpsates. Parasjagu sorteerimisaluseks oleva tulba pealkirja kõrval on nool, mis näitab sorteerimise suunda. Vahetades sorteerimisaluseks olevat tulpa, sorteeritakse tabel esmalt valitud tulba alusel kasvavas järjekorras, iga järgnev klõps valitud tulba päisel vahetab sorteerimise suunda.

Kui tudeng võtab oma õpingukava aluseks mõne tüüpõpingukava, kopeerib süsteem sealolevad ained automaatselt tudengi õpingukavasse. Kui mõnd ainet õpetab antud semestril ainult üks õppejõud, valitakse automaatselt see õppejõud. Kui aga ainet õpetab semestril mitu õppejõudu, lisatakse aine õpingukavasse ilma õppejõuta ning detailvaates kuvatakse vastavasisuline teade ja tudeng peab õppejõu veerus olevast rippmenüüst valima, millise õppejõuga ta soovib ainet sel semestril õppida. Õppejõu valikut saab kinnitada rippmenüü kõrval oleva rohelise nupukesega, mis on klõpsatav ainult siis, kui rippmenüüs on valik tehtud. Peale kinnitamist laetakse õpingukavas olevad ained AJAX päringu abil uuesti. Kui õpingukavas pole enam ühtegi ainet, millel oleks kinnitamata õppejõud, kaob teade ning aktiveerub õpingukava esitamise võimalus.

Juhuks, kui mõnd õppeainet õpetab mitu sama nime ja ametinimetusega õppejõudu, kuvatakse lisaks õppejõu nimele ja ametile rippmenüüs ka õppejõu e-postiaadressi. Kuna rakendusse sisselogimiseks kasutatakse e-postiaadressi ja see on ühtlasi kasutajate tabelis kandidaatvõti, võib selle unikaalsuses kindel olla.

Õpingukavasse saab aineid lisada õpingukava ainete tabeli all oleva vormi kaudu. Lisatavaid aineid saab otsida kahel moel: õppeaine ja õppejõu järgi. Otsingualust saab vahetada sakkide abil. Otsinguvormis on kaks rippmenüüd.

Õppeaine järgi otsides tuleb esmalt valida esimesest rippmenüüst õppeaine, mida tudeng soovib õpingukavasse lisada. Rippmenüüs kuvatakse ainult neid õppeaineid, mida sel semestril õpetatakse ja mida tudeng ei ole juba oma õpingukavasse lisanud. Kui tudeng rippmenüü avab, on menüüelementide kohal otsingulahter, millest saab aine nimetuse ja ainekoodi järgi soovitud ainet otsida. Menüüelemente filtreeritakse automaatselt ja alles jäävad ainult otsingulahtri sisule vastavad ained.

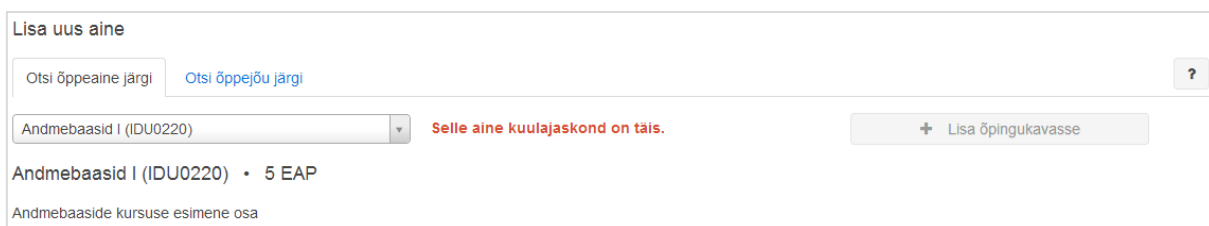


Joonis 3.3.12. Õppeainete rippmenüü vaikimisi ja täidetud otsingulahtriga.

Iga aine kohta näidatakse aine nimetust, ainekoodi ja ainepunktide arvu. Kui ainele on seatud deklaratsioonide arvu piirang, kuvatakse lisaks ka õppeainele jäänud vabade kohtade arvu.

Nimekirjast aine valimisel kuvatakse vormi alla õppeaine info ja kirjeldus, lisaks laetakse AJAX päringuga kõrvalolevasse rippmenüüsse seda ainet õpetavate õppejõudude nimekiri. Kui ainet õpetab ainult üks õppejõud, valitakse ta automaatselt, kuna see aitab kasutaja aega kokku hoida (RISO uuringu põhimõtte 6.1.15 Võimalusel tee kasutaja eest eelvalikud, Trinidad Consulting OÜ 2009). Vastasel juhul peab tudeng valima rippmenüüst õppejõu, kellele ta soovib ainet deklareerida. Ka õppejõudude rippmenüü on filtreeritav. Aine õpingukavasse lisamise nupp aktiveerub alles siis, kui valitud on nii õppeaine kui õppejõud.

Juhul, kui ainele ei ole enam vabu kohti jäänud, kuvatakse ainet siiski õppeainete nimekirjas märkega „0 vaba kohta“. Sellise lähenemise kasuks otsustas autor koos kaastudengitega vältimaks segadust, et soovitud ainet ei ole nimekirjas. Kui tudeng valib nimekirjast sellise aine, kuvatakse õppejõu valiku asemel teadet „Selle aine kuulajaskond on täis.“ Kuna õppejõudu ei saa valida, ei saa ka ainet õpingukavasse lisada (vt Joonis 3.3.13).



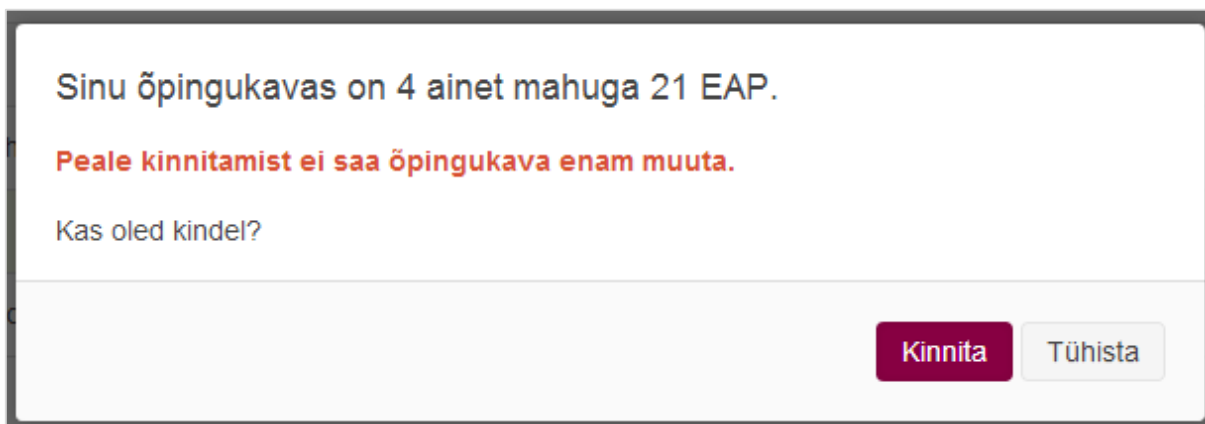
Joonis 3.3.13. Teade, kui tudeng valib õppeaine, mille kuulajaskond on täis.

Õppejõu järgi otsimisel tuleb esimesest rippmenüüst esmalt valida õppejõud, kellele soovitakse ainet deklareerida. Rippmenüüs kuvatakse kõiki õppejõude, kes sel semestril mõnd ainet õpetavad. Valides rippmenüüst õppejõu, laetakse AJAXiga kõrvalolevasse rippmenüüsse kõigi ainete nimekiri, mida valitud õppejõud sel semestril õpetab. Kui tudeng on kõik õpetatavad ained juba omale õpingukavasse lisanud, kuvatakse vastavasisulist teadet. Kui nimekirjas on vaid üks aine, valitakse see automaatselt, vastasel juhul peab tudeng rippmenüüst soovitud aine valima. Aine valimisel kuvatakse vormi all selle aine infot ja kirjeldust ning juhul, kui valitud aine kuulajaskond on täis, ka vastavasisulist teadet. Sel juhul hoolimata sellest, et nii õppejõud kui aine on valitud, ei lase süsteem siiski ainet õpingukavasse lisada. Ka õppejõu järgi otsides on mõlemad mainitud rippmenüüd filtreeritavad.

Kõik ainete lisamise ja kustutamise seotud tegevused salvestatakse automaatselt. Nii aine lisamisel kui eemaldamisel kuvatakse ekraani ülaserivas vastavat teadet. Viimati lisatud ainet kuvatakse rohelisel taustal.

Lehe allservas on võimalik õpingukava esitada ja kustutada. Vasakus lehe servas olev õpingukava esitamise nupp on kuvatud suuremana, kuna tõenäoliselt soovib tudeng koostatud õpingukava pigem esitada kui kustutada. Õpingukava kustutamise nupp on lehe paremas servas ning vähem tähelepanu tõmbava kujundusega, et töövoog loogiline jätkamine paremini silma paistaks (Wroblewski 2005). Juhul, kui tudeng soovib õpingukava kustutada, otsib ta spetsiaalselt õpingukava kustutamiseks mõeldud nuppu.

Mõlemal nupuvajutusel kuvatakse kinnitust küsivat modaalakent, kuna mõlemad tegevused on pöördumatud (vt Joonis 3.3.15).



Joonis 3.3.14. Õpingukava esitamise kinnitust küsiv modaalaken.

Peale õpingukava kinnitamist laetakse õpingukava info AJAX päringu abil uuesti ning õpingukava ei ole võimalik enam muuta.

Õpingukava kustutamise modaalaken on sama kui Joonis 3.3.8 (lk 28).

3.3.9 Kasutaja andmete vaade



Isikukood	37707071234
E-postiaadress	t990999@ttu.ee
Matriklinumber	990999
Seisund	immatricleeritud
Õpperühm	IAPB62

Joonis 3.3.15. Kasutaja andmete vaade.

Kasutaja andmete vaates kuvatakse sisselogitud kasutaja andmeid. Antud vaade ei ole käesoleva töö juures prioriteetne ning autor näeks tulevase õppeinfosüsteemi tudengi andmete vaates kindlasti rohkem informatsiooni.

4. Loodud veebiliidese võrdlus praeguse TTÜ Õppeinfosüsteemi veebiliidese

Käesoleva töö raames valminud veebiliidest võrreldakse Tallinna Tehnikaülikoolis kasutusel oleva õppeinfosüsteemi veebiliidese, Töö autoril on isiklik kokkupuude ainult kõnealuse süsteemiga, samuti on töö aluseks olnud infosüsteemi ülesehitus sarnane TTÜs kasutusel oleva õppeinfosüsteemiga. Kuna töö keskendub veebiliidese prototüüpimisele ning autoril puudub ligipääs praeguse õppeinfosüsteemi rakenduse ülesehitusele, võrreldakse antud peatükis vaid veebiliideseid.

Olemasoleva liidese ja käesoleva töö raames realiseeritud lahendusi võrreldakse jaotises 3.2 kirjeldatud materjalide alusel.

4.1 Üleüldine kasutajaliidese ülesehitus

Praeguse õppeinfosüsteemi kasutajaliides on ülesehitatud HTMLis sisalduvatele *frame* ja *frameset*-elementidele ehk raamidele (vt Joonis 4.1.1). Raamide kasutamine aitab jaotada lehe erinevateks osadeks, et võimaldada sektsioonide korduvkasutamine ja vähendada koodikordust.

The screenshot displays the TTÜ Learning Information System (Õppeinfosüsteem) interface. At the top, the user is logged in as 'Kaspar Kasik' (tudeng (tavaine) - 123997A9B). The navigation menu includes: Õpingukavad, Sooritusd, Avaldused, Arved ja lepingud, Õppeainete kodulehed, Minu andmed, Teated, Küsitlused, Muud, and Esileht. The main content area features three announcements: 'Õpingukava saad esitada kuni 12.02.2015 17:00 minu õpingukavad', 'Õppetoetuste taotluste esitamine on alanud. Taotlusti saab esitada kuni 28.02.2015 23:59 esita taotlus', and 'Punase joone päev on 16.02.2015'. Below these is a table of news (Teated (uued)) with columns for 'kuupäev', 'saaja', and 'pealkiri'. The table contains one entry: '05.02.2015 Merike Siidoja' with the headline 'Kandideerimine 2015/2016. õa välisõppe kohtadele on alanud!'. A 'kõik teated' link is also present. The right sidebar contains links for 'Õpingukava (alustamata) õpingukava on alustamata' and 'Koosta tunniplaan'. The footer includes the address 'Tallinna Tehnikaülikool, Ehitajate tee 5, 19086 Tallinn', navigation links, and the version 'VER: 26/01/15'.

Joonis 4.1.1. Praeguse õppeinfosüsteemi esileht sisselogitud tudengile.

Raamide kasutamist soovitatakse üldiselt vältida, kuna nende kasutamine toob kaasa mitmeid probleeme (Nielsen 1996, Horton 2006):

- Väikese ekraaniga seadmetel võib raamide kuvamine keeruliseks osutuda.
- Veebilehitseja tagasiliikumise nupp ei pruugi käituda vastavalt kasutaja ootustele.
- Raamide kasutamine ei võimalda külastajal veebilehte brauseri järjehoidjatesse lisada
- Veebilehitseja aadressiriba väärtus ei kohandu avatud vaatele – seega kasutaja ei saa jagada linki samale vaatele.
- Raame kasutavat veebilehte printides ei pruugi kasutaja saada soovitud tulemust.

Ka *wu* keele muster Avalikel lehtedel mitte raame kasutada (No frames on public sites, 27) (Graham, 2002) soovitab raamide kasutamist vältida, muuhulgas ka ülaltoodud põhjustel.

Praeguses õppeinfosüsteemis on raamidega seotud vigade esinemine küllaltki tihe. Ka käesoleva töö kirjutamise ajal tekkis küllaltki kummaline viga (vt Joonis 4.1.2):

ÕPPEINFOSÜSTEEM

Kaspar Kask tudeng (lavaine) - 1239971APB

Tagasiside | Abi

Logi välja EST | ENG vana | uus

Õpingukavad Sooritusused Avaldused Arved ja lepingud Õppeainete kodulehed Minu andmed Teated Küsitlused Muud Esileht

Invalid Portal Session **Invalid Portal Session**

An error was encountered while processing your Portal request, because your portal session is no longer valid. You have been logged out and you will automatically be redirected to the OracleAS Portal home page in 30 seconds. Click [OracleAS Portal home page](#) to go directly to the OracleAS Portal home page, or if your browser does not automatically redirect you. If you continue to have problems while accessing OracleAS Portal, close all your browser instances and try again.

Tallinna Tehnikaülikool, Ehitajate tee 5, 19086 Tallinn. TTÜ Siseveeb | TTÜ kodulehekülj | TTÜ Raamatukogu

VER. 11/05/15

Joonis 4.1.2. Veateade mitmes raamis õppeinfosüsteemis.

Ülaltoodud veateade tekkis, kuna sisselogitud kasutaja sessioon aegus. Kuna kasutaja üritas liikuda ülemise navigatsiooniriba kaudu teise vaatesse, millel on teistsugune külgriba, üritas süsteem ka külgriba kuvava raami sisu muuta ning ka seal kuvatakse veateadet.

Raamide kasutamisega seondub ka asjaolu, et veebilehe *title*-atribuut ehk pealkiri, mida kuvatakse veebilehitseja tiitliribal, on staatiline ehk ei muutu vastavalt sellele, millisel alamlehel kasutaja parasjagu viibib. Käesoleva töö raames valminud rakenduses on igale lehele antud ka unikaalne pealkiri kujul „<alamleht> | Õppeinfosüsteem“ tagamaks, et kasutaja saab ka minimeeritud akna või järjehoidjasse lisatud lehe puhul kiiresti teada, millise sisuga lehega on tegu (*wu* muster Unikaalsed nimed lehtedele, pealkirjadele ja meta-siltidele (Unique names for pages, titles and meta-tags, 62), Graham 2002).

Raamide kasutamise vastu räägib ka asjaolu, et HTML keele standardeid haldav World Wide Web Consortium (W3C) on 2014. aasta oktoobris valminud HTML keele 5. versiooni spetsifikatsioonis märkinud *frame* elemendi iganenuks (W3C 2014).

Kasutajaliides on jaotatud kolme raami: üks raam paikneb lehe ülaserivas ning on terve veebilehitseja akna laiune. Ülejäänud leht on jaotatud kaheks – vasakpoolne, kitsam raam sisaldab endas alammenüüd ning parempoolne raam aktiivset sisu.

Kasutajaliidese ülaserivas asuvas raamis on navigatsiooniriba, mille vasakus servas on logo. Riba keskel on tudengi andmed ning rippmenüü, milles saab sisselogitud isik vahetada rollide vahel. Seejärel on lingid tagasiside vormile ja ÕISI abiinfole, võimalus välja logida, keelt vahetada ja kasutajaliidese värvide vahel vahetada.

Navigatsiooniriba alumises servas on viited peamistele kategooriatele, millel klõpsates laetakse sisu alammenüü ja aktiivse sisu raamidesse.

4.2 Õpingukavade vaade

Klõpsates navigatsiooniriba lingil Õpingukavad, laetakse sellekohane alammenüü. Aktiivse sisu raami laetakse aga esileht. See pikendab kasutaja töövoogu lisaklõpsu ja –laadimisega, kuna õpingukavade nimekirja kuvamiseks tuleb vasakpoolsest menüüst omakorda klõpsata lingil „Minu õpingukavad“ (vt Joonis 4.2.1). Kui õpingukavade nimekiri laetaks juba navigatsiooniribal klõpsamise peale, kiirendaks see nende kasutajate töövoogu, kes soovisidki ligipääsu sellele, samas ei muudaks töövoogu aeglasemaks neile, kes soovivad ligipääsu mõnele muule alammenüü elemendile.

Joonis 4.2.1. Õpingukavade üldvaade õppeinfosüsteemis.

Õpingukavade üldvaates kuvatakse vaikimisi vaid käesoleva semestri õpingukava. Kõikide õpingukavade kuvamiseks on tarvis teha eraldi otsing, mis autori hinnangul pakub liialt võimalusi täpsustamiseks ja seega rikub KISS põhimõtet (*wu* muster Hoia asjad lihtsana (38), Graham 2002) (vt Joonis 4.2.2).

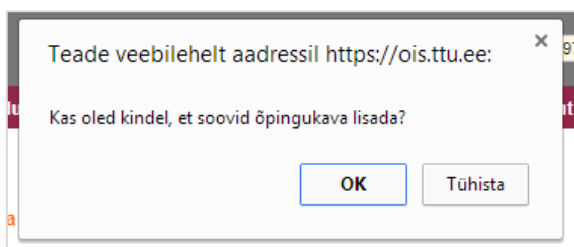
Joonis 4.2.2. Õpingukavade otsingu vorm.

Otsinguvorm pakub võimalust otsida väljade järgi, mida autori hinnangul valdav enamus tudengitest ei kasuta, nagu esitamise kuupäev ja kuupäevade vahemik ning õpingukavas olevate EAPde hulk ja vahemik ning ainete arv. Semestri valiku rippmenüüs on võimalik valida ka

tulevikus olevaid semestreid, mille kohta ei ole võimalik õpingukava omada. Lähtudes sellest, et nominaalajaga bakalaureuseõppe läbimise jooksul teeb tudeng omale kuus õpingukava, ei ole seda autori hinnangul liiga palju, et kõik tudengi õpingukavad korraga välja kuvada.

Aktiivse semestri alustamata õpingukava kohta on nimekirja juba tehtud sissekanne ning õpingukava täitmiseks tuleb klõpsata vastava sissekande ees olevale väikesele nupukesele, mille nimi ÕISI abimaterjalides on „Lisa uus / võlukepike“. Kuigi siin võib mõista, et tegu on n-ö. „uue asja välja võlumisega“, on see siiski pigem ebaloomulik metafoor ja vastuolus *wu* mustriga 56, mis soovitab saidil kasutada kujundeid, mis on vastava saidi kasutajale mõistetavad. Üleüldiseks lisamist tähistavaks märgiks on siiski kujunenud plussmärk, mis juba iseeneses kannab millegi lisandumise tähendust. Seda on kasutatud ka antud töö raames valminud realisatsioonis (vt. Joonis 3.3.6, lk. 27). Samuti on nupp ÕISile omaselt väga väike, mis teeb selle tabamise keeruliseks ning on kindlasti vastuolus Fitts'i seadusega (Tognazzini 2014), mille järgi on sihtpunkti tabamiseks kuluv aeg sõltuvalt sihtpunkti suurusest ja kaugusest selleni. Käesolevas prototüübis on üritatud nupud võimalikult mõistliku suurusega teha vastavalt nende ülesandele ja kontekstile. Osa tegevustele, nt. objektide kustutamise nuppudele on teadlikult tehtud väiksemad nupud, et kasutaja peaks tegevuse sooritamiseks teadlikult nupu pihta sihtima.

Kui tudeng soovib juba nimekirjas olevat õpingukava täitma hakata, kuvab süsteem dialoogiakna, kus küsitakse, kas tudeng on kindel, et soovib õpingukava lisada (vt Joonis 4.2.3).



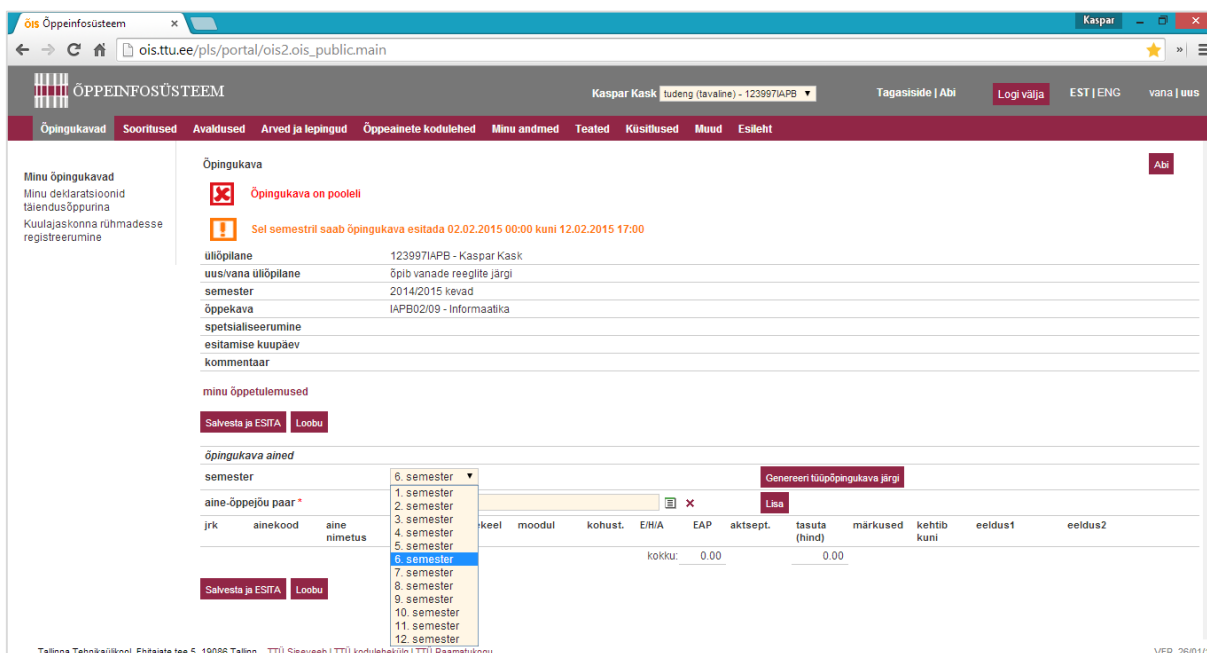
Joonis 4.2.3. Dialoogiaken, mis küsib, kas tudeng soovib uut õpingukava lisada.

Käesoleva töö autor leidis koos kaastudengitega, et antud dialoogiakna ainus eesmärk võiks olla see, et tudeng oleks kindel, mida nupuvajutus teeb. Selle töö raames realiseeritud rakenduses on selle tagamiseks kasutatud nuppu, millel on kirjas „Lisa uus õpingukava“. See vastab ka *wu* mustrile Ei ebameeldivatele üllatustele (No unpleasant surprises, 46) (Graham 2002). Samuti võib dialoogiakna sõnastus segadust tekitada, kuna nimekirjas juba oli üks

alustamata õpingukava ja tudeng soovib seda alustada, mitte uut lisada. Tuleks kasutada *wu* mustrit Sobiv sõnastus (Acceptable wording, 50) (Graham 2002).

4.3 Õpingukava detailvaade

Kui tudeng on kinnitanud, et soovib uue õpingukava lisada, avaneb õpingukava detailvaade (vt Joonis 4.3.1).



Joonis 4.3.1. Õpingukava detailvaade

Õpingukava detailvaates on võimalik õppekavasse lisada aine-õppejõu paare ning genereerida õpingukava tüüpõpingukava järgi, valides rippmenüüst soovitava semestri. Sellisel genereerimisel suunatakse tudeng järgmisesse vaatesse, kus on võimalik vaadata tüüpõpingukava aineid ning kui ainet õpetab semestril mitu õppejõudu, saab valida, kellele soovitakse ainet deklareerida. Selle töö raames valminud rakenduses on võimalik tüüpõpingukavadega eraldi vaates tutvuda ning soovi korral ühe klõpsuga vastav tüüpõpingukava enda omale aluseks võtta. Sarnaselt TTÜ ÕISiga saab valida õppejõudu neile ainetele, mida õpetab mitu õppejõudu. Küll aga lisatakse ained kohe tudengi õpingukavasse ning erinevalt praegusest liidesest puudub vajadus täiendavalt salvestada.

ÕPPEINFOSÜSTEEM Kaspas Kask | tudeng (tavaline) - 123997IAPB Tagasiside | Abi Logi välja EST | ENG vana | uus

Õpingukavad Sooritud Avaldused Arved ja lepingud Õppeainete kodulehed Minu andmed Teated Küsitused Muud Esileht

Minu õpingukavad
Minu deklaratsioonid
täiendusõppurina
Kuulajaskonna rühmadesse
registreerumine

Õpingukava Abi

Õpingukava on pooleli

Sel semestril saab õpingukava esitada 02.02.2015 00:00 kuni 12.02.2015 17:00

üliõpilane 123997IAPB - Kaspas Kask
uus/vana üliõpilane õpib vanade reeglite järgi
semester 2014/2015 kevad
õppekava IAPB02/09 - Informaatika
spetsialiseerumine
esitamise kuupäev
kommentaar

minu õppetulemused

Salvesta ja ESITA Loobu

Õpingukava ained

aine-õppejõu paar Lisa

jrk	aineкод	aine nimetus	õppejõud	õppekeel	moodul	kohust.	ENIA	EAP	aktsept.	tasuta (hind)	märkused	kehtib kuni	eeldus1	eeldus2
1	IDU0200	Veebipõhiste rakenduste arhitektuur, disain ja tehnoloogia	Raul Liivrand	eesti keel	Põhiõpe (Infosüsteemide arendamine)	jah	E	5.00		jah		20.06.2015		
2	ITV0130	Kasutajaliidesed	Jaagup Iive	eesti keel	Eriõpe / peariia (Eriõpe)	jah	H	5.00		jah		20.06.2015		
3	TET3070	Mikro- ja makroökonomika	Raisa Kokkota	eesti keel	Üldõpe (Sotsiaal- ja majandusteadused)	jah	E	4.00		jah		20.06.2015		
								kokku:	14.00	0.00				

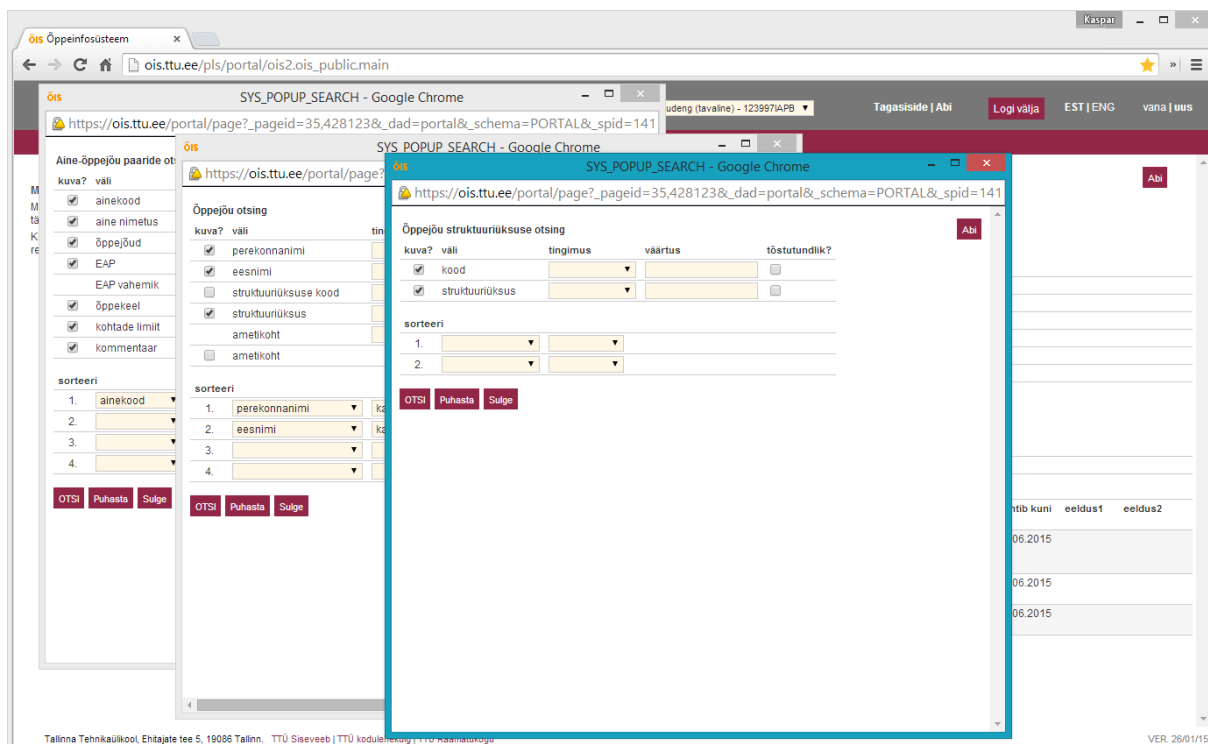
Salvesta ja ESITA Loobu

Tallinna Tehnikaülikool, Ehitajate tee 5, 19086 Tallinn. TTU Siseveeb | TTU kodulehekülj | TTU Raamatukogu VER: 26/01/15

Joonis 4.3.2. Õpingukava detailvaade peale tüüpõpingukava järgi genereerimist.

Lisaks tüüpõpingukava järgi genereerimisele (vt Joonis 4.3.2) on võimalik aineid ka käsitsi lisada. Selleks on lahter aine-õppejõu paar, mille kõrval oleval „Hüplikakna (pop-up) nupul“ klõpsates avaneb aken, milles on võimalik otsida aine-õppejõu paare. Hüplikakende kasutamine mõjub töövoogu häirivalt, kuna kasutaja peab nüüd oma tähelepanu pöörama uuele aknale. Samuti seostuvad hüplikaknad kasutajatele tihti soovimatute reklaamidega ja paljudel kasutajatel on installeeritud hüplikaknaid blokeeriv tarkvara, mis tuleb spetsiaalselt välja lülitada (Nielsen 2004).

Avanenud hüplikaknas on võimalik õppeaine otsimiseks sisestada aineкод, aine nimetus ja täpne EAPde arv või vahemik, samuti saab määrata sorteerimiskriteeriumeid. Õppejõu otsimiseks saab avada omaette hüplikakna, kus saab õppejõu otsimiseks sisestada ees- ja perekonnanime ja struktuuriüksuse koodi (vt Joonis 4.3.3). Rippmenüüst saab valida ametikoha ja sorteerimisalused. Õppejõu struktuuriüksuse otsimiseks saab avada omaette hüplikakna, milles on võimalik struktuuriüksuseid otsida koodi ja struktuuriüksuse nime järgi.

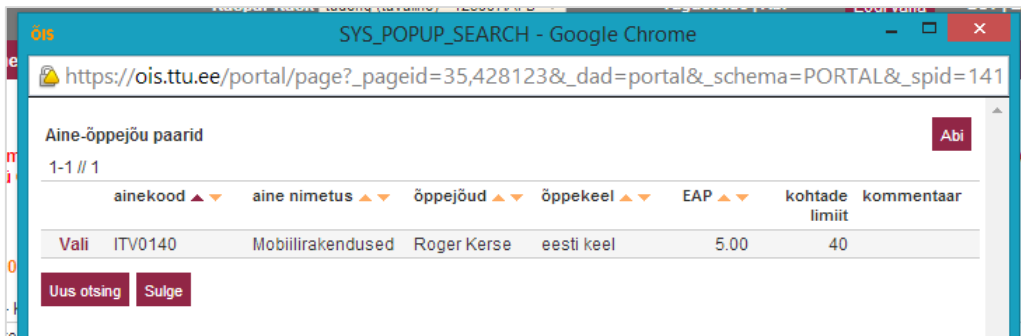


Joonis 4.3.3. Õppeinfosüsteemi õpingukavasse aine lisamine.

Enim on võimalik õpingukavasse aine lisamiseks avada kolm hüpikakent ning täita 68 vormielementi (lahtrit, rippmenüüd, märkeruutu).

Käesoleva töö raames valminud liideses on aine õpingukavasse lisamiseks võimalik (ja tarvis) täita kaks lahtrit. Kuna ainet on võimalik lisada kahel moel, on põhimõtteliselt selleks mõeldud neli lahtrit, ent korraga saab täita vaid kaks. Realiseeritud veebiliideses õpingukavasse ainete lisamist on kirjeldatud jaotises 3.3.8. Lahenduse kasuks otsustati peale praeguse liidesega sügavuti tutvumist ja konsulteerimist kaastudengitega. Selgus, et põhiliselt soovitakse õpingukavasse ainete lisamiseks neid otsida kas ainekoodi, aine nimetuse või õppejõu nime järgi ja sellised võimalused ka realiseeriti.

Olles aine-õppejõu paaride lehelt otsingu sooritanud, kuvatakse tulemusi tabeli kujul (vt Joonis 4.3.4).



Joonis 4.3.4. Aine-õppejõu paaride otsingu tulemus.

Tabelis on kuvatud ainekode, aine nimetus, õppejõud, õppekeel, EAPde arv, kohtade limiit ja kommentaar. Kahjuks ei ole kuvatud, mitu kohta veel ainele vaba on. Käesoleva töö raames realiseeritud liideses on see aine valiku juures välja toodud.

Õpingukava ained														
aine-õppejõu paar *														
jrk	ainekode	aine nimetus	õppejõud	õppekeel	moodul	kohust.	E/H/A	EAP	aktsept.	tasuta (hind)	märkused	kehtib kuni	eeldus1	eeldus2
1	IDU0200	Veebipõhiste rakenduste arhitektuur, disain ja tehnoloogia	Raul Liivrand	eesti keel	Põhiõpe (Infosüsteemide arendamine)	jah	E	5.00		jah		20.06.2015		
2	ITV0130	Kasutajaliidesed	Jaagup Irv	eesti keel	Eriõpe / peeriala (Eriõpe)	jah	H	5.00		jah		20.06.2015		
3	ITV0140	Mobiilirakendused	Roger Kerse	eesti keel	Vabaõpe	ei	A	5.00		jah		20.06.2015		
4	TET3070	Mikro- ja makroökonomika	Raisa Kokkota	eesti keel	Üldõpe (Sotsiaal- ja majandusteadused)	jah	E	4.00		jah		20.06.2015		
								kokku:	19.00	0.00				

Joonis 4.3.5. Õpingukavas olevate ainete tabel.

Õpingukavasse lisatud ainete tabelis kuvatakse aine kohta hulgaliselt infot: järjekorranumber õpingukavas, ainekode, aine nimetus, õppejõud, õppekeel, moodul, kas aine on kohustuslik, aine hindamise tüüp, EAPde arv, kas deklaratsioon on aktsepteeritud, kas aine on tasuta, märkused, deklaratsiooni kehtivus, eeldusaine 1 ja eeldusaine 2.

Autori realiseeritud liideses on kuvatud vaid ainekode, aine nimetust, EAPde arvu, õppejõudu ja moodulit. Õppekeelt, aine kohustuslikkust, aine hindamise tüüpi ja aine tasu käesoleva töö raames valminud rakenduses ei salvestata. Aine järjekorranumbrit õpingukavas ei ole autori hinnangul tarvis kuvada, kuna sel tabel puudub informatiivne väärtus. Kuna praeguse õppekorralduse alusel kehtivad deklaratsioonid niikuinii ainult ühe semestri, ei ole ka deklaratsiooni kehtivust tarvis kuvada. Eeldusaineid võiks kuvada aine kirjelduse juures peale aine nimekirjast valimist, kuna see info on oluline enne aine õpingukavasse lisamist. Veateadet deklaratsiooni mitteaktsepteerimise kohta võiks kuvada alles siis, kui õppejõud on

deklaratsiooni tagasi lükanud. Praeguse realisatsiooni üheks plussiks on kindlasti see, et nn. KISS põhimõtte (*wu* muster Hoia asjad lihtsana (38), Graham 2002) alusel kujundatud tabelit on lihtsam hoomata, samas see kuvab peamise vajaliku informatsiooni.

Nii praeguses ÕISis kui loodud liideses on viimati lisatud aine kuvatud rohelisel taustal, et see oleks paremini märgatav.

Aine õpingukavast eemaldamisel märgitakse praeguses ÕISis rida punase tausta ja läbikriipsutatud tekstiga, autori realisatsioonis aga eemaldatakse rida tabelist ja ka andmebaasist täielikult. Kuna ainet on lihtne uuesti õpingukavasse lisada, ei pea autor vajalikuks eemaldatud aine hoidmist õpingukava juures, samuti on autoril isiklikult tekkinud segadus, miks aine eemaldamisel tabelist ära ei kadunud.

Üldise probleemina kõikide aineid sisaldavate vaadete juures leiab autor selle, et EAPde arvu kuvatakse kahe komakohaga ka juhul, kui aine eest saadavate ainepunktide arv on täisarv. Samuti pole autor seni kohanud ühtegi õppeainet, mille eest saadavate ainepunktide arv ei oleks täisarv.

Peale õpingukava esitamist kuvatakse õpingukavade vaates jätkuvalt oranži kirja ning hüüumärgiga teadet sisuga „Sel semestril saab õpingukava esitada ...“. Teate värvist ja ohtu sümboliseerivast hüüumärgist võib jääda mulje, et midagi on jätkuvalt puudu ka peale õpingukava esitamist. Realiseeritud liideses kuvatakse avakuval teadet, et tudeng on oma õpingukava edukalt esitanud.

5. Kokkuvõte

Antud bakalaureusetöö põhieesmärgiks oli luua programmeerimiskeelt PHP ja andmebaasisüsteemi Oracle kasutatav näiterakendus andmebaaside õppeaineid õppivatele tudengitele. Näiterakendus loodi Erki Eessaare poolt Andmebaasid I ja II ainete tarvis koostatud Ülikooli infosüsteemi õpingukavade allsüsteemi näiteprojekti alusel.

Töö tulemusena valmis nõutud rakendus ja lisaks ka võrdlus loodud rakenduse kasutajaliidese ja praeguse Tallinna Tehnikaülikooli õppeinfosüsteemi õpingukavade koostamise kasutajaliidese vahel.

Võrdluses on välja toodud mitmeid murekohti praegu kasutusel oleva õppeinfosüsteemi kasutajaliidese. Tulevastes õppeinfosüsteemi arendustes tuleks samuti jälgida häid kasutajaliidese tavasid ja disainimustreid, nagu on tehtud käesoleva töö raames valminud kasutajaliidese puhul.

Töö raames valminud kasutajaliidest on kindlasti võimalik edasi arendada. Tulevasteks uurimisteedeks võiks olla näiteks infosüsteemi kasutajaliidese ligipääsetavuse tagamine, millele antud töös tähelepanu pole pööratud.

1. Kas eesmärk saavutati?

Töö käigus valmis seni puudunud näiterakendus, mida on võimalik andmebaaside õppeainete tudengitel kasutada näitena sellest, millist rakendust on võimalik PHP ja Oracle abil luua. Lisaks on rakendusel kaasaegne ja headele disainimustritele vastav kasutajaliides (vt „Sissejuhatus“ jaotis 1.1, lk 11).

2. Põhitulemuste loetelu

Töö põhitulemuseks on eesmärgis püstitatud näiterakenduse ja häid kasutajaliidese kavandamise tavasid ja disainimustreid järgiva kasutajaliidese valmimine. Lisaks on loodud kasutajaliidest võrreldud kasutusel oleva õppeinfosüsteemi samalaadse kasutajaliidese ning välja on toodud mitmeid praeguses õppeinfosüsteemis leiduvaid vastuolusid kasutajaliideste ekspertide koostatud materjalidega (vt „Sissejuhatus“ 1.2, lk 11).

Summary

The main goal of this bachelor's thesis was to create an example web application using PHP programming language and Oracle database management system that could be used by students studying database subjects. The application was based on an example project by Erki Eessaar. Its topic is Study Plan Subsystem of a University Information System. The example project was created for subjects Databases I and II.

The required application was created as the main result of this thesis. The application is an example of a type of application that can be created by using PHP and Oracle. A modern user interface was also designed by using best practices and design patterns created by specialists of this field. A comparison study was carried out between the user interface of the created application and the current user interface for creating study plans in Tallinn University of Technology's Study Information System.

Several problems about the current user interface design and usability were pointed out in this thesis. The good practices and design patterns of creating user interfaces that were used during the creation of this application's user interface should also be taken into account when developing a new version of the Study Information System.

The created user interface can definitely be improved. For instance, a future research topic could be improving accessibility in such information systems that was not considered in this thesis.

Kasutatud kirjandus

1. British Columbia Institute of Technology (2015) CodeIgniter Web Framework – [Online] <http://www.codeigniter.com/> (20.05.2015)
2. BuiltWith Pty Ltd (2015) Framework technologies Web Usage Statistics – [Online] <https://trends.builtwith.com/framework> (14.05.2015)
3. Design science (methodology) - Vikipeedia, vaba entsüklopeedia (2015) – [Online] https://en.wikipedia.org/wiki/Design_science_%28methodology%29 (21.05.2015)
4. Eessaar, E. (2014) TTÜ: Andmebaasi näidisprojekt „Ülikooli infosüsteemi õpingukavade allsüsteem“ – [Online] õppekeskkond Maurus <http://maurus.ttu.ee/> (19.05.2015)
5. Free Software Foundation, Inc. (2015) Various Licenses and Comments about Them – GNU Project - Free Software Foundation – [Online] <https://www.gnu.org/licenses/license-list.html#X11License> (23.05.2015)
6. Horton, S. (2006) Avoid using frames – [Online] http://universalusability.com/access_by_design/frames/avoid.html (16.05.2015)
7. Hudson, P. (2015) Interpreting vs. Compiling – Hacking with PHP - Practical PHP – [Online] <http://www.hackingwithphp.com/2/2/2/interpreting-vs-compiling> (16.05.2015)
8. Google (2015) AngularJS — Superheroic JavaScript MVW Framework – [Online] <https://angularjs.org/> (12.05.2015)
9. Graham, I. (2002) Web Usability – [Online] <http://www.trireme.com/WU/> (19.05.2015)
10. Kasutajajuhend (otsinguvormid) (2015) – [Online] TTÜ õppeinfosüsteem (18.05.2015)
11. Less (n.d.) Getting started | Less.js – [Online] <http://lesscss.org/> (14.04.2015)
12. Nielsen, J. (1996) Original Top 10 Mistakes in Web Design – [Online] <http://www.nngroup.com/articles/original-top-ten-mistakes-in-web-design/> (15.05.2015)
13. Nielsen, J. (2004) The Most Hated Advertising Techniques – [Online] <http://www.nngroup.com/articles/most-hated-advertising-techniques/> (15.05.2015)

14. Nielsen, J. (2008) 10 Best Application UIs – [Online] <http://www.nngroup.com/articles/10-best-application-uis/> (02.05.2015)
15. PHP Group, The (2015) PHP: Installation – Manual – [Online] <http://php.net/manual/en/> (01.05.2015)
16. Stevens, S. (2012) An Introduction to Views & Templating in CodeIgniter – [Online] <http://code.tutsplus.com/tutorials/an-introduction-to-views-templating-in-codeigniter--net-25648> (20.03.2015)
17. Tallinna Tehnikaülikooli õppekorralduse eeskiri (2014) – [Online] <http://www.ttu.ee/public/t/tudengile/oppeinfo/juhendid/OKE.pdf> (22.05.2015)
18. Tognazzini, B. (2014) First Principles of Interaction Design (Revised & Expanded) – [Online] <http://asktog.com/atc/principles-of-interaction-design/> (06.05.2015)
19. Trinidad Consulting OÜ (2009) Kasutajakeskse veebi lehekülgede disain – [Online] https://www.mkm.ee/sites/default/files/kasutajakeskse_veebi_lehekylgede_disain.pdf (09.05.2015)
20. Vallaste, H. (2015) e-Teatmik: IT ja sidetehnika seletav sõnaraamat – [Online] <http://vallaste.ee/> (20.05.2015)
21. PHP - Vikipeedia, vaba entsüklopeedia (2015) – [Online] <http://et.wikipedia.org/wiki/PHP> (13.05.2015)
22. Wroblewski, L. (2005) Web Application Form Design – [Online] <http://www.lukew.com/ff/entry.asp?1502> (15.05.2015)
23. World Wide Web Consortium (W3C) (2014) 11 Obsolete features — HTML5 – [Online] <http://www.w3.org/TR/2014/REC-html5-20141028/obsolete.html#frames> (17.05.2015)
24. YOOftheme GmbH (2015) UIkit – [Online] <http://getuikit.com/index.html> (20.05.2015)