

KLAIPĖDOS UNIVERSITETAS
GAMTOS IR MATEMATIKOS MOKSLŲ FAKULTETAS
INFORMATIKOS KATEDRA

ŽILVINAS VAIRA

Besimokančiojo modelio vaizdavimas virtualioje mokymo(si) aplinkoje

Baigiamasis magistro darbas

Darbo vadovas: prof. habil. dr. Antanas Andrius Bielskis

Darbo konsultantė: lekt. dr. Dalia Baziukaitė

Klaipėda, 2007

Anotacija

Adaptyviose sistemose prisitaikymas prie besimokančiojo yra paremtas duomenų apie besimokantįjį rinkimu, analizavimu ir mokymosi aplinkos pritaikymu besimokančiojo poreikiams. Tam būtina parinkti tokius „sekimo“ duomenis, pagal kuriuos būtų galima daryti išvadas sistemos lygmenyje. Šie duomenys turi būti suprantami sistemai, nuolat atnaujinami, lengvai surenkami. Jie gali būti įvairūs, pradedant nuorodų paspaudimo laiku, baigiant akies žvilgsnio nukreipimo vieta. Pagal tai sistema gali transformuoti duomenis ir – priklausomai nuo savo paskirties – duoti patarimus ar tam tikromis priemonėmis pavaizduoti naudingą informaciją, tuo pačiu atsižvelgdama į rezultatus. Tačiau daugelis šių sistemų yra tik prototipo stadijoje arba naudojamos nesudėtingiems procesams vykdyti. Mums žinomos adaptyvios sistemos yra skirtos vieno dalyko probleminei sričiai. Šiame darbe tirsime besimokančiojo modelio perteikimo galimybę virtualioje mokymo(si) aplinkoje (VMA), kai besimokantysis dalyvauja viename arba keliuose nuotoliniuose kursuose.

RAKTINIAI ŽODŽIAI: besimokančiojo modelis, VMA, adaptyvumas, adaptyvi sistema, duomenų rinkimas, nuotoliniai kursai.

Abstract

In adaptive systems adaptation to learner is defined by collection of data about learner, collected data analysis and learning environment adaptation to learners needs. Looking forward to this goal it is necessary to select such logging data, which can be used to make conclusions in a system level. Such data should be acceptable to system, interactive and easy to collect. Data used can be various, starting from click time of the link ending with eye look direction angle. Such system can provide data transformations and depending on purpose, give suggestions. Using necessary tools depict useful information, reacting to results. Most of currently build systems are only in prototype stage or are used for unsophisticated process. Most of them are one subject systems. In this work we address a possibility of student model's depiction in a VLE, when student participate in one or more than one distance course.

KEYWORDS: student model, VLE, adaptation, adaptability, adaptive system, data collection, distance course, distance learning, personalized environment.

Turinys

Terminų ir santrumpų žodynas	4
Įvadas	5
1. Besimokančiojo modelis adaptyviose mokymo(si) aplinkose	7
1.1. Besimokančiojo samprata ir mokymosi efektyvumas	9
1.2. Siūlomos priemonės besimokančiojo modelio perteikimui	11
1.3. Problemų, kurias siekiama spręsti perteikiant besimokančiojo modelį, formulavimas	17
1.4. Besimokančiojo modelio panaudojimo galimybės ir ribos	18
1.5. Galimi sprendimo būdai	19
1.6. Skyriaus apibendrinimas	19
2. Besimokančiojo modelio naudojimas nuotolinių studijų efektyvumui padidinti	21
2.1. Naudojamų technologijų galimybių įvertinimas	21
2.2. Adaptyvumo samprata besimokančiojo modelio vaizdavime	21
2.3. Lanksti programuojamos terpės architektūra	22
2.4. Agentinės technologijos nuotoliniame el. mokyme	22
2.5. Sprendimų rengimo sistemos skaičiavimų panaudojimo eksperimentas	22
2.6. Papildomi matematiniai skaičiavimai	29
2.7. Skyriaus apibendrinimas	30
3. Nuotolinių studijų posistemės vizija, koncepcinis modelis	31
3.1. Nuotolinių studijų sistemos vartotojų identifikavimas	31
3.2. Tikslų modelis	35
3.3. Procesų modelis	37
3.4. Organizacinis modelis	40
3.5. Resursų aspektas	42
3.6. Galimas nuotolinių studijų modifikacijos veikimo procesas	42
3.7. Skyriaus apibendrinimas	43
4. Besimokančiojo modelio adaptyvioje VMA posistemėje įdiegimas	45
4.1. Duomenų bazės modifikavimas	45
4.2. Dėstytojo sąsajos loginės dalies modifikavimas	47
4.3. Vartotojo sąsajos loginės dalies modifikavimas	50
4.4. Skyriaus apibendrinimas	52

Rezultāti.....	55
Išvados.....	56
Perspektyvas.....	57
Literatūra.....	58
Ilustrācijas.....	62
Priedai.....	64

Terminų ir santrumpų žodynas

- AHP** Analitinis Hierarchinis Procesas
- AIMS** Dalykinė programa skirta mokomosios medžiagos peržiūrėjimui, mokomoji medžiaga yra modifikuojama remiantis vartotojo modeliu
- BM** besimokančiojo modelis
- DB** duomenų bazė
- DBS** duomenų bazių sistema
- DBVS** duomenų bazių valdymo sistema
- IT** technologijos, kuriomis kuriama, saugoma ar skleidžiama bet kurios rūšies informacija.
- KU** Klaipėdos universitetas
- MySQL** nemokama duomenų bazių valdymo sistema
- MOODLE** Modular Object-Oriented Dynamic Learning Environment
- PHP** (angl. *Hypertext Preprocessor*) tarnybinės stoties scenarijų vykdymo kalba ir interpretatorius
- SRS** Sprendimo rengimo sistemos
- SELFEDU** Naujasis Moodle savarankiško mokymo(si) formatas (angl. *Self Educated*)
- STyLE-OLM** Dalykinė programa, sukurta specialiai studento modelio formavimui ir yra visiškai nepriklausoma nuo konkrečios sistemos, ji skirta tik vartotojo modelio suformavimui [18]
- VMA** virtuali mokymo(si) aplinka

Įvadas

Jau seniai pasaulyje diskutuojama apie adaptyvių sistemų poreikį ir taikymą. Jų veikimo principas – tai informacijos apie tam tikrus realaus pasaulio objektus rinkimas ir apdorojimas. Tokia sistema, surinktą informaciją apdorojusi įvairiais algoritmais ir skaičiavimais, turėtų padaryti išvadas ir, remdamasi tom išvadom, turėtų netiesiogiai paveikti realaus pasaulio objektą. Tokios protingos/mąstančios sistemos yra plačiai taikomos technikos srityje, tačiau čia jų darbas vyksta su aiškiai apibrėžtu duomenų kiekiu ir dažniausiai su daiktais, o ne su žmonėmis. Tačiau šias sistemas sieja bendras tikslas – jos kuriamos specialistams paduoti. Nuotolinio mokymo(si) sistemose toks specialistas yra dėstytojas. Siūlomų nuotolinių kursų mokymo medžiagą dažnai sudaro didelės apimties dokumentai arba nuorodos į juos, tų dokumentų kiekis neretai per didelis, be to, nėra patogių įrankių, kaip tokioje gausybėje duomenų atsirinkti tai, kas reikalinga besimokančiajam. Besimokančiajam taip pat labai svarbu, kad sistemos ribose mokymosi aplinką būtų galima kiek įmanoma pritaikyti savo reikmėms. Šiuos trūkumus turėtų pašalinti adaptyvių komponentų įdiegimas Virtualioje Mokymo(si) Aplinkoje (VMA). Dažniausiai virtualios mokymo(si) sistemos jau turi įrankius ir duomenų bazines, skirtas informacijai apie besimokančius rinkti. Nors specialaus formato kursams, skirtiems savarankiškomis studijoms, to nepakanka, tačiau ši informacija jau sudaro nemažą reikalingų duomenų kiekį. Reikia sužinoti ir paties studento požiūrį bei vertinimą, kad būtų galima tiksliau nustatyti besimokančiojo tikslus ir gabumus. Tam buvo sumanyta panaudoti duomenų modelius, kurie perteiktų informaciją apie studentą sistemoje. Kad duomenys būtų tikslūs, turi būti nuolat stebimas tiek besimokantysis, tiek ir sistemos bendravimas su juo. Duomenų modelis turi atitikti ir eilę kitų reikalavimų. Labai svarbu yra nuspręsti, kokius duomenis rinkti. Virtualioje mokymo(si) aplinkoje objektas, apie kurį kaupiami duomenys, yra studentas ar moksleivis (apibendrintai – besimokantysis), todėl ir duomenų modelis vadinamas *besimokančiojo modeliu*. Gana nedaug yra nuveikta mokymo(si) sistemų prisitaikymo arba adaptavimo srityje. Todėl šiame darbe pamėginsime išnagrinėti galimybes ir būdus, kaip besimokančiojo modelį galima perteikti adaptyvioje virtualaus mokymo(si) sistemoje.

Darbo tikslas:

Perteikti besimokančiojo modelio informaciją VMA joje prieinamomis priemonėmis.

Uždaviniai:

1. Aprašyti žinomų adaptyvių sistemų besimokančiojo modelį sudarančią informaciją;
2. Pasirinkti tinkamiausią besimokančiojo modelio struktūrą, kuri bus pritaikyta pasirinktoje VMA;
3. Parinkti ir aprašyti priemones, kuriomis įmanoma perteikti besimokančiojo modelį;
4. Sukurti programinį modelį, skirtą besimokančiojo modelio perteikimo priemonėms įdiegti VMA;

Norėtūsi atkreipti dėmesį, kad tema nėra pasirinkta atsitiktinai, naujosios posistemės parengiamieji darbai buvo atlikti ankstesnėse studijose. Jau yra suformuota tam tikra terpė nuotolinių studijų sistemoje „Moodle“, skirta adaptyviam komponentui realizuoti. Be šios atviro kodo sistemos darbe bus naudojama papildoma programinė įranga: „Microsoft Visio“, „Macromedia Dreamweaver“, MySQL duomenų bazių valdymo sistema, „Expert Choice“ mokomoji versija, „Microsoft Office“ programų paketas.

Magistro baigiamojo darbo rezultatai aprobuoti mokslinėje konferencijoje ir publikacijoje:

D. Baziukaitė, D. Idzelytė, Ž. Vaira, V. Vičkus (2006) Virtualios mokymo(si) aplinkos adaptyvi intelektualios posistemės: koncepcinis modeliavimas ir diegimas. Technologijos mokslo darbai vakarų Lietuvoje, KU, p.141-151.

1. Besimokančiojo modelis adaptyviose mokymo(si) aplinkose

Šioje dalyje panagrinėsime įvairias adaptyvias sistemas ir jų veikimo principus. Trumpas apibrėžimas, kas tai yra adaptyvi sistema, jau buvo suformuluotas įvadinėje dalyje: tai sistema, kuri automatiškai renka ir apdoroja informaciją apie tam tikrus realaus pasaulio objektus bei įvairių algoritmų ir skaičiavimų pagalba sugeba daryti išvadas ir, jų besiremdama, gali netiesiogiai, bet realiai paveikti realaus pasaulio objektą. Tai gana abstraktus adaptyvios sistemos apibrėžimas, be to, adaptyvios sistemos yra taikomos labai skirtingose srityse, tačiau jų veikimo esmei nusakyti to pakanka. Siekiant suformuluoti platesnį požiūrį į adaptyvias sistemas, neapsiribota vien tik nuotoliniam ar įprastam mokymui(si) skirtomis adaptyviomis sistemomis. Darbe pabandysime panagrinėti ir su mokymu(si) nedaug bendro turinčias technikas bei metodus adaptyvumui sukurti [3].

Pirmiausia reikėtų aptarti, kas tai yra besimokančiojo modelis, kadangi šis terminas bus dažnai naudojamas. Besimokančiojo modelis daugelyje literatūros šaltinių aprašomas panašiai. Trumpai tariant – tai yra duomenys apie besimokantįjį. Tačiau tai labai netikslus apibrėžimas. Visų pirma, tie duomenys nėra tik informacija apie patį besimokantįjį, bet ir daug kitų atributų, kurie besimokantįjį sieja su virtualia mokymosi aplinka. Taip pat besimokančiojo modelį gali sudaryti ir taisyklės, kurių pagalba duomenys yra paverčiami į žinias, suprantamas sistemai, o tos žinios perteikiamos pasitelkiant vartotojui suprantamą formą, nesudėtingą grafinę ir tekstinę vartotojo sąsają [3]. Dažniausiai literatūroje tokio modelio sudarymas svarstomas keliais aspektais. Pirmiausia – kokia informacija turi sudaryti besimokančiojo modelį ir kas yra nereikalinga. Be standartinių vartotojo duomenų (tokių kaip vardas, pavardė, gimimo metai ir kt.) besimokančiojo modelio sudaryme svarbūs sistemos ir vartotojo sąsajai užtikrinti reikalingi duomenys (tokie kaip vartotojo aplinkos nustatymai, prisijungimo laikas ir pan.). Kita problema – kokius metodus naudoti, kad informacija taptų suprantama sistemai, kad pati sistema galėtų įvertinti ir padaryti išvadas apie studento modelį. Galiausiai iškyla klausimas, kaip pateikti apdorotą informaciją, kad būtų galima padėti vartotojui, šiuo atveju besimokančiajam, jo mokymosi procese. Be to, įvairiuose literatūros šaltiniuose keliamos ir papildomos problemos, su kuriomis susiduriama jau bandant taikyti adaptyvias sistemas, pvz., „šalto starto“ problema, kai mokymosi proceso pradžioje besimokančiojo modelyje nėra duomenų – norint juos išgauti naudojami testai, tačiau šie dažnai būna gremėzdiški ir jų gaunama informacija gali būti netiksli [19].

Pagrindinė priežastis, dėl kurios buvo pasirinktas adaptyvumo taikymas nuotolinių studijų sistemose, yra ta, kad adaptyvių sistemų kūrimas kol kas apsiriboja nelanksčiu jų pritaikymu, dažnai

jos tėra tik eksperimentinio pobūdžio ir nuotolinėms studijoms, kur jų reikėtų labiausiai, tegali pasiūlyti tik labai primityvius sprendimus [2].

Toliau panagrinėsime plačiau galimas problemas, kurios dažnai aptiriamos adaptyvių sistemų ir besimokančiojo modelio formavimo procese. Pirmiausia pabandysime identifikuoti patį besimokantįjį ir įprasto bei nuotolinio mokymosi būdo efektyvumą.

1.1. Besimokančiojo samprata ir mokymosi efektyvumas

Besimokančiuoju galima vadinti aukštosios mokyklos studentą, moksleivį, stažuotoją ar bet koki kitą žmogų, siekiantį naujų žinių ir tam naudojančių specialias priemones. Besimokantysis turi tikslą, kuris nusako, kodėl jam reikalingos žinios, turi išankstines žinias ir naudojami tam tikromis priemonėmis, reikalingomis įgyti naujų žinių. Mokymosi tikslai gali būti įvairūs: kelti kvalifikaciją, išmokyti kalbą, įgyti profesiją ar tiesiog patenkinti smalsumą. Nuo tikslų dažnai priklauso tai, kiek besimokantysis deda pastangų kažką išmokyti. Išankstines žinias galima suprasti kaip jau turimą kvalifikaciją, diplomą ar šiaip kasdieniniame gyvenime sukauptą patirtį. Šios žinios labai svarbios mokymosi efektyvumui. Pavyzdžiui, jei studentas iš asmeninės patirties geba sklandžiai kalbėti anglų kalba, turbūt neefektyvu bus mokytis pradedančiųjų anglų kalbos kursų. Priemonės mokymui(si) yra labai įvairios. Pirmiausia atskirime du mokymosi būdus, kadangi šiame darbe tai labai svarbu. Tai įprastas ir nuotolinis mokymas(is). Pastarasis atsirado palyginti neseniai, kartu su naujomis technologijomis (vaizdo ir garso nuotolinis perdavimas, pasaulinis kompiuterių tinklas ir t.t.). Šie du mokymosi būdai vienas nuo kito labai smarkiai skiriasi. Nuotoliniame mokyme(si) prarandamas dėstytojo glaudus kontaktas su besimokančiuoju, kas labai svarbu siekiant pritaikyti mokymo(si) kursą kiekvienam studentui individualiai. Kita vertus, nuotolinio mokymo(si) sistemos turi papildomų įrankių, kurių tradiciniame mokyme nėra. Tokie įrankiai duoda dėstytojui papildomų žinių apie studentus, kurių tradiciniame mokyme neįmanoma buvo gauti. Tačiau tai nekompensuoja besimokančiojo ir dėstytojo tiesioginio bendravimo nebuvimo. Galimas ir tarpinis mokymo(si) būdas tarp nuotolinio ir tradicinio, kai paskaitose papildomai naudojamos nuotolinės sistemos praturtina mokymą(si) moderniais įrankiais. Taip labai paprastai galima būtų išsivaizduoti skirtumus tarp tradicinio ir nuotolinio mokymo(si), bei abiejų bendro panaudojimo efektyvumą [45].

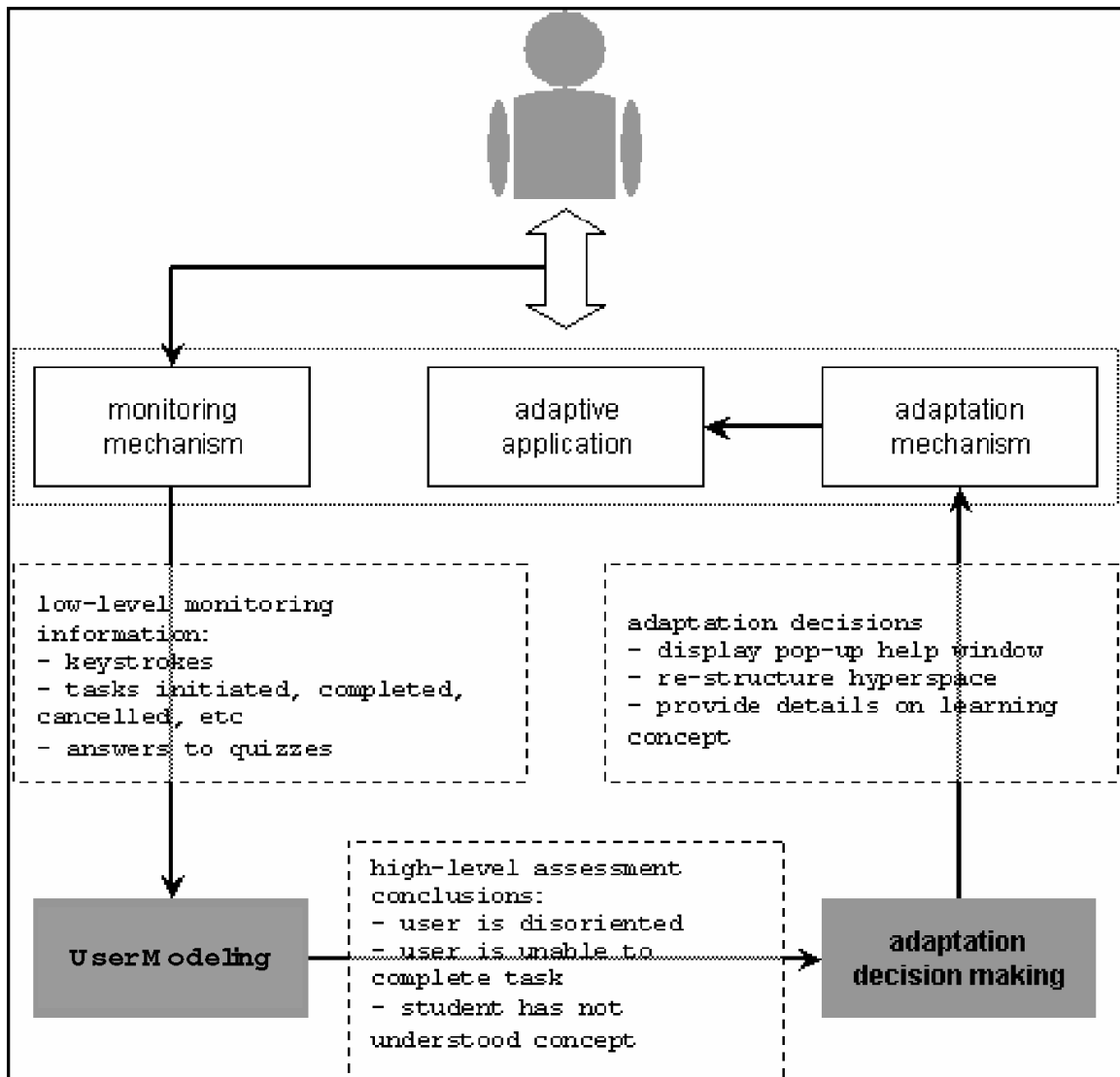
Vienas iš būdų priartinti nuotolines studijas prie tradicinio mokymo(si) - tobulinti mokymo(si) įrankius, naudojamus nuotolinėse sistemose. Taip pat reikia nepamiršti, kad naudotis naujais įrankiais taip pat reikia sugebėti ir tai neturėtų papildomai apkrauti dėstytojo ar besimokančiojo. Kalbant apie įrankius, daugelis nuotolinių sistemų savo rinkiniuose siūlo įvairius variantus: siekiant sumažinti bendravimo stoką, diegiami pokalbių kambariai, žinučių siuntimo sistemos; norint padidinti besimokančiojo egzaminavimo efektyvumą ir saugumą, pateikiami įvairius testavimo moduliai. Bet jei susimąstysime, kodėl besimokančiajam reikalingas glaudus bendradarbiavimas su dėstytoju ir kodėl jis toks svarbus, pamatysime, kad pagrindinė priežastis yra individualaus dėmesio kiekvienam studentui trūkumas. Studijuojant įprastai, dėstytojas gali atsižvelgti į kiekvieno studento sugebėjimus ir poreikius, o mokantis nuotoliniu būdu to neįmanoma

padaryti. Todėl toliau šiame darbe mes remsimės prielaida, kad **studento mokymo(si) aplinkos gebėjimas prisitaikyti prie studento poreikių gali sumažinti prarają tarp dėstytojo ir besimokančiojo**. Be to, žvelgsime į studento mokymąsi iš studento pusės, **todėl kad dauguma įrankių yra skirti dėstytojams stebėti studentą bei jam padėti, o pačiam studentui analizuoti savo mokymo(si) procesą bei rezultatus nėra galimybių [44,16]**.

Taigi kiek kitokiu kampu pažvelgę į besimokantįjį bei į nuotolinio mokymo(si) efektyvumą mažinančias priežastis, galime apibrėžti, koks turėtų būti nuotolinių studijų besimokantysis ir kaip galima būtų jo mokymosi procesą padaryti efektyvesnį ir kokybiškesnį. **Besimokantysis** – tai studentas, kuris savarankiškai nori siekti naujų žinių, turi jau sukauptą mokymosi patirtį bei turi kompiuterinio raštingumo minimalių įgūdžių. **Mokymosi efektyvumas** grindžiamas mokymo(si) aplinkos (mokymosi šaltinio, veiklų, vertinimo ir kt.) prisitaikymu studento individualiems poreikiams. Rezultatui pasiekti bus nagrinėjami adaptyvių sistemų, skirtų nuotoliniam mokymui(si) modeliai. Siekiant dar labiau susiaurinti nagrinėjamą sritį, apibrėžiame **besimokančiojo modelį** kaip žinių ir duomenų rinkinį apie besimokantįjį, į kurį įeina kurso planas, lygio pasirinkimas, „sekimo“ duomenys bei juos perteikiančios priemonės suprantamos besimokančiajam [2]. Šis apibrėžimas susiaurina tiriamo besimokančiojo modelio struktūrą ir sudėtį. Literatūroje [39] pateikiami apibrėžimai ir standartai besimokančiojo modelį vaizduoja apibendrintai, mums reikės tikslesnio varianto [17].

1.2. Siūlomos priemonės besimokančiojo modelio perteikimui

Adaptyviose sistemose naudojamas besimokančiojo modelis ir jo vieta visoje adaptyvioje sistemoje skirtingų autorių publikacijose yra aprašomi skirtingai, tačiau bendri bruožai būdingi daugeliui tokių modelių. Aptarsime bendresnį besimokančiojo modelio, kurį galima būtų taikyti įvairiose sistemose, variantą. Autoriaus [9] straipsniuose iškeltos problemos ir sprendimai yra labai konkretūs ir universalūs, todėl šiais straipsniais remiasi daugelis ta kryptimi dirbančių mokslininkų. Šiame straipsnyje kalbama apie konkrečią adaptyvaus mokymosi organizavimo schemą:



1 pav. Peter Brusilovsky siūloma adaptyvaus mokymosi organizavimo schema [9]

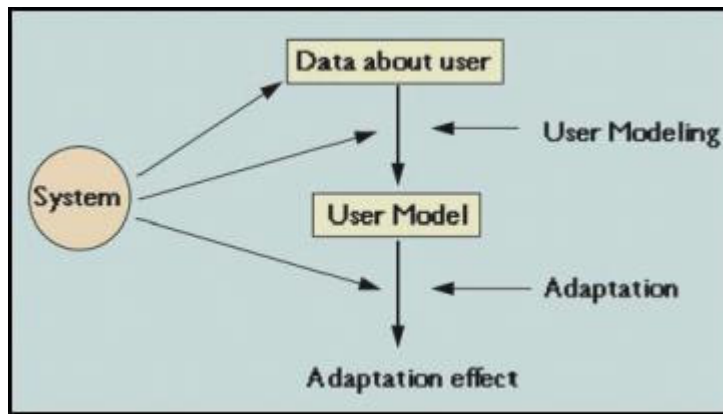
Paveikslėlyje (1 pav.) vaizduojama bendra adaptyvaus mokymosi organizavimo schema, kurią sudaro stebėjimo mechanizmas (angl. *Monitoring Mechanism*), vartotojo modelio formavimas (angl. *User Modeling*), adaptacijos sprendimo priėmimas (angl. *Adaptation Decision Making*),

adaptacijos mechanizmas (angl. *Adaptation Mechanism*), adaptyvi dalykinė programa (angl. *Adaptive Application*). Iš šios schemos galima nesunkiai suprasti patį adaptyvumo procesą ir besimokančiojo modelio indėlį adaptyvumui. Reikia pažymėti, kad ši schema panaudota adaptyviam Interneto medžiagos naršymui. Ji nėra skirta nuotolinėms studijoms. Tačiau ir modelis, ir jo vieta sistemoje yra pakankamai logiškai ir aiškiai pavaizduoti. Panagrinėsime schemą išsamiau.

Šis procesas yra suskirstomas į du sluoksnius [9]:

1. **Vartotojo modelio įvertinimas.** Šiame sluoksnyje keliami klausimai: ar teisingai suformuotas besimokančiojo modelis pagal gautus duomenis ir kaip tai patikrinti? Ar teisingai nustatytas vartotojo interesas? Galimas sprendimas būtų testai, taip pat besimokantysis galėtų pats peržiūrėti savo modelį ir jį koreguoti. Siūlomas ir žinių klasifikavimas (pradedantysis, pažengęs, ekspertas). Įvertinus ir nustatčius, kad adaptyvumas duos norimą rezultatą, modelio informacija gali būti užfiksuojama duomenų bazėje ir galės būti panaudota naujoje iteracijoje (1 pav.).
2. **Adaptyvaus sprendimo priėmimo įvertinimas.** Keliami klausimai: ar priimti sprendimai bus produktyvūs? ar jie atitiks besimokančiojo tikslus? ar jie bus nukreipti teisinga linkme? Visa tai taip pat galėtų būti vertinama tikslingu testu. Kad įvertintume tikslais paremtą adaptavimą, galima suformuoti vartotojui individualų tikslą. Įvertinus ir nustatčius, kad adaptyvumas duos norimą rezultatą, modelio informacija gali būti užfiksuojama duomenų bazėje ir galės būti panaudota naujoje iteracijoje (1 pav.).

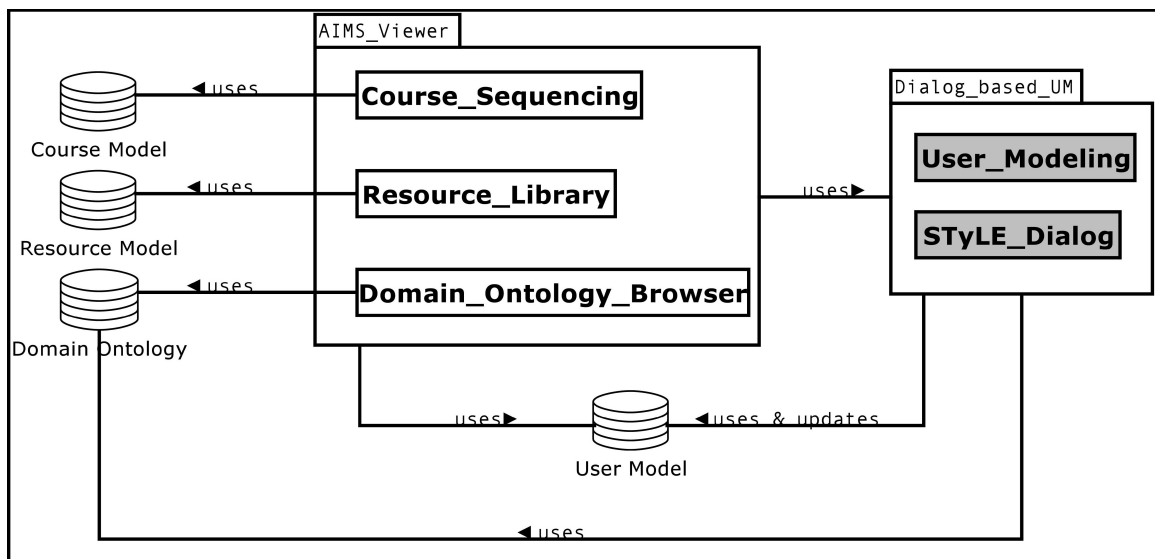
Nors čia siūloma schema nėra labai sudėtinga, tačiau besimokančiojo modelio vietą sistemoje galima išreikšti dar paprasčiau. Visiškai paprastą ir abstraktų besimokančiojo modelį pateikia savo straipsnyje Grace de la Flor [25]. Šį modelį (2 pav.) autorius vadina „vartotojo modeliu“, neatsižvelgdamas į konkrečią modelio taikymo sritį. Tai gana svarbu norint įsivaizduoti tokio modelio vietą sistemoje aukštesniame abstrakcijos lygmenyje.



2 pav. Grace de la Flor siūlomas paprasčiausias vartotojo modelis [25]

Panašios besimokančiojo ar vartotojo modelio schemas aprašomos daugelio straipsnių autorių. Nesigilinant į smulkmenas, tai gali būti tinkama struktūra nuotolinių studijų sistemos besimokančiojo modeliui vaizduoti. Tačiau išgilinus toks vaizdas visiškai nenusako vidinės modelio vaizdavimo struktūros ir išskylančių problemų, todėl toliau panagrinėsime sudėtingesnius besimokančiojo modelio vaizdavimo atvejus.

Vania Dimitrova yra pateikusi nemažai straipsnių, kelis iš jų – distancinio mokymo(si) tematika [21]. Autorės siūlomas besimokančiojo modelis (3 pav.), skirtas mokomosios medžiagos peržiūrėjimo sistemai, yra šiek tiek sudėtingesnis ir sudarytas remiantis ontologijų taikymu.



3 pav. Besimokančiojo modelis panaudotas sujungiant dvi mokymuisi skirtas sistemas [18]

Diagramoje (3 pav.) pateikiamas gana sudėtingas besimokančiojo vaizdavimas ir papildomai yra pateikiami resursų ir kursų modeliai. Pakitimai besimokančiojo modelyje vyksta palaipsniui ne

tik kurso pradžioje ir pabaigoje, bet ir laikinai keičiasi tam tikrose stadijose. Reikalingas mechanizmas šiems pokyčiams nustatyti. Didelė dalis problemų autorės straipsnyje išspręsta sujungiant dvi skirtingais principais veikiančias sistemas, AIMS ir STyLE-OLM. AIMS skirta mokomosios medžiagos peržiūrėjimui, mokomoji medžiaga yra modifikuojama remiantis vartotojo modeliu. AIMS pagrinde naudoja kurso modelį, resurso modelį ir pagrindines ontologijas. AIMS remiasi vartotojo modeliu ir pagal jį sudaro pasiūlymus studentui, kokia kryptimi jam reikėtų mokytis: pagal tai, ką studentas mokėsi, kiek mokėsi ir kiek išmoko, yra pakeičiamas resurso modelis. Taip pat visa, ką studentas turi išmolti, yra suskirstyta į konkrečias sąvokas, terminus. Taip galima sukcentruoti studento dėmesį į tam tikras resurso vietas naudojant pagrindines ontologijas, pasirenkant iš jų bendro sąrašo ir studento modelio tas, kurios dar neįsisavintos. Tuos terminus nustato dėstytojas, šioje vietoje dėstytojas nustato mokymosi tikslą, kuris turėtų nukreipti studentą teisinga mokymosi linkme, tačiau taip būna ne visada, nes studentas gali turėti savo visai kitokią tikslą ir mokymosi įpročius.

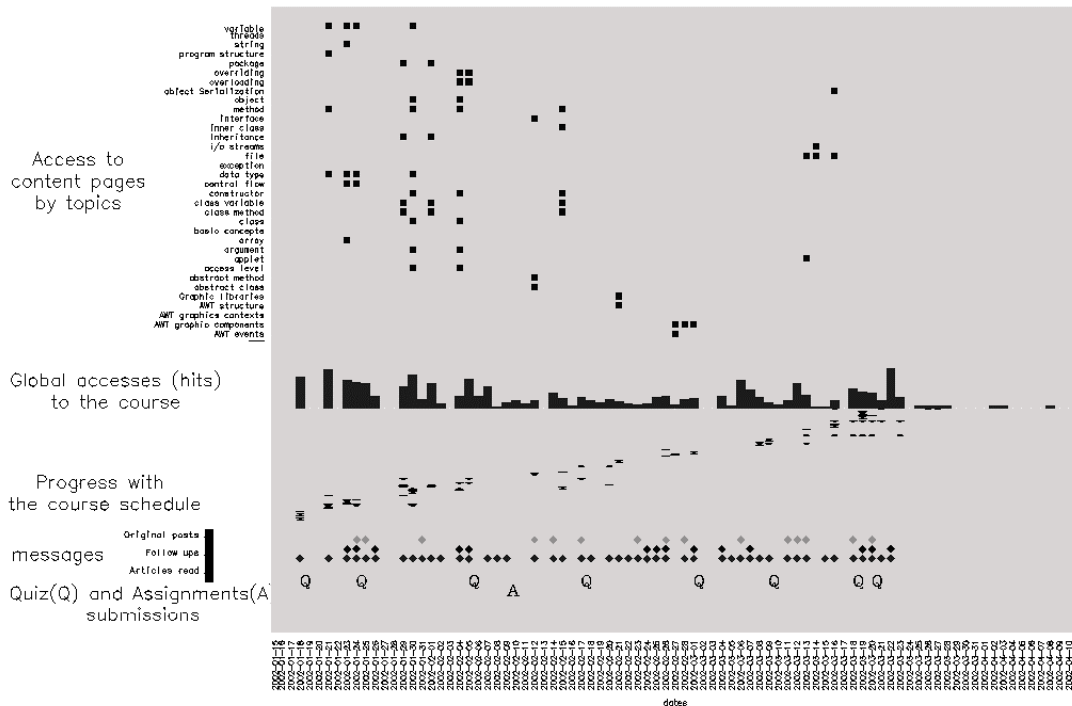
STyLE-OLM yra sukurta specialiai studento modelio formavimui ir yra visiškai nepriklausoma nuo konkrečios sistemos, ji skirta tik modelio suformavimui [18]. Ji buvo pritaikyta keliose sistemose (finansų skaičiavimo srityje). Ji remiasi dialogu su besimokančiuoju – šis dialogas formuoja pagrindines ontologijas ir dialogo modelį.

Sujungimas vykdomas taip: dialogu paremta sistema formuoja tą patį studento modelį, kurį naudoja AIMS, tai ji daro pagal modelį sudarytą iš dialogų. Taip išvengiama „šalto starto“ problemos bei paliekama galimybė studentui koreguoti savo modelį dialogo pagalba; taip pat STyLE-OLM keičia pagrindines ontologijas, t.y. dėstytojo nustatytą mokymosi tikslą, kryptį, šioje vietoje išvengiama dėstytojo ir studento semantikos nevienodumo problemos.

Kitame tos pačios autorės straipsnyje sprendžiama besimokančiojo modelio vaizdavimo problema nuotolinių studijų sistemoje, tačiau čia nagrinėjamas besimokančiojo modelio vaizdavimas iš dėstytojo perspektyvos – jis skirtas dėstytojui įvertinti besimokančiojo veiklą ir pasiekimus dirbant su nuotolinių studijų sistema [19]. Nors visa tai yra skirta dėstytojui, o ne studentui, tačiau panašią grafinę terpę galima būtų naudoti ir studentui savo paties darbo įvertinimui.

Summary of student's behaviours

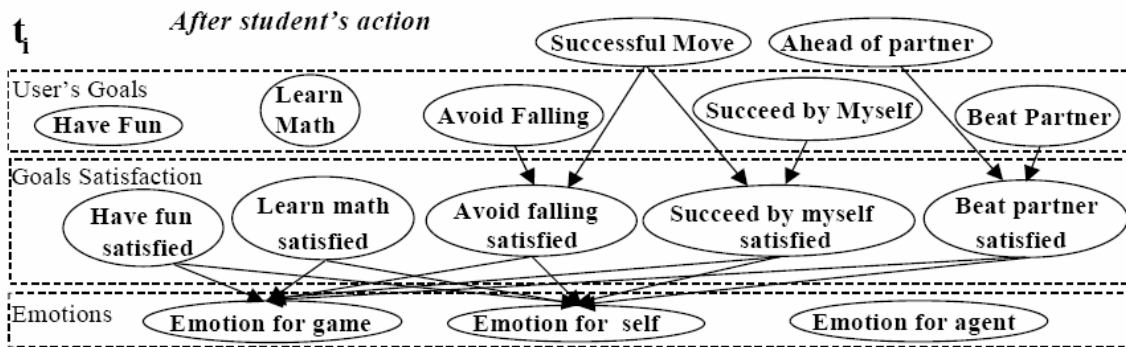
from 2002-01-15 to 2002-04-11
Student: "francesco"



4 pav. Grafinis besimokančiojo modelio vaizdavimas nuotolinių studijų sistemoje [19] (žr. prieduose)

Paveikslėlyje (4 pav.) galima matyti ir interpretuoti studento mokymosi procesą. Tai labai naudingas ir vaizdus būdas dėstytojui įvertinti studento darbą. Diagramoje vaizduojami testų rezultatai, dalyvavimas forumuose, laikas, praleistas kursuose, kurso dalys, kurios buvo aktyvuotos ir visa ši informacija perteikiama laiko juostoje. Tačiau studentui tokių grafinių duomenų pateikimas gali neduoti norimos naudos: studentui tai gali būti per daug sudėtinga informacija, kad jis galėtų priimti teisingą sprendimą. Dėstytojas vis dėlto atlieka ir edukacijos specialisto funkciją, jis turi žinių ir patirties, kuriomis remdamasis gali analizuoti tokią grafinę informaciją.

Straipsnyje [11] aprašomas visiškai kitoks besimokančiojo modelio sudarymo būdas. Aprašomas galimo besimokančiojo modelio sudarymas pagal tam tikrus parametrus, nustatytus specialaus agento, kuris realizuotas žaidimo forma. Šioji yra paremta tikslų, tikslų įgyvendinimo ir emocijų modeliu (5 pav.).



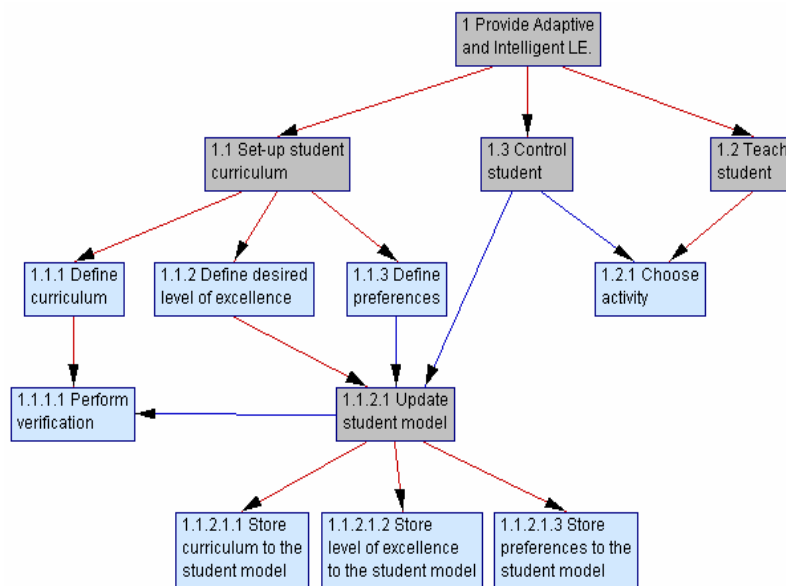
5 pav. Modelis, remiamas besimokančiojo tikslų, tikslų įgyvendinimo ir emocijų modeliu [16]

Tačiau tai nėra labai priimtinas modelio formavimo būdas nuotolinių studijų sistemoje, visų pirma dėl to, kad dėl nagrinėjamos sistemos technologinės vidinės bazės. Ir gana sudėtinga būtų rimtai pagerinti mokymosi procesą naudojant vien tik agentą ir žaidimo formą studento modeliui formuoti, tačiau idėja verta dėmesio.

Yra nemažai straipsnių, siūlančių netradicinius besimokančiojo modelio sudarymo būdus. Gana įdomus pavyzdys galėtų būti toks: aprašomas vartotojo modelio sudarymas remiantis studento žvilgsnio stebėjimu, nustatant ties kuriuo objektu studentas labiausiai sutelkia dėmesį; šis būdas yra naudojamas psichologijoje [12]. Toks modelio sudarymo būdas galimas naudojant šiuolaikines technologijas, tačiau sunkiai praktiškai pritaikomas nuotolinėse studijų sistemose.

Besimokančiojo modeliui perteikti tink įvairios priemonės ir duomenų rinkiniai, tačiau reikia pabrėžti, kad šis darbas yra testinis ir jau yra suformuota terpė nuotolinių studijų sistemoje, skirta besimokančiojo modelio įdiegimui. Kadangi besimokančiojo modelio atvaizdavimas yra tik dalis visos adaptyvios sistemos, remiamasi bendra strategija, suformuluota D. Baziukaitės [2]. Šioje strategijoje yra apibrėžiama besimokančiojo modelio panaudojimo savarankiškose nuotolinėse studijose galimybė, modeliai suformuoti taikant Mase metodiką. Trumpai aptarsime, kaip besimokančiojo modelis įsivaizduojamas bendru atveju, kartu su kitų adaptyvių priemonių rinkiniu.

Šiame darbe toliau remsimės jau suformuluotu nuotolinės adaptyvios sistemos tikslų modeliu (6 pav.). Kadangi dalis sistemos jau yra paruošta, modifikuota pagal konkrečius modelius bei prielaidas, darbas ir toliau bus orientuojamas ta pačia kryptimi. Suformuluoti modeliai ir teorinis aspektas nusako besimokančiojo modelio vietą, visoje sistemoje, tačiau nenusako metodų bei technologijų, reikalingų realizacijai, nenusako pačios modelio struktūros. Idėjos, kuriomis remiamasi, yra išdėstytos straipsnyje [2].



6 pav. Sistemos tikslų hierarchija [2]

1.3. Problemų, kurias siekiama spręsti perteikiant besimokančiojo modelį, formulavimas

Dažnai išskyla „Šalto starto“ problema. Adaptyvios sistemos susiduria su pirmą kartą prisijungusio vartotojo problema, kai dar neegzistuoja apibrėžtas besimokančiojo modelis. Šiuo atveju yra reikalingos greito modelio formavimo technikos – joms dažniausiai naudojami verifikacijos (pradinio žinių patikrinimo) testai; ankstesniame skyriuje buvo pateikti keli netradiciniai šios problemos sprendimo būdai (pvz., pradinis dialogas) [9,19,40].

Gana dažna problema, išskylanti veikiančiose adaptyviose sistemose ir aptariama ne vieno straipsnio autorių, yra jau **turimų žinių ir patirties naudojimas** [9]. Tas pat dokumentas gali turėti skirtingą įtaką skirtingiems žmonėms priklausomai nuo anksčiau įgytų žinių. Reikalingi metodai anksčiau įgytų žinių diagnostikai.

Vartotojo modeliai, paremti tiktai besimokančiojo-sistemos sąveikos analize, yra netikslūs. Realiame gyvenime dėstytojai naudoja numanomas būdas tam, kad įvertintų mokinio gabumus ir rezultatus. Mokymo sistemoms reikalingi tikslūs, apibrėžti mechanizmai, kad paaiškintų gautus besimokančiojo modelius [19,40].

Kurso autoriaus ir besimokančiojo semantika gali būti skirtingos. Ši problema gali lemti netikslią mokinio diagnozę arba netinkamą svarbios medžiagos rekomendaciją. Be to, šis skirtumas paaiškina, kodėl negalima pasikliauti vien tik besimokančiojo nuomone, paremta jo turimomis žiniomis, ir kodėl reikalingas aiškinamasis dialogas [19,40].

Kurso autoriaus tikslai ir besimokančiojo tikslai gali skirtis. Dauguma mokymo sistemų remiasi prielaida, kad šie du tikslai yra vienodi, todėl jos nesitaiko prie galimų specifinių mokinio tikslų. Mechanizmai, aiškinantys mokinio tikslus, yra svarbūs efektyviam rekomendavimui ir vadovavimui [19,40].

Besimokančiojo tikslai, gabumai ir žinios keičiasi, todėl iškyla besimokančiojo modelio kintamumo problema. Taip pat gali būti keliamas klausimas, kaip išvengti per didelio adaptyvumo, kai jis jau tampa neveiksmingas, kai adaptyvumo nebuvimas yra efektyvesnis [9,40].

Originalus „šalto starto“ testų nepopuliarumo problemos sprendimas [19] galėtų būti paremtas siūlymu naudoti visos grupės modelį: taip būtų galima išvengti ir “šalto starto problemos” ir kitų sunkumų.

Pagrindinė problema, skatinanti kelti adaptyvių komponentų reikalingumo nuotolinių studijų sistemose klausimą, yra ta, jog negzistuoja integruoto adaptyvaus mokymo(si) modulis nuotolinių studijų sistemose. Aprašomas tik dalinis adaptyvumas WebCt ir Blackboard sistemose [50].

Tai yra bendro pobūdžio ir restruktūrizacijos problemos. Svarstant sistemos realizacijos aspektus, galima išskirti **įgyvendinamumo problemas**, t.y. programinės įrangos galimybių ribotumą. Tačiau pagrindinė, šiame darbe bene svarbiausioji problema – **adaptyvių komponentų sudėtingumas**. Visų nuotolinių sistemų pagrindinis tikslas yra kiek galima labiau minimizuoti vartotojo sąsajos sudėtingumą, kadangi tobulėjant sistemai, atsirandant naujoms idėjoms ir sprendimams, jos komponentų kiekis didėja – baiminamasi, kad tai gali kelti tam tikrą sumaištį naudojantis sistema. Be to, edukologijos specialistai griežtai kritikuoja sistemos, kuri siekia pakeisti specialistą, koncepciją.

Toliau darbe bandysime giliau ir detaliau pažvelgti į šias problemas, bei į galimus sprendimus.

1.4. Besimokančiojo modelio panaudojimo galimybės ir ribos

Šiek tiek plačiau aptarsime besimokančiojo modelio galimybes bei jas ribojančius aspektus. Visų pirma reikia atkreipti dėmesį į tai, kad nuotolinių studijų sistemos naudojamos tinkle t. y. Internetu. Tai iš dalies apriboja besimokančiojo ir sistemos bendradarbiavimo galimybes, kadangi sistemą ir besimokantį skiria duomenų persiuntimo sąnaudos. Taip pat nepageidaujami yra sudėtingi algoritmai, nes sistemos vartotojų dažnai yra labai daug, todėl bet koks apkrovimas sukels pavojingus trikdžius.

Be šių pagrindinių techninių apribojimų dar reikia atkreipti dėmesį, kad modelio realizacijai bus naudojama „Moodle“ sistema, kuri yra realizuota PHP kalba su MySQL duomenų baze. Siekiant išlaikyti nuoseklumą ir modifikavimo galimybę, bus laikomasi bendrų „Moodle“ sistemoje nusistovėjusių programavimo bei duomenų saugojimo principų.

1.5. Galimi sprendimo būdai

Atlikus analizę, galima išskirti galimus besimokančiojo modelio atvaizdavimo sprendimo būdus. Pirmiausia atkreipsime dėmesį į grafinės informacijos atvaizdavimą. Grafinė terpė leidžia sutalpinti ir efektyviai pateikti didelius informacijos kiekius [19]. Jei pateikiamos diagramos yra nesudėtingos, palydimos tiksliais, skaičiavimais paremtais komentarais, jos turėtų besimokančiajam suteikti didelį postūmį savianalizės procese. Čia galima išvelgti tam tikrą požiūrį į patį modelį. Kadangi sakoma, jog sistema negali visiškai atstoti dėstytojo, pažvelkime į šią problemą kiek kitokiu kampu. Besimokančiojo modelis ne pakeis dėstytoją, bet perkels dalį jo sprendimų besimokančiajam, pateiks informaciją taip, kad besimokantysis nesunkiai galėtų vertinti savo mokymosi procesą pats.

Apibrėžkime pačių duomenų modelyje pobūdį. Pirmiausia, tai – **sekimo duomenys, testavimo rezultatai ir besimokančiojo pasirinkimai**. Adaptyvumas grindžiamas tuo, kad kiekvieno besimokančiojo atveju šie duomenys bus skirtingi. Modelio atvaizdavimui naudojamų skaičiavimų tikslumas bei atvaizduoto modelio suprantamumas nulems prisitaikančio kurso efektyvumą. Pats adaptyvumo procesas bus pasiekiamas bendraujant su besimokančiuoju per vartotojo sąsają, kurioje nemažai nustatymų besimokantysis nustatys pats [16].

1.6. Skyriaus apibendrinimas

Skyriaus pradžioje aptarti pagrindiniai vartojami terminai. **Besimokantysis** – tai studentas ar moksleivis, kuris savarankiškai nori siekti naujų žinių, turi jau sukauptą mokymosi patirtį, bei turi kompiuterinio raštingumo įgūdžių. **Mokymosi efektyvumas** grindžiamas mokymosi aplinkos prisitaikymu studento individualiems poreikiams. **Besimokančiojo modelis** suprantamas kaip žinių ir duomenų rinkinys apie besimokantįjį, į kurį įeina: kurso planas, lygio pasirinkimas, „sekimo“ bei testavimo duomenys, bei juos atvaizduojančios priemonės, suprantamos besimokančiajam.

Darbe bus remiamasi prielaida, kad studento mokymo(si) aplinkos gebėjimas prisitaikyti prie studento poreikių gali sumažinti prarają tarp dėstytojo ir besimokančiojo. Suformuluotas požiūris, kad dauguma VMA įrankių yra sudaryti taip, kad dėstytojas galėtų stebėti besimokantįjį, analizuoti

mokymo(si) procesą ir jam padėti, o pačiam studentui analizuoti savo mokymą(si) bei rezultatus nėra galimybių.

Literatūroje aprašomi keli besimokančiojo modelio struktūros variantai, vienas iš bendresnių galėtų būti toks:

Duomenys apie besimokantįjį → sujungimas į modelį → besimokančiojo modelis → adaptacija → adaptacijos efektas [25]

Literatūroje taip pat aprašomi netradiciniai modelio formavimo būdai, pvz., akies žvilgsnio krypties fiksavimas studento modelyje ar verifikacijos atlikimas žaidimo forma [16]. Yra pateikiamas gana tikslus ir dažniausiai naudojamas besimokančiojo modelio formavimo metodas [9].

Darbe iškeltos problemos [19,40] ir klausimai, susiję su besimokančiojo modelio įdiegimu:

- Kokia informacija turi sudaryti modelį, o kas yra nereikalinga;
- Kokius metodus panaudoti tam, kad modelį padarytume suprantamą sistemai;
- Kaip kuo aiškiau pateikti sistemos išvadas besimokančiajam;
- „Šalto starto problema“;
- Vartotojo modeliai, paremti tikrai besimokančiojo – sistemos sąveikos analize, yra netikslūs;
- Kurso autoriaus ir besimokančiojo semantika gali būti skirtingos;
- Kurso autoriaus tikslai ir besimokančiojo tikslai gali skirtis;
- Besimokančiojo tikslai, gabumai ir žinios keičiasi.

Taip pat buvo aptartos besimokančiojo modelio panaudojimo galimybės ir ribos.

2. Besimokančiojo modelio naudojimas nuotolinių studijų efektyvumui padidinti

Kaip teoriškai aprašomas besimokančiojo modelis galėtų būti realizuotas nuotolinių studijų sistemoje, gana nelengva išivaizduoti. Duomenys, sudarantys modelį, be žinių nieko verti. Modelyje reikia suformuluoti skaičiavimus ir algoritmus, reikalingus duomenų atvaizdavimui, bei sukurti vartotojo sąsają bendravimui. Visa tai turi tilpti į nuotolinės sistemos rėmus, todėl reikia įvertinti, kokias technologijas ar metodikas galima naudoti ir kokie skaičiavimai duotų efektyvius rezultatus. Šiame skyriuje aptarsime technologinius sprendimus, įgalinančius besimokančiojo modelio realizaciją nuotolinių studijų sistemoje.

2.1. Naudojamų technologijų galimybių įvertinimas

Darbo rezultatams gauti naudosime tam tikras technologijas, todėl būtina aptarti šių technologijų galimybes.

PHP ir MySQL technologijos sudaro tam tikrus rėmus būsimai modifikacijai. Sprendimo rengimo sistemų naudojamos duomenų skaičiavimų technikos nereikalauja didelių laiko sąnaudų ir galėtų būti taikomos informacijos apdorojimui. Interaktyviai vartotojo sąsajai gauti plačias galimybes suteikia Javascript technologija.

Grafinės informacijos vaizdavimui PHP kalboje yra pakankamas procedūrų rinkinys diagramoms ar sudėtingesniems grafiniams komponentams kurti.

MySQL terpė suteikia pakankamą erdvę reikalingiems duomenims saugoti.

2.2. Adaptyvumo samprata besimokančiojo modelio vaizdavime

Kaip jau minėjome anksčiau, adaptyvumo samprata įvedama kaip prie besimokančiojo poreikių prisitaikančio turinio panaudojimas nuotolinėse studijose. Pats adaptyvumas gaunamas besimokančiajam bendraujant su sistema. Bendravimas šiuo atveju suprantamas kaip tam tikrų pasirinkimų registravimas duomenų bazėje, sekimo duomenų bei testavimo rezultatų saugojimas [16,17].

Pats prisitaikymas vykdomas pagal besimokančiojo pasirinkimus, besimokančiajam suteikiant tam tikrus pasiūlymus ir mokymosi proceso analizės rezultatus, pagal kuriuos besimokančiajam lengviau nuspręsti, kaip turėtų pasikeisti mokymosi aplinka.

Pagal patį adaptyvumo procesą galima skirti du adaptyvumo principus. Pirmasis – tai su vartotojo veiksmis susijęs sistemos prisitaikymas prie besimokančiojo. Besimokantysis per vartotojo sąsają registruoja savo pasirinkimus, prie kurių sistema ir prisitaiko. Kitoks adaptyvumas gaunamas panaudojus besimokančiojo modelį, kur sistema prisitaiko prie besimokančiojo, remdamasi iš modelio pasikeitimų gautais duomenimis [16,17].

2.3. Lanksti programuojamos terpės architektūra

Realizacijai keliami architektūriniai reikalavimai: visų pirma – programavimo stiliaus, naudojamo „Moodle“ kūrėjų, išlaikymas; antra – pakartotinis panaudojimas. Pastarasis būtinas, todėl, kad posistemės realizacijos siekiamas rezultatas yra prototipas, kuris – atlikus bandymus su realiais vartotojais – bus modifikuojamas. Dėl šių priežasčių programos kodas turi būti atviras pokyčiams, todėl turi būti laikomasi šablonų struktūros, naudojama objektinė technologija.

Šios strategijos bus laikomasi realizacijos proceso metu.

2.4. Agentinės technologijos nuotoliniame el. mokyme

Agentinės technologijos glaudžiai susijusios su adaptyviomis sistemomis [37,42]. Galutinio ir vieningo agento apibrėžimo nėra, tačiau jį galima supaprastintai apibūdinti kaip programinį komponentą, turintį aplinkos sensorius, tikslą bei įrankius aplinkai paveikti. Tam tikromis agentinėmis savybėmis pasižymi ir planuojama modifikacija.

2.5. Sprendimų rengimo sistemos skaičiavimų panaudojimo eksperimentas

Sprendimo rengimo sistemų (SRS) skaičiavimų galimam panaudojimui buvo atliktas eksperimentas, kuriuo siekiama įvertinti tokių skaičiavimo sistemų efektyvumą.

Pamėginkime įvertinti, ar SRS yra reikalingos nuotolinėms studijoms, ir pamėginkime suformuluoti konkrečias priežastis, kodėl reikėtų taikyti tokius skaičiavimus.

Pirmiausia galime teigti, kad nuotolinėse studijose mokymas nėra pakankamai lankstus ir neprilygsta realiam besimokančiojo bendravimui su dėstytoju. Čia galima formuluoti pirmąją priežastį:

- Nuotolinių studijų kokybė kol kas neprilygsta individualizuotam mokymui;

Problemos, kurias išskėlėme, nėra paprastai išsprendžiamos. Jų sprendimas yra sudėtingas, kadangi reikia remtis didelių kiekių kriterijū, o mokomosios medžiagos kiekis yra didelis:

- Problemoms spręsti reikia taikyti sudėtingus algoritmus;

Taip pat galima paprieštarauti, kad sistema sprendimų negali priimti pati; tačiau planuojama neapsiriboti tik SRS integracija – SRS bus tik dalis vadinamojo studento modelio, kuri turėtų sudaryti ir grafinę informaciją, ir įvairūs papildomi įrankiai. Be to, galutinį sprendimą priims pats studentas:

- SRS kaip viena iš priemonių gali būti pakankamai patogus būdas priimti sprendimus.

Pirmiausia pateiksime Analitinio Hierarchinio Proceso (AHP) modelį [8]. Duomenys modelyje yra išgalvoti. Žinių ir testų vertinimai parinkti pagal nuotolinės studijų sistemos duomenų bazę. Modelyje naudojami kriterijai iš duomenų bazės:

- Testų rezultatai (matuojami 100 balų skalėje)
- Prisijungimų skaičius prie mokomosios medžiagos (vnt.)
- Mokymosi laikas (matuojamas minutėmis)
- Atliktų užduočių įvertinimas (matuojamas 100 balų skalėje)
- Papildomos medžiagos skaitymo prisijungimų skaičius (vnt.)
- Papildomos medžiagos skaitymo laikas (matuojamas minutėmis)

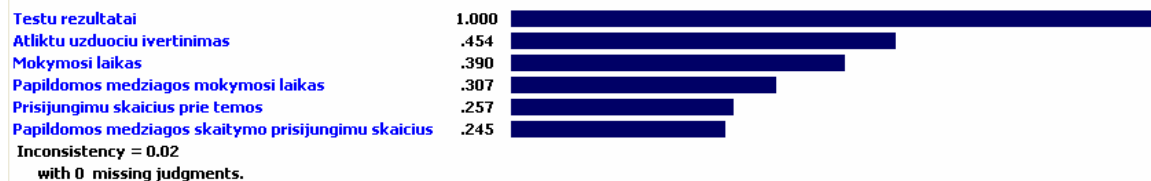
Pirmiausia nustatome kriterijų įverčius (7 pav.). Šiuo atveju įverčiai buvo parinkti kaip pavyzdys. Atsižvelgta į tai, kad testo rezultatai turi didžiausią svarbą. Tačiau iškilo problema, kad galbūt testo rezultatus reikėtų naudoti kaip atskirą kriterijų, pagal kurį studentas galėtų lyginti savo mokymosi rezultatus, nustatytus naudojantis SRS.

	Testu rezu	Prisijungim	Mokymosi	Atliktu uzdu	Papildomo	Papildomo
Testu rezultatai		5.68	2.17	1.52	3.67	3.93
Prisijungimu skaičius prie temos			1.55	1.58	1.17	1.0
Mokymosi laikas				1.39	1.74	1.23
Atliktu uzduociu ivertinimas					1.39	1.2
Papildomos medžiagos skaitymo prisijungi						1.47
Papildomos medžiagos mokymosi laikas	Incon: 0.02					

7 pav. Įverčių nustatymas

Įverčiai nustatomi lyginant atskirus kriterijus vieną su kitu, t.y. lentelėje matomi koeficientai yra gaunami suteikiant vienam kriterijui tam tikrą svorį prieš kitą kriterijų. Šis procesas atliekamas su visais kriterijais (7 pav.).

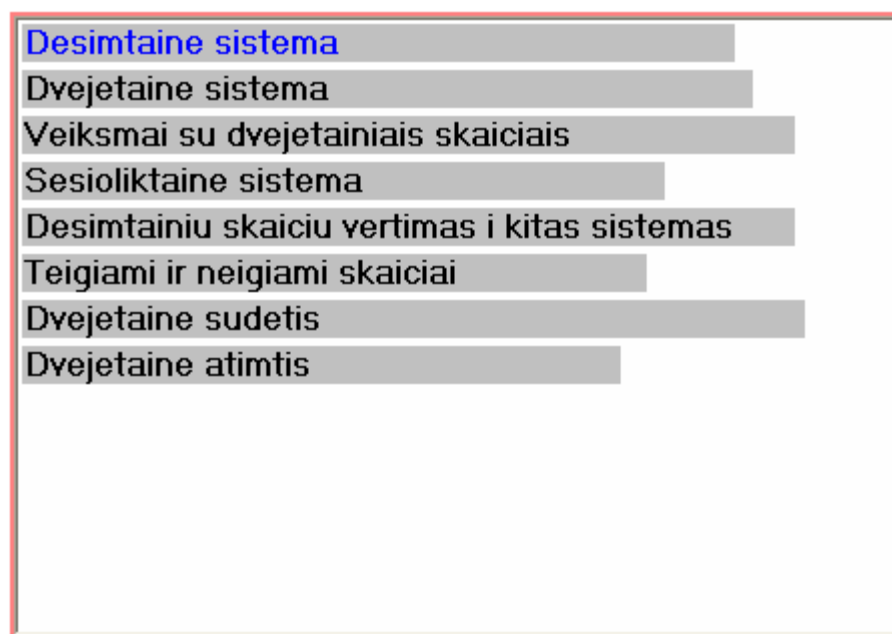
Toliau surikiuosime kriterijus pagal apskaičiuotus prioritetus, bei normalizuosime įverčių skalę (8 pav.).



8 pav. Įverčių rūšiavimas ir normalizavimas

Normalizavus kriterijus ir surūšiuavus juos pagal įverčius, galime labai aiškiai pamatyti, kokie mokymosi kriterijai yra svarbiausi ir turės daugiausiai įtakos priimant sprendimus.

Suformavus kriterijus ir atlikus jų vertinimą, vykdomas alternatyvų vertinimas kiekvieno kriterijaus atžvilgiu (9. pav). Vertinimas nustatomas pagal numatytus matavimo vienetus. Šiuo atveju alternatyvos – tai mokomosios medžiagos skyriai ir temos, medžiaga suskirstyta pakankamai smulkiai, o testavimo rezultatai susiję su konkrečia tema ar skyriumi.



9 pav. Temomis suskirstyta mokomoji medžiaga

Pademonstruosime kelis temos įvertinimo parinkimo langus.

	Desimtaine	Dvejetainė	Veiksmai s	Sesioliktai	Desimtaini	Teigiami ir	Dvejetainė	Dvejetainė
Desimtaine sistema		3.0	2.0	1.09	2.41	1.21	1.19	1.28
Dvejetainė sistema			2.0	1.22	1.13	1.31	1.33	1.16
Veiksmai su dvejetainiais skaiciais				5.0	1.18	1.15	1.28	1.33
Sesioliktaine sistema					3.0	1.13	1.15	1.16
Desimtainiu skaičiu vertimas i kitas sistemas						4.0	1.19	1.48
Teigiami ir neigiami skaciai							3.0	1.17
Dvejetainė sudėtis								2.0
Dvejetainė atimtis	Incon: 0.11							

10 pav. Įverčiai parinkti pagal testų rezultatus

Paveikslėlyje (10 pav.) parenkami įverčiai dabar jau mokomosios medžiagos temoms. Kiekviena tema viena kitos atžvilgiu įgauna prioritetą. Toks įverčių formavimas yra labai patogus ir tikslus, kadangi vertinami ne visi komponentai iš karto, o atskirai vienas prieš kitą.



11 pav. Testo rezultatų normalizacija

Toliau (11 pav.) atliekama komponentų normalizacija ir rūšiavimas. Temos išrikiuojamos pagal gautus rezultatus po kryžminio svorių suteikimo. Pateiksime dar kelis temų įverčių rezultatus (12, 13, 14, 15 pav.)

	Desimtaine	Dvejetainė	Veiksmai s	Sesioliktai	Desimtaini	Teigiami ir	Dvejetainė	Dvejetainė
Desimtaine sistema		2.0	3.0	3.0	3.0	2.0	2.0	2.0
Dvejetainė sistema			3.0	3.0	4.0	3.0	4.0	2.0
Veiksmai su dvejetainiais skaiciais				3.0	3.0	4.0	4.0	3.0
Sesioliktaine sistema					2.0	6.0	3.0	3.0
Desimtainiu skaičiu vertimas i kitas sistemas						1.0	5.0	3.0
Teigiami ir neigiami skaciai							2.0	4.0
Dvejetainė sudėtis								2.0
Dvejetainė atimtis	Incon: 0.43							

12 pav. Įverčiai parinkti pagal mokymosi laiką

Kadangi įverčiai parenkami temoms, šis procesas kartojamas kelis kartus. Kartojame tol, kol neparenkame įverčių kiekvienam mokymosi rezultatų komponentui (7,8 pav.).



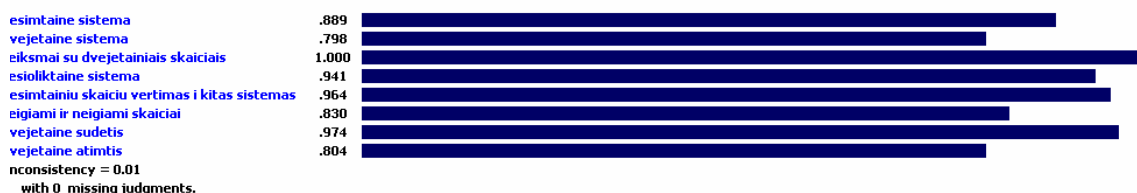
13 pav. Temos rezultatų normalizacija

Galima atkreipti dėmesį, kad skirtingų mokymosi rezultatų atžvilgiu temos įgauna skirtingas reikšmes ir išrikiuojamos skirtingai (15,13,11 pav).

	Desimtaine	Dvejetai	Veiksmai s	Sesioliktai	Desimtaini	Teigiami ir	Dvejetai	Dvejetai
simtaine sistema		1.19	1.28	1.06	1.0	1.21	1.07	1.29
ejetaine sistema			1.27	1.2	1.63	1.2	1.35	1.32
ksmai su dvejetainiais skaiciais				1.13	1.07	1.43	1.28	1.28
sioliktaine sistema					1.27	1.29	1.06	1.12
simtiniu skaičiu vertimas i kitas sistemas						1.37	1.29	1.23
giami ir neigiami skaiciai							1.15	1.56
ejetaine sudetis								1.17
ejetaine atimtis	Incon: 0.01							

14 pav. Įverčiai parinkti pagal atliktas užduotis

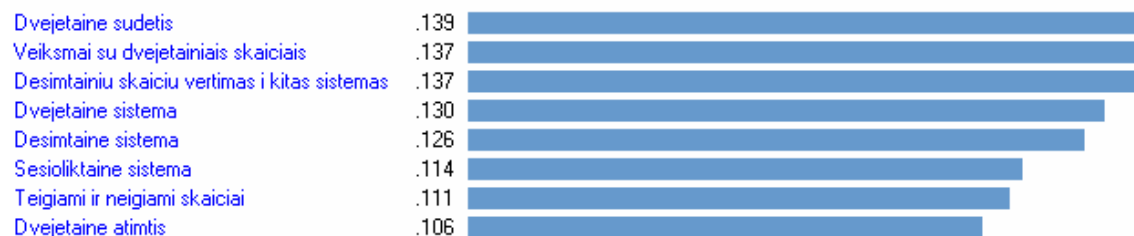
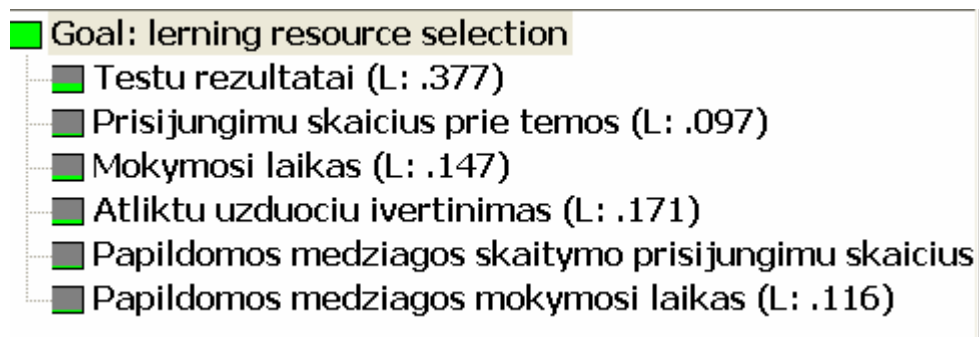
Atlikus įverčių priskyrimą pagal paskutinius mokymosi rezultatus, šiuo atveju (14 pav.) pagal atliktas užduotis, turime kelis temų įverčių rinkinius, pagal kuriuos toliau galėsime apskaičiuoti ir priimti sprendimą, kuri tema buvo išmokta blogiausiai ar geriausiai.



15 pav. Temos rezultatų normalizacija

Nustatome, kuri tema buvo išmokta geriausiai, ir kuri blogiausiai (16 pav.). Didžiausią įtaką padarė testų laikymas, todėl medžiagos mokymasis gali būti įvertintas netiksliai. Pasinaudojus papildomomis priemonėmis (dinamine struktūra ir kt.) galima palyginti, koks būtų rezultatas, jei kriterijai būtų kitokie (16 pav.).

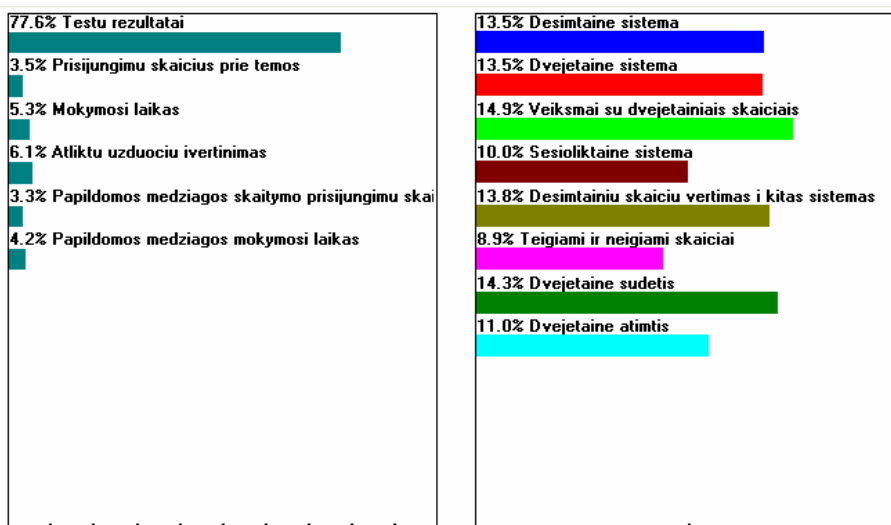
Čia naudojama speciali sistema, kuri skirta spręsti problemas AHP metodu. Galima atkreipti dėmesį į tai, kokius AHP komponentus bei papildomus įrankius galima būtų įtraukti (ar neįtraukti) į modeliuojamąjį SRS. Patys skaičiavimai nėra sudėtingi ir nereikalauja laiko sąnaudų, didžiausia problema yra prioritetų priskyrimas, kaip nustatyti kokie prioritetai geriausiai atitinka komponentus ir kaip dėstytojui nepridėti papildomo darbo su nuotoline sistema.

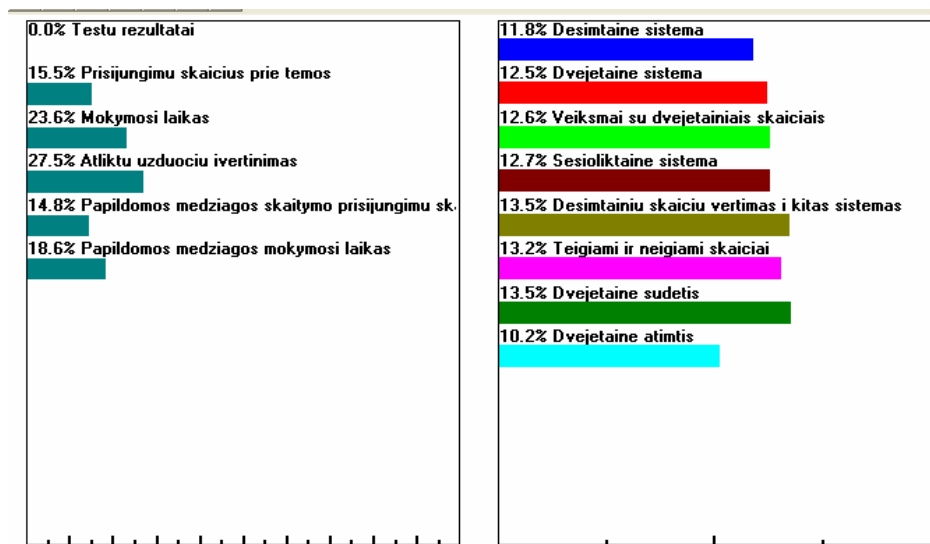


16 pav. Galutiniai rezultatai

Matome, kad geriausiai buvo išmokta dvejetainių skaičių sudėtis, o blogiausiai – dvejetainių skaičių atimtis (16 pav.), tačiau akivaizdu, kad buvo naudojami netikri duomenys, kadangi dvejetainė sistema neišmokta, o dvejetainių skaičių sudėtis išmokta.

Pateiksime papildomą dinaminį modelį, kad būtų matoma testų rezultatų įtaka mokymosi vertinimui (17 pav.).





17 pav. Dinaminis modelis

Kaip matome, mokymosi proceso vertinimas ir testų rezultatų vertinimas gana smarkiai skiriasi. Todėl reikėtų testų rezultatus lyginti atskirai. Savaiame aišku, skirsis ir rezultatai, gauti remiantis kitais kriterijais, tačiau šios problemos sprendimo tikslas yra surasti mažiausiai išmoktas ir mažiausiai suprastas mokomosios medžiagos dalis. Kiti modeliai bus formuluojami taip, kad būtų galima nustatyti, kodėl ta medžiaga buvo išmokta blogai.

Modeliuojant galimą SRS taikymą besimokančiojo modelio formavimui buvo naudota Expert Choice sistema, tiksliau jos “Trial” versija. Ši sistema leidžia patogiai taikyti AHP metodą, tačiau paties įrankio naudoti nuotolinių studijų sistemoje nėra galimybių.

AHP labai praverstų nuotolinių studijų sistemoje besimokančiam studentui [8] – nors tai nėra tradicinis SRS taikymas, tačiau tai turėtų palengvinti mokymąsi. Toks įrankis veikiausiai pagreitintų mokomosios medžiagos įsisavinimą, tačiau jo negalima kurti be specialistų konsultacijų, kadangi mokymosi procesas – sudėtingas psichologinis reiškinys. Taip pat besimokančiojo modeliui formuluoti neužtenka vien tik integruoti SRS į nuotolinių studijų sistemą – tai galėtų būti tik vienas iš besimokančiojo modelio komponentų.

Realizacijai panaudota tik nedidelė AHP komponentų dalis. Kadangi mes kol kas priimsime sprendimus naudodamiesi tik dviem aspektais – testų rezultatais ir mokymosi intensyvumu – skaičiavimas tampa gana paprastas. Be to, skirti prioritetus kol kas yra gana sudėtinga – juos galima nustatyti tik su specialisto pagalba arba atlikus bandymus su realiais respondентаis. Todėl kol kas apsiribosime duomenų normalizacija bei nesudėtingu sprendimo parinkimu iš dviejų komponentų.

2.6. Papildomi matematiniai skaičiavimai

Siekiant įdiegti modifikaciją į tinklinę sistemą susiduriame su rimta problema. Labai sunku tokiose sistemose išmatuoti laiką, kuris buvo praleistas mokantis. Tokia galimybė yra, tačiau „Moodle“ sistemoje ši technologija naudojama tik atliekant kontrolinius testus. O tai nelabai tikslinga naudoti resursų skaitymo procese. Todėl buvo panaudoti tam tikri matematiniai skaičiavimai, leidžiantys apeiti laiko intervalų naudojimą. Bus remiamasi resurso paspaudimais per dieną ir tų paspaudimų dažnumu. Kad išvengtume priklausomybės nuo konkretaus paspaudimų skaičiaus, panaudosime eksponentinį dėsnį ir rezultatus vienai dienai skaičiuosime pagal formulę (1):

$$y = 1 - e^{-x/k} \quad (1)$$

Formulėje koeficientas k nusako paspaudimų skaičiaus įtaką rezultatui, kuo didesnis koeficientas, tuo daugiau paspaudimų reikės, norint pasiekti geresnį rezultatą. Formulės privalumas yra tas, kad vienu paspaudimu jau galima pasiekti iki 63% medžiagos mokymosi įvertinimo, tačiau kuo daugiau kartų spaudžiama nuoroda, tuo mažiau procentų didėja (pvz.: 10 paspaudimų bus 95%). Šitaip išvengiama galimo sukčiavimo (nes vien spaudinėjimai dar neduoda labai didelių rezultatų) ir normalizuojami rezultatai.

Norėdami atrasti „sekimo“ ir testavimo duomenų kritinius taškus taip pat naudosime tam tikrus skaičiavimus. Pirmiausia, kritiniai taškai susidės iš skirtumo tarp testavimo ir „sekimo“ duomenų. Nenormalus t.y. kritinis taškas bus suprantamas kaip toks, kuris neatitinka daugumos taškų dėsningumo. Tačiau nenaudosime sudėtingų skaičiavimų, nes labai svarbu neturėti laiko sąnaudų, be to, nėra tikslinga naudoti sudėtingus skaičiavimus. Vadinasi, mums labiausiai tinka standartinio nuokrypio arba dispersijos skaičiavimai. Standartinis nuokrypis bus patikimesnis, kadangi duomenys normalizuoti ir išreiškiami procentais, skaičiuosime pagal formules (2):

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i \quad \sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2} \quad (2)$$

Kritiniai taškai mums leis aptikti tokius dalykus kaip neefektyvią mokomąją medžiagą arba blogai susikurtą mokymosi planą.

2.7. Skyriaus apibendrinimas

Skyriuje yra trumpai aptariamos naudojamos technologijos, jų galimybės ir ribojimai būsimai modifikacijai. Adaptyvumo samprata yra apibrėžiama dvejopai [16,17]:

- Besimokantysis pasinaudodamas vartotojo sąsaja registruoja savo pasirinkimus, pagal kuriuos sistema ir prisitaiko;
- Sistema prisitaiko prie besimokančiojo iš modelio pasikeitimų gautų duomenų pagalba.

Trumpai yra aptariamos agentinės technologijos bei agentiniai sistemos bruožai, nes tai glaudžiai susiję su adaptyviomis sistemomis.

Darbe buvo atliktas eksperimentas, skirtas sprendimų rengimo sistemų skaičiavimų panaudojimo galimybei patikrinti. Eksperimento metu nustatyta, kad tokie skaičiavimai galėtų būti naudojami modifikacijoje, tačiau pateikiamos ir papildomos išvados:

- Realizacijai bus panaudota tik nedidelė dalis AHP komponentų;
- Sprendimai priimami tik dviem aspektais, todėl skaičiavimai pasidaro gana paprasti;
- Skirti prioritetus kriterijams kol kas yra gana sudėtinga, tokius prioritetus galima nustatyti tik su specialisto pagalba arba atlikus bandymus su realiais respondentais.

Modifikacijoje yra panaudoti tam tikri matematiniai skaičiavimai:

- Tam, kad išvengtume mokymo(si) intensyvumo priklausomybės nuo konkretaus paspaudimų skaičiaus, panaudosime eksponentinį dėsnį ir vienos dienos rezultatus skaičiuosime pagal eksponentinę formulę (1);
- Nenormaliems kritiniams taškams aptikti pasirinkta naudoti standartinį nuokrypį (2). Kritiniai taškai leis aptikti tokius dalykus kaip neefektyvią mokomąją medžiagą arba blogai susikurtą mokymo(si) planą.

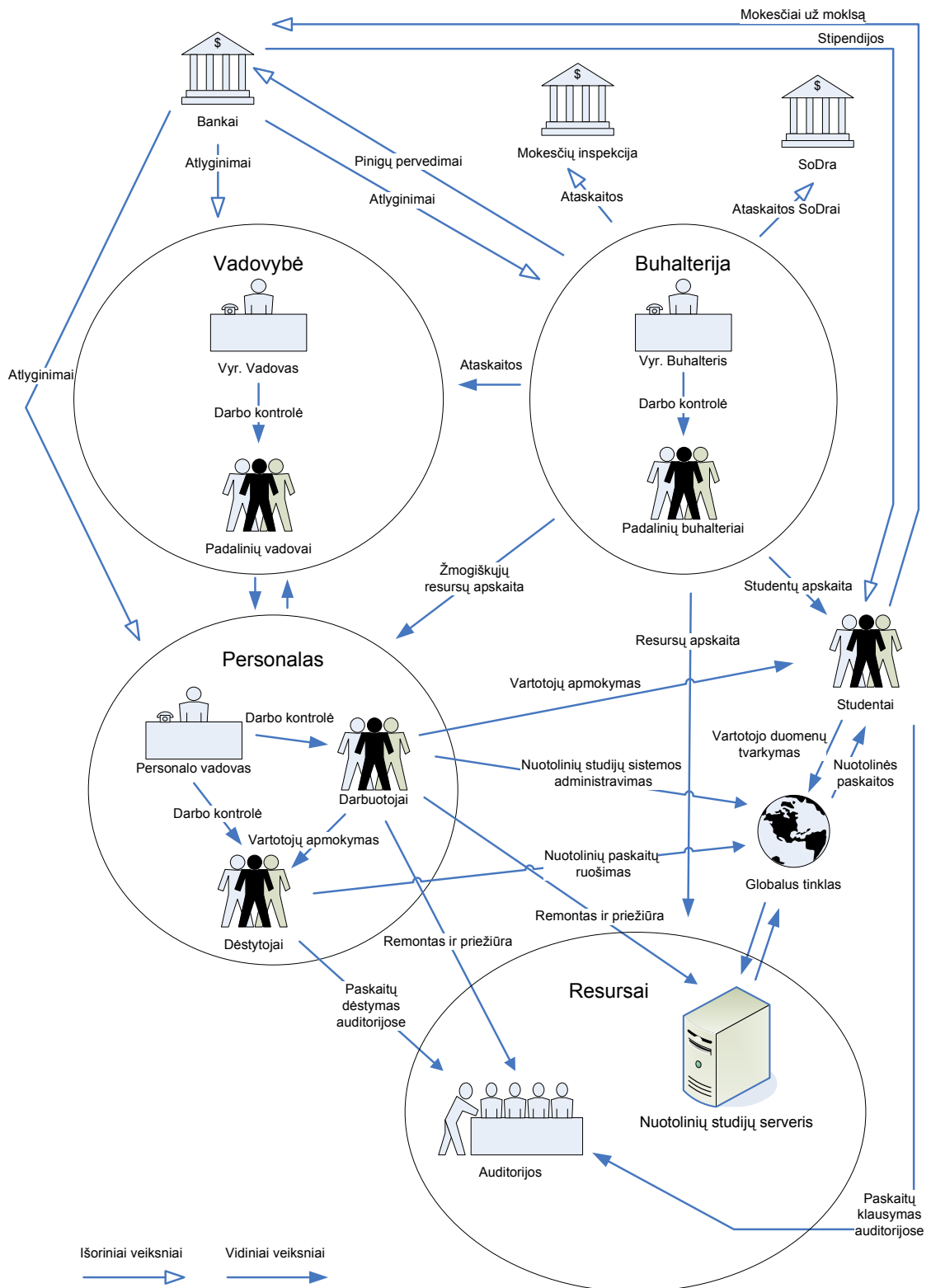
3. Nuotolinių studijų posistemės vizija, koncepcinis modelis

Siekdami realizuoti nuotolinių studijų modifikaciją, ištirsime sistemą, jos aplinką, vartotojus bei suformuluosime savarankišką mokymosi procesą. Sistemos probleminėms sritims atvaizduoti įvairiais aspektais panaudosime UML diagramas [23,47].

3.1. Nuotolinių studijų sistemos vartotojų identifikavimas

Organizacijos „raiškusis paveikslėlis“ skirtas grafiškai atvaizduoti organizacijoje vykstančius informacijos ir daiktų srautus bei aktorius ir jų poreikius. „Raiškusis paveikslėlis“ atspindi bendrą organizacijos vaizdą ir jame turi būti visi aktoriai ir visi srautai, nepriklausomai nuo tiriamos dalies [23]. Šiuo atveju tiriamos organizacijos „raiškusis paveikslėlis“ pravers lengviau išvaizduoti tiriamos dalies vietą sistemoje ir jos aplinkoje bei identifikuoti sistemos vartotojus. Kadangi rengiamai adaptyviamai posistemei ir besimokančiojo modeliui svarbiausia yra naudojama sistema ir jos vartotojai, kitos organizacijos dalys toliau nebus nagrinėjamos. Taip daroma todėl, kad lyginant su tiriamą dalimi, visa dalis yra labai didelė ir nėra prasmės nagrinėti visos sistemos ir jos aplinkos. Praktiškai nagrinėti tenka ir nuotolinių studijų programų sistemos dalį, todėl būtų gana sudėtinga atsižvelgti ir į beveik jokios tiesioginės įtakos nedarančius organizacinius vienetus.

„Raiškusis paveikslėlis“ taip pat yra šiek tiek supaprastintas, kad būtų galima aiškiau išskirti tiriamąją dalį (18 pav.).



18 pav. Nuotolinės sistemos ir jos organizacinės aplinkos "Raiškusis paveikslėlis"

Toliau aprašysime mus dominančius vartotojus ir resursus, kurie tiesiogiai susiję su nuotolinių studijų sistema.

Pagrindiniai tiriamosios organizacijos dalies, kurią nuo šiol vadinsime nuotolinėmis studijomis, uždaviniai yra šie:

- 1) suteikti galimybę studentams gauti mokomąją medžiagą ir žinias naudojantis nuotolinių studijų sistema;
- 2) įvertinti studentų įgytas žinias taip pat naudojantis šia sistema.

Pagrindiniai vartotojai yra šie: studentai, dėstytojai, darbuotojai (nuotolinių studijų sistemos administratoriai); bei jų naudojami resursai: nuotolinių studijų serveris, globalus tinklas, nuotolinių studijų sistema ir jos duomenų bazė.

Žemiau pateikiamas vartotojų ir resursų aprašas padėsiantis suvokti kiekvieno iš jų paskirtį nuotolinėse studijose.

Dėstytojai

Dėstytojų pagrindinė funkcija nuotolinėse studijose yra paruošti kokybišką mokymosi medžiagą ir ją patalpinti nuotolinių studijų sistemos duomenų bazėje, kuri būtų prieinama studentams. Taip pat formuoti užduotis ir testus, skirtus studentų savarankiškam darbui bei jų egzaminavimui, kontroliuoti mokymosi procesą, skatinti, motyvuoti ir kt.

Studentai

Studentai naudojami mokomąją medžiagą, papildomais įrankiais ir veiklomis, kurie yra sistemoje. Taip pat atlieka testus ir užduotis, kurie reikalingi žinių įvertinimui.

Darbuotojai (Administratoriai)

Prižiūri nuotolinių studijų sistemą, bei nuotolinių studijų serverį, taip pat rūpinasi, kad sistema būtų prieinama visiems studentams globaliajame tinkle. Administratoriai nustato sistemos vartotojus ir suteikia jiems privilegijas arba apribojimus. Be to, administratoriai konsultuoja dėstytojus, kaip naudotis nuotolinių studijų sistema.

Nuotolinių studijų serveris:

Fizinė kompiuterinė sistema, skirta sistemai ir jos duomenų bei žinių bazei laikyti.

Nuotolinių studijų sistema

Nuotolinių studijų sistemos pagrindinė funkcija užtikrinti darnią ir funkcionalią nuotolinių studijų virtualią aplinką. Nuotolinių studijų sistema turi teikti dėstytojams, studentams ir

darbuotojams įrankius reikalingus jų užduotims atlikti. Bei užtikrinti duomenų saugojimą ir saugumą.

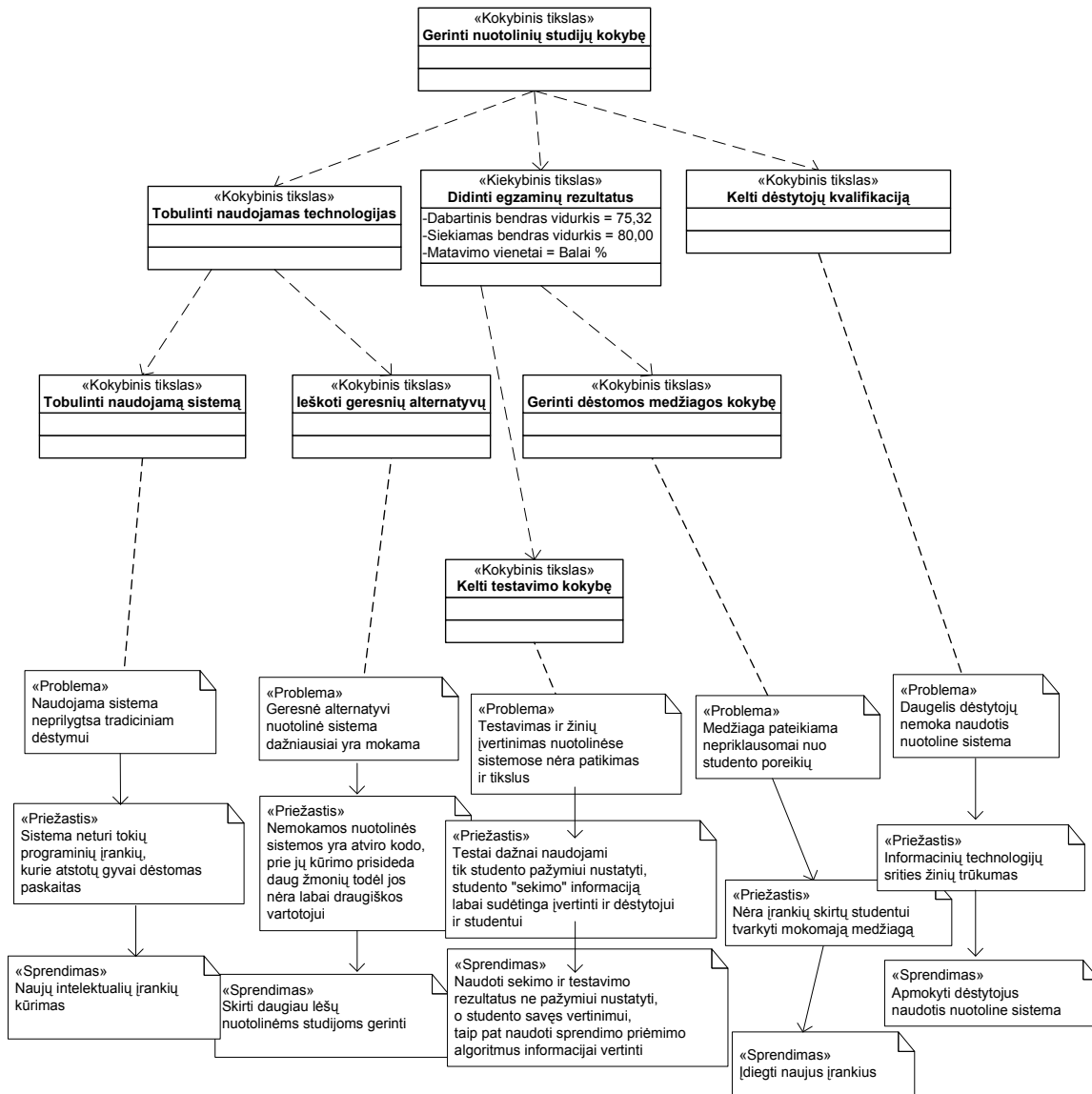
Globalus tinklas

Globalus tinklas taip pat svarbus nuotolinėse studijose. Nors jis kaip resursas atrodo savaime suprantamas dalykas, tačiau prisiminus, kad tai vienintelė terpė, kurios pagalba sujungiami resursai ir vartotojai, pasidaro labai svarbu, kad būtų užtikrinta prieiga prie globalaus tinklo.

Problemų identifikavimui reikalingas organizacijos vaizdas gaunamas keliais aspektais. Tai galėtų būti tikslų, procesų, organizacinis, resursų ir kiti aspektai. Aspektų pasirinkimas priklauso nuo pačios nagrinėjamos organizacijos ir konkrečios modifikuojamos jos dalies. Visi aspektai bus nagrinėjami pasitelkiant modelius. Pirmiausia tai – tikslų aspektas. Tikslų aspektas padės pamatyti bendrą vaizdą spręstinių problemų, kurios atsiranda organizacijai siekiant tikslų, pvz., kaip gerinti studijų kokybę ar kelti dėstytojų kvalifikaciją.

3.2. Tikslų modelis

Šiame modelyje vaizduojami tikslai, kurių siekia organizacija [23]. Tikslai vaizduojami medžio forma, aukščiausiame medžio lygmenyje yra pats bendriausias įmonės tikslas apimantis daug iš jo išplaukiančių konkretesnių tikslų. Tikslų medyje be tikslų gali būti vaizduojamos ir sprendžiamos problemos. Tikslai daro tiesioginę įtaką organizacijos elgsenai (19 pav.).



19 pav. Nuotolinių studijų tikslų medis

Tikslų medžio (19 pav.) keliamos problemos yra daugiau informacinės sistemos įgyvendinimo klausimai. Tačiau tai labai svarbu, norint suformuluoti pačios posistemės integraciją į nuotolinių studijų sistemą, taip pat nustatyti posistemės poreikį nuotolinėse studijose. Tikslų modelio pagalba nustatytos šios problemos:

- Nuotolinių studijų sistemos vis dar neprilygsta tradicinėms studijoms;
- Besimokančiojo egzaminavimas ir žinių įvertinimas nuotolinių sistemų pagalba nėra patikimas;
- Mokymosi medžiaga pateikiama nepriklausomai nuo besimokančiojo poreikių;
- Besimokančiajam sunku įvertinti pačiam savo mokymosi procesą;
- Dėstytojai turi mažai informacijos, kaip veiksmingai ir kokybiškai pateikti mokymosi medžiagą ir kitas veiklas, studentui naudojantis nuotolinių studijų sistema;

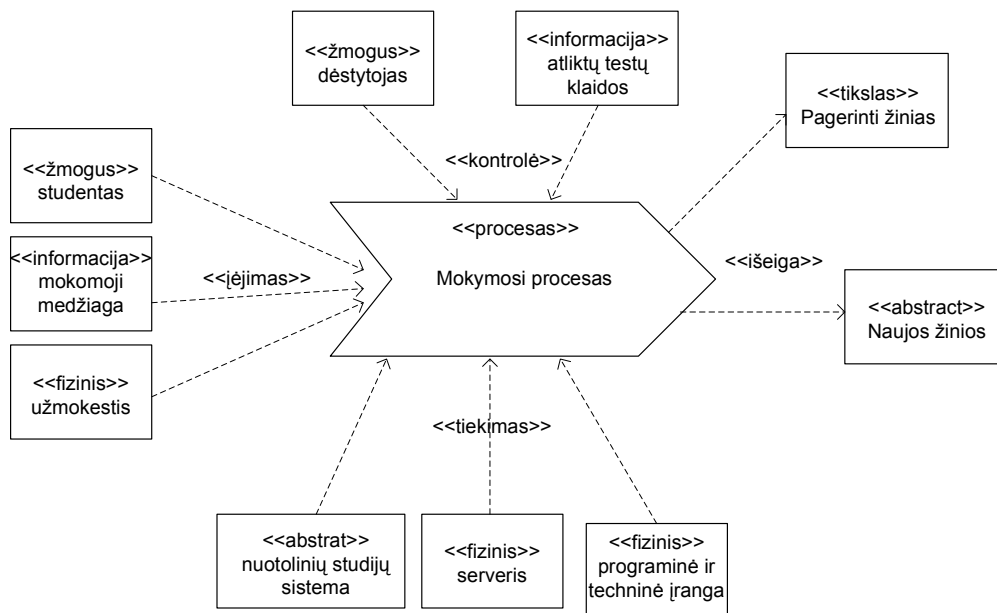
Formuluojant šias problemas galima padaryti išvadą, kad nuotolinių studijų sistemai reikalingi nauji sudėtingesni įrankiai, kurie galėtų atstoti tradicines paskaitas. Taip pat reikia atsižvelgti ir į tai, kad negalima kopijuoti tradicinėse paskaitose taikomų metodų, kadangi nuotolinės studijos labai skiriasi nuo tradicinių dėl atstumo tarp studentų ir dėstytojų. Išskyla pagrindinė problema: kaip motyvuoti studentą mokytis pačiam be dėstytojo? Kaip pateikti medžiagą, kad ji būtų pakankamai aiški ir dinamiška? Iš čia kyla ir problemų sprendimai. Reikalingi papildomi įrankiai, reikalingas specialus mokomosios medžiagos apdorojimas, suskirstymas ją talpinant. Taip pat reikia atkreipti dėmesį į tai, kad kuriant naujus medžiagos talpinimo įrankius yra apkraunamas dėstytojas – į tai būtina atsižvelgti ir nekurti sudėtingų sistemų. Todėl sistemos intelektualizavimas yra vienas iš realių sistemos patobulinimo būdų.

Toliau panagrinėsime nuotolinių studijų mokymosi procesą.

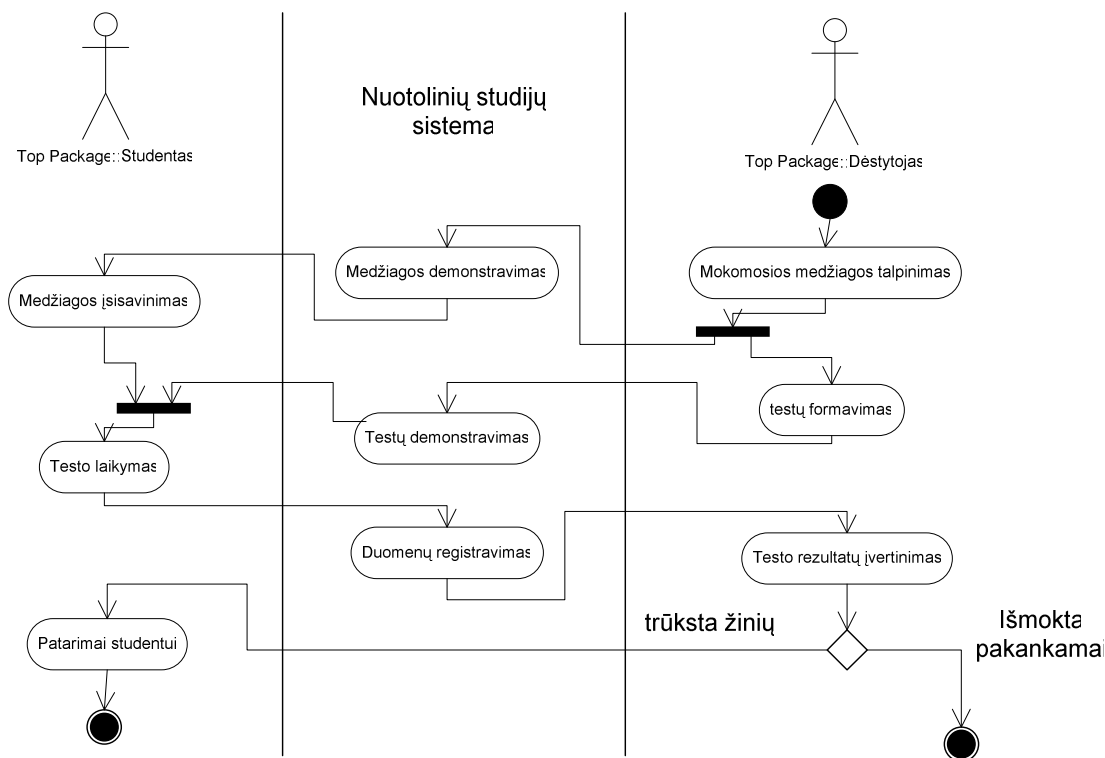
3.3. Procesų modelis

Proceso modeliavimas susideda iš resursų tiekimo ir proceso tikslo, įeinančių resursų ir išeinančio rezultato. Procesas turi turėti motyvaciją, tam yra vaizduojamas proceso tikslas, nusakantis kodėl šis procesas vyksta. Procesas taip pat turi įėjimą ir išėjimą. Įėjimą sudaro resursai (žaliavos ar kažkokie kiti pradiniai komponentai), kurie bus proceso eigoje transformuojami į naują resursą, t.y. bus pagamintas produktas – proceso išeiga [47]. Tačiau vien tik įeinančio resurso nepakanka tam, kad įvykdytume transformaciją, proceso vykdymo metu reikia tiekti papildomus resursus, tai – žmonių žinios, gamybos informacija, proceso kontrolė ir t.t. Tai yra papildomi resursai, kurie reikalingi proceso įvykdimui.

Mokymo(si) procesas nuotolinių būdu smarkiai skiriasi nuo įprasto mokymo(si) proceso. Čia daugiausiai sprendimų turi daryti pats besimokantysis. Pasirinkti kokiais resursais naudotis, kokias užduotis atlikti, kiek laiko skirti mokymuisi, kaip įvertinti jau įgytas žinias ir t.t. Pamėginsime išskirti pagrindines problemas, su kuriomis susiduria besimokantysis mokymo(si) procese.



20 pav. Mokymo(si) proceso modelis nuotolinių studijų sistemoje



21 pav. Detalizuotas mokymo(si) proceso modelis

Žvelgdami į mokymosi procesą vaizduojančias diagramas (20, 21 pav.), galime pastebėti vietas, kuriose besimokančiajam reikia priimti sprendimus. Pirmiausia tai – proceso pradžioje. Studentas turi įvertinti mokymosi medžiagą, pasirinkti tai kas yra svarbiausia jam ir suplanuoti mokymosi laiką. Tačiau pradžioje jis neturi jokių papildomų priemonių tai atlikti, išskyrus galimus dėstytojo komentarus. Besimokančiojo žinių pradiniam įvertinimui, literatūroje dažniausiai sutinkamas sprendimas yra pradinių žinių nustatymo testo laikymas. Tai galėtų būti tinkamas problemos sprendimas.

Tolimesnės problemos iškyla, kai besimokantysis jau yra mokymo(si) proceso eigoje. Jam reikia įvertinti tai, ką jis išmoko ir numatyti, kur reikia papildomai gilinti žinias. Vėlgi, įrankių padedančių tai padaryti nėra. Čia labai praverstų SRS metodai. Beje, iš modelio matome, kad ši problema iškeliama tik pasibaigus mokymo(si) procesui. Čia susiduriame su įprasto mokymo problema – testai dažniausiai sudaromi tam, kad dėstytojas galėtų įvertinti studento žinias, o ne tam, kad studentas galėtų pats save įvertinti. Todėl dėstytojui tenka svarbus uždavinys – suformuluoti medžiagą taip, kad ji būtų išskaidyta į atskiras smulkias dalis (skyrus ir temas), ir kad kiekviena dalis turėtų savęs testavimo galimybes, papildomą literatūrą, papildomas užduotis ir kita, kas neįeitų į studento pažymio nustatymą. Studento užduotis, skirtas pažymiui įvertinimo, reikia griežtai atskirti

nuo mokomųjų. Įrankiai, sudėtingai mokymosi medžiagai formuoti, nuotolinėje sistemoje jau yra sukurti ankstesniuose studijose.

Toliau pabandydysime tiksliau apibrėžti kelias problemas:

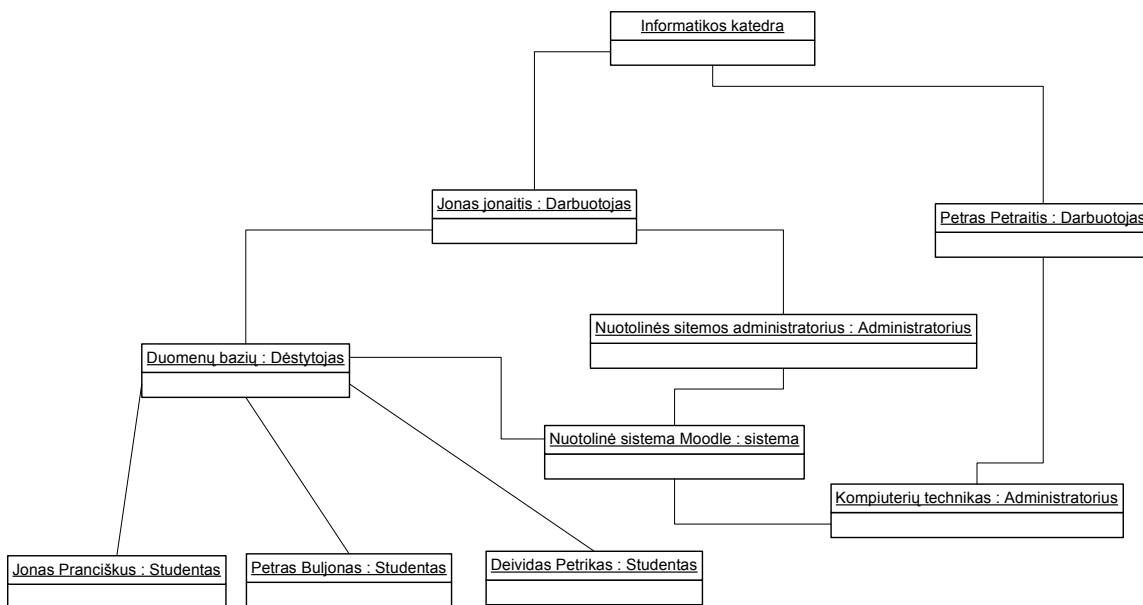
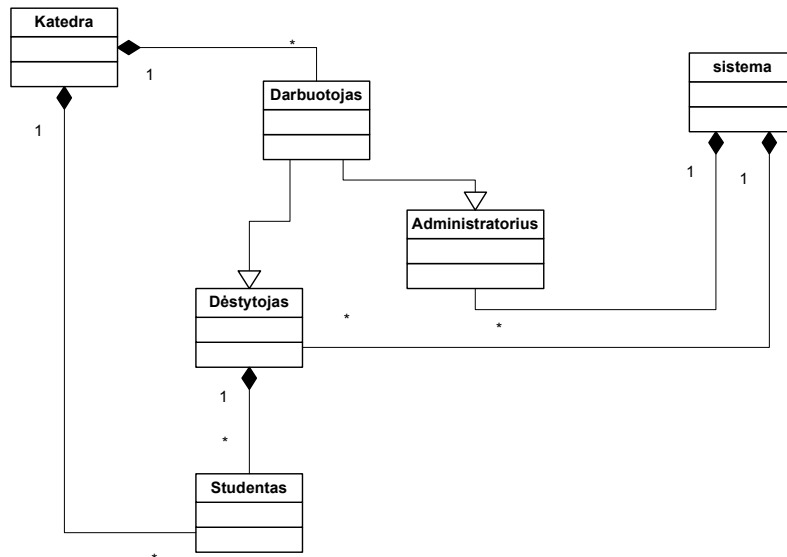
- Kokiai mokomosios medžiagos daliai skirti daugiau mokymo(si) laiko?
- Kokio tipo medžiagą pasirinkti mokytis?
- Ar bandyti mokytis visą medžiagą ar jau yra įgytų žinių anksčiau?
- Ar skaityti papildomą literatūrą?
- Ar atlikti papildomas užduotis?
- Kurioje mokymosi proceso dalyje pasirinktas neteisingas sprendimas?
- Kaip įvertinti mokymo(si) metu įgytas žinias?
- Kuri mokymosi medžiagos dalis per mažai įsigilinta mokymosi proceso metu?
- Kodėl mokymosi medžiaga buvo nepakankamai išmokta?

Suformulavę problemas, aiškiai matome, kad priimti sprendimus tikrai gali padėti besimokančiojo modelio vaizdavimas. Tai bus aptariama tolimesniuose skyriuose. Taip pat, projektuojant modifikaciją, reikia atsižvelgti ir į tai, kad turi būti lengvai suprantami patarimai besimokančiajam, o ne sudėtinga informacija dėstytojui.

Toliau panagrinėsime organizacinį modelį.

3.4. Organizacinis modelis

Organizacinės struktūros supratimas yra svarbus ne tik dėl pertvarkymo tikslų, bet ir dėl to, kad išsiaiškintume kiekvieno organizacinio vieneto atsakomybę. Organizacijos yra struktūrizuotos tam, kad būtų geriau valdyti įmonę. Organizacijos padalinimas į mažesnius vienetus bei atsakomybės už tam tikrą veiklą paskirstymas atskiriems padaliniams, įgalina patogiau valdyti organizacijos produkciją.



22 pav. Nuotolinių studijų organizacinis modelis

Organizacinė struktūra yra vaizduojama klasių ir objektų diagramomis [47]. Turint galvoje, kad organizacijos struktūra gali keistis, buvo sumodeliuota apibendrinanti klasių diagrama.

Organizaciniame modelyje (22 pav.) klasių ir objektų diagramoje nevaizduojama visa universiteto struktūra, nes sprendžiamos problemos yra žemesniame sisteminiame lygmenyje. Klasių diagrama vaizduoja bendrą objektų sąveikos modelį, o objektų diagrama – konkretų organizacinės struktūros pavyzdį.

Organizacinės struktūros modelio pagalba galime rasti keletą esminių problemų, tiesiogiai nesusijusių su besimokančiojo modelio projektavimu. Iš organizacinės struktūros galime aiškiai matyti, kad nuotolinės sistemos administravimu rūpinasi tik dėstytojai ir administratoriai, tuo tarpu studentai gali tik naudotis sukonfigūruota sistema. Iškyla pagrindinė problema – kaip studentui pritaikyti sistemą savo asmeniniams poreikiams. Ryšys tarp studento ir sistemos supaprastintai nėra vaizduojamas, kadangi studentai neturi jokios sistemos valdymui.

Tai dar kartą parodo reikalingumą – įdiegti papildomus įrankius. Pagrindinės problemos keliamos organizacinio modelio:

- Atgalinio ryšio tarp sistemos ir studento nebuvimas;
- Kaip galima būtų pritaikyti virtualią mokymo(si) aplinką kiekvienam studentui?
- Per mažai įrankių, kuriais galima būtų studentui konfigūruoti ne tik savo duomenis sistemoje, bet ir pačią mokymosi aplinką.

Tai yra gana svarbios problemos ne tik projektuojamai sistemai, bet ir nusistovėjusiai organizacinei struktūrai. Reikia atsižvelgti ir į tai, kad studentas nuotolinėse studijose tampa labiau savarankišku organizacinės struktūros vienetu.

Toliau panagrinėsime sistemą resursų aspektu, kadangi tai gali būti svarbu besimokančiojo modelio įgyvendinimui adaptyvioje posistemėje.

3.5. Resursų aspektas

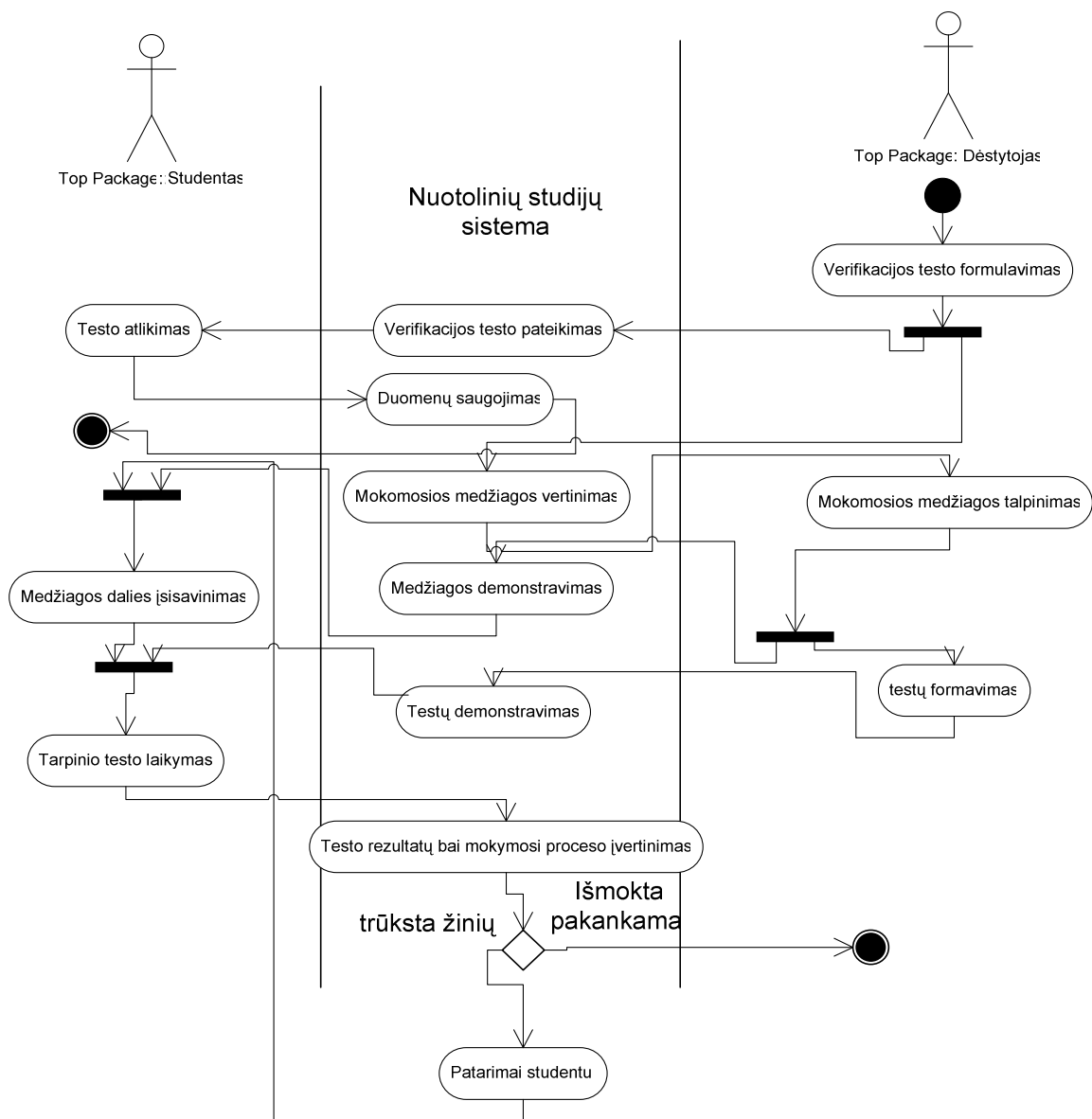
Kadangi resursų struktūra nėra sudėtinga, galime remtis „raiškioju paveikslėliu“. Pirmiausia atkreipsime dėmesį į žmogiškuosius išteklius ir jų žinias. Studentai, dėstytojai ir administratoriai bei jų žinios yra pagrindiniai žmogiškieji ištekliai nuotolinėse studijose. Nauji resursai – tokie, kaip mokymosi proceso produktas, bus naujos studentų žinios ir sugebėjimai. Fiziniai resursai – tai nuotolinių studijų sistema, serveris ir duomenų bazė, naudojama nuotolinių studijų sistemos. Tam, kad būtų galima integruoti besimokančiojo modelį, galima pasinaudoti jau esama duomenų baze. Taigi pagrindinė resursų aspekto keliami problema galėtų būti – kokią informaciją iš duomenų bazės naudoti.

Taigi pamėginsime išvardinti galimas problemas, kurias kelia resursų struktūra:

- Iš kur gauti informaciją besimokančiojo modeliui atvaizduoti?
- Kokią informaciją naudoti?

3.6. Galimas nuotolinių studijų modifikacijos veikimo procesas

Didelė dalis sprendimų priėmimo perleidžiama sistemai. Tačiau nereikia suprasti, kad pati sistema priima sprendimus. Čia norima pasakyti, kad sistema apdoroja turimą informaciją ir formuluoja konkrečius patarimus, pagal kuriuos besimokantysis pats nusprendžia kokią sprendimą jam priimti. Žemiau pateikiamas galimas naujas mokymosi procesas (23 pav.). Tačiau jame dabar demonstruojamas tik vienos medžiagos dalies mokymas(is), visas mokymo(si) procesas turėtų tiek iteracijų, kiek būtų mokomosios medžiagos dalių, neįtraukiant tik verifikacijos.



23 pav. Sprendimų priėmimo procesas

3.7. Skyriaus apibendrinimas

Skyriuje nustatyti sistemos vartotojai ir apibrėžta sistemos aplinka:

- Dėstytojai;
- Studentai;
- Darbuotojai;
- Nuotolinių studijų serveris;
- Nuotolinių studijų sistema;
- Globalus tinklas.

Sistemos vizija suformuluota panaudojant kelių aspektų sistemos modelius:

- Tikslų modelis formuluoja išvadą, kad nuotolinių studijų sistemai reikalingi nauji sudėtingesni įrankiai, kurie galėtų atstoti tradicines paskaitas.
- Procesų modelyje galime išskirti tam tikras vietas, kuriose besimokančiajam reikia priimti sprendimus. Pirmiausia studentas turi įvertinti mokymosi medžiagą, pasirinkti tai – kas svarbiausia jam – ir suplanuoti mokymosi laiką, tačiau pradžioje jis neturi tam reikalingų priemonių. Jam taip pat reikia įvertinti tai, ką jis išmoko, ir numatyti, kur reikia papildomai gilinti žinias. Galima konstatuoti, kad testai dažniausiai sudaromi tam, kad dėstytojas galėtų įvertinti studento žinias, o ne tam, kad studentas galėtų pats save įvertinti.
- Organizacinis modelis padeda suprasti tai – kad studentas nuotolinėse studijose tampa labiau savarankišku organizacinės struktūros vienetu.

Skyriaus pabaigoje yra suformuluojamas nuotolinių studijų modifikacijos veikimo procesas, turima galvoje, kad sistema apdoroja informaciją ir formuluoja konkrečius patarimus, pagal kuriuos besimokantysis pats nusprendžia kokį sprendimą jam priimti.

4. Besimokančiojo modelio adaptyvioje VMA posistemėje įdiegimas

Sistemos modifikavimui buvo suprojektuotos diagramos, kurios nusako, kokie nauji objektai bus įjungti į sistemą. Kadangi sistema yra tinklinė, faktiškai galima atskirti dvi sistemos dalis: **dėstytojo (administratoriaus)** ir **vartotojo** sąsajos. Čia kalbama apie sistemos ribojimąsi su vartotojais. Administravimo atžvilgiu turime sistemos sąsają su dėstytoju arba administratoriumi, kur reikia suformuoti resursus ir pasirinkti nustatymus, bei turime studento (vartotojo) sąsają, kuri skirta studentui naudotis resursais. Šiose sistemos dalyse išsiskiria ir daugiau aspektų, tačiau mus domina kol kas tik šie du.

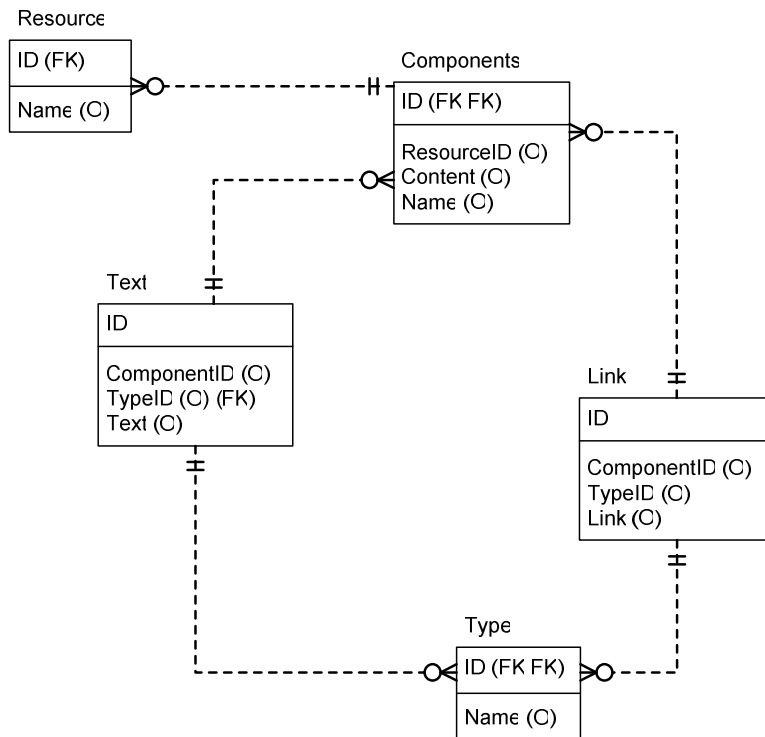
Taip pat atskirsime sąsajos su vartotoju **vaizdavimą** ir **loginę** dalis.

Sąsajos išvaizda sistemoje jau turi nusistovėjusius standartus, modifikuoti reikės tik atskiras, specifines vaizdavimo dalis, tokias kaip – grafikų rodymas. Grafikų išvedimui bus panaudoti specialūs metodai. Šiuo metu tai nėra labai svarbu, nes ši darbo dalis skirta suformuoti bendrą komponentų struktūrą, nepriklausomą nuo pasirinktų realizacijos įrankių.

Taigi šiame skyriuje aptarsime loginę sistemos dalį, kodo realizavimo galimybes bei sistemos modifikavimo konkrečias vietas.

4.1. Duomenų bazės modifikavimas

Duomenų bazei reikalingas naujas komponentas. Kadangi esamoje sistemos duomenų bazėje nėra sudėtinio resurso modelio, buvo suformuluotas naujas duomenų bazės komponentas. Reikia dar paminėti, kad sudėtinio resurso įvedimas ir demonstravimas jau yra realizuotas, tačiau žvelgiant iš „Moodle“ sistemos naudojamų resursų struktūros pusės, būtų patogiau, jei visi resursai būtų saugomi, naudojant tas pačias technologijas. Be to, siekiant modifikaciją padaryti lanksčiau ir pritaikomą kitoms sistemoms, reikėtų atsižvelgti į įvairias realizacijos galimybes.



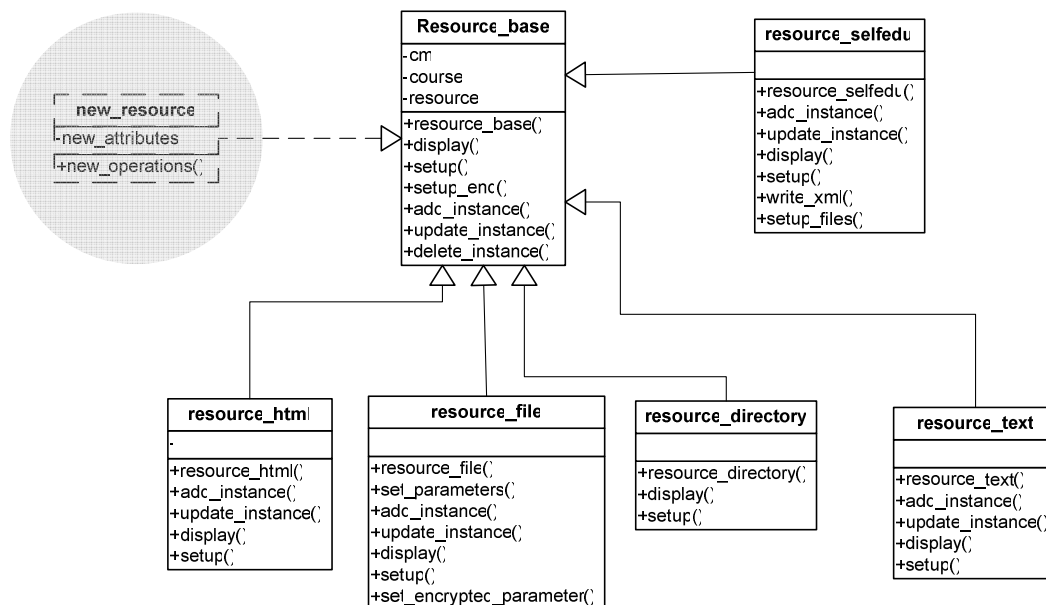
24 pav. Esybių ryšių duomenų bazės fragmento diagrama

Esybių ryšių diagramoje, pavaizduotame duomenų bazės fragmente (24 pav.), resursas turi komponentų sąrašą, kuris gali susidėti iš teksto arba iš nuorodos. Ši diagrama parodo tik dalį duomenų bazės, tą kuri reikalinga naujo tipo resursui saugoti. Likusi duomenų bazės dalis lieka nepakitusi. Kadangi resurso tipai praktiškai skiriasi tik tuo, ar komponentas yra tekstas ar objektas, duomenys buvo suskirstyti į dvi kategorijas: tekstas saugomas duomenų bazėje, o objektai saugomi failų sistemoje. Duomenų bazėje apie objektą saugomas tik jo adresas – nuoroda į failą.

Failų įkėlimą galima organizuoti, sinchronizuojant su duomenų bazės atnaujinimu. Failų saugojimas duomenų bazėje nėra lankstus būdas, be to reikalauja daugiau sistemos resursų.

4.2. Dėstytojo sąsajos loginės dalies modifikavimas

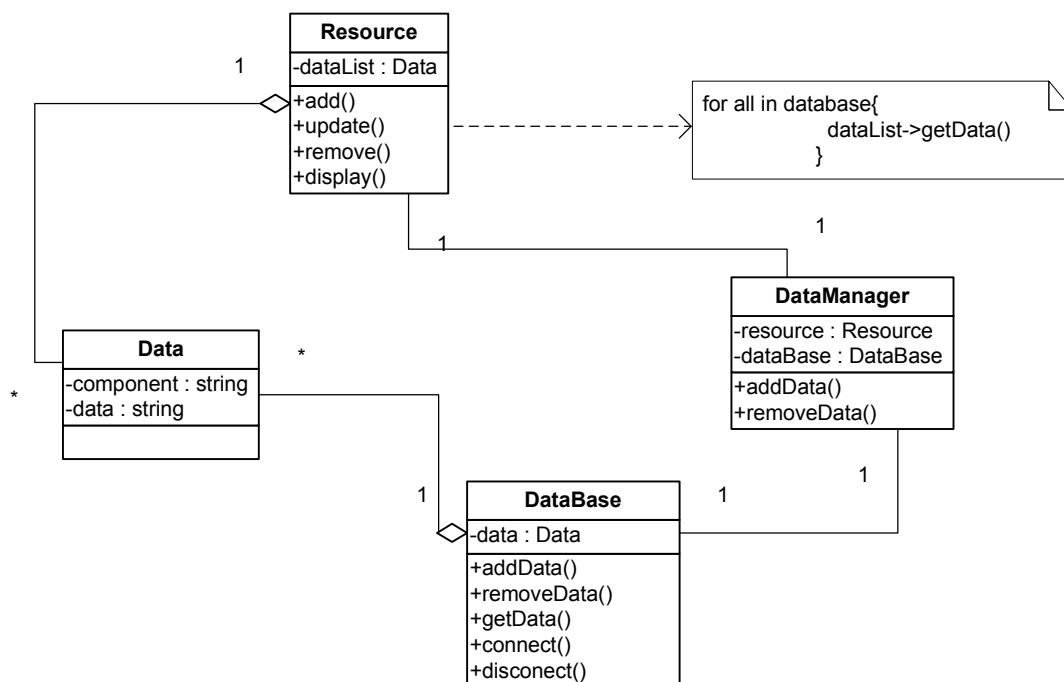
Dėstytojo (administratoriaus) sąsaja yra skirta įkelti arba modifikuoti mokomąją medžiagą, naują resursą. Pakankamai lanksti resurso redagavimo vartotojo sąsaja jau yra sukurta ankstesnėse studijose, todėl vartotojo sąsaja nebus modifikuojama. Modifikuojami tik duomenų saugojimo algoritmai. Pačioje sistemoje resurso modifikavimo mechanizmas yra lankstus – naujas komponentas yra nesunkiai pridedamas (25 pav.).



25 pav. VMA resursų klasių struktūra [3]

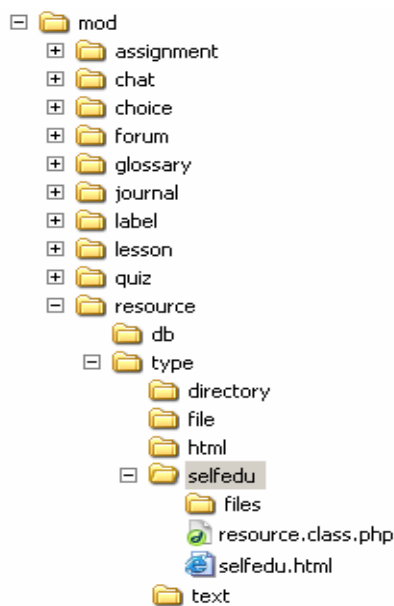
Tačiau esamų operacijų resursui saugoti, vaizduoti ir redaguoti nepakanka, kadangi klasės suprojektuotos vientisam resursui [3]. Tam, kad būtų galima įkelti naujo tipo resursą, buvo suprojektuota nauja klasių struktūra, kuri bus realizuota, kaip papildoma sistemos dalis.

Taip pat išskirtos klasės operavimui su duomenų baze ir su failų sistema. VMA sistemoje, operavimui su duomenimis, yra parašytos specialios procedūros/funkcijos, kurias galima naudoti globaliai. Tačiau mes pabandydysime duomenų saugojimo procedūras apjungti į objektus ir klases, siekiant lankstesnio panaudojimo ir specifinio pritaikymo (26 pav.).



26 pav. Resurso modifikacijos klasių struktūra

Skirtingų resursų struktūros kodas yra saugomas panaudojant failų struktūrą. VMA yra nusistovėjusi failų struktūra, kurios reikėtų ir laikytis. Visi vartotojo sąsajos ir kodo failai turi būti saugomi tam skirtame kataloge, išlaikant klasių failo pavadinimo struktūrą. Papildomi failai jokių apribojimų nėra.



27 pav. Failų sistema

Visi papildomi failai ir papildomos formos bus saugomi atskirame kataloge (27 pav.). Reikia pabrėžti, kad šioje vietoje, mes turime susieti ir šaltinio atvaizdavimą. Kadangi šaltinio klasė („Resource_class“) turi procedūrą, skirtą resurso atvaizdavimui, tai ši procedūra turi būti susijusi su specialiu šaltinio vaizdavimu, kuris aprašytas sekančiame skyriuje.

4.3. Vartotojo sąsajos loginės dalies modifikavimas

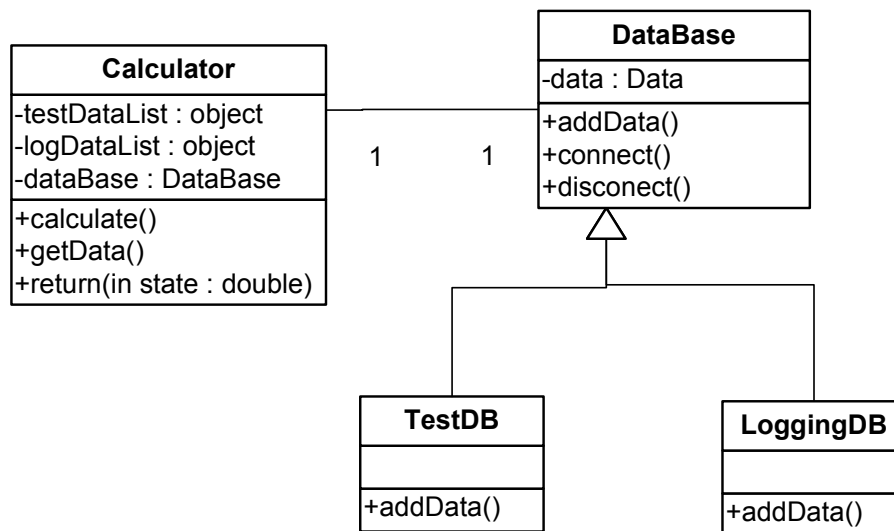
Pradėsime nuo resurso perteikimo ir tam reikalingų komponentų. Resurso vaizdavimui jau yra sukurti specifiniai komponentai – resursas yra išvedamas į ekraną temų ir potemių struktūra. Tačiau naujas resursų išvedimo funkcionalumas turėtų būti atskirų komponentų mokymosi intensyvumo ir testavimo rezultatų įvertinimo galimybė. Mokymosi intensyvumas bus vaizduojamas raudonos ir žalios spalvų perėjimu bei procentinės išraiškos apjungimu. Spalvos bus septynios ir dalins intensyvumą į septynis intervalus (28 pav.).



28 pav. Spalvų perėjimas

Šiame skyriuje aptarsime kaip duomenys ir operavimo jais logika turėtų būti realizuota klasių struktūromis.

Diagrama (29 pav.) vaizduoja komponentą, kuris turėtų papildyti resurso vaizdavimo procedūrą. Resursas – „resource_class“ su klase – „Calculator“ turėtų būti susietas asociacijos ryšiu, kuris leis naudoti tokią struktūrą išvedimo duomenų skaičiavimui.



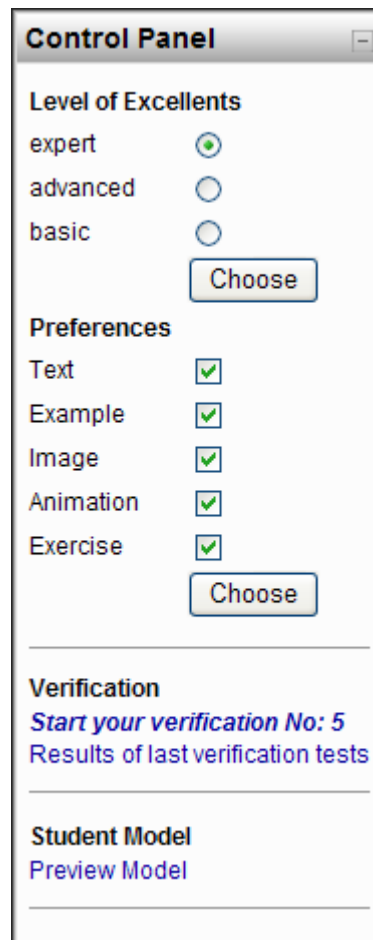
29 pav. Resurso vaizdavimo komponento klasių struktūra

Kadangi intensyvumo skaičiavimams naudojami sistemos vartotojų istorijos įrašai, bei vartotojų testų rezultatų įrašai, duomenų skaitymas iš duomenų bazės turi būti realizuotas skirtingai. Šių dviejų struktūrų atskyrimas turi būti realizuotas klasių paveldėjimu. Skaičiavimo klasė yra skirta

apdoroti duomenims, gautiems iš duomenų bazės valdymo klasės. Pati klasė grąžina apdorotus duomenis resurso vaizdavimo procedūrai, kur jie bus transformuojami į grafinius komponentus.

Dar vienas komponentas, kuris turi įeiti į studento modelį, yra studento sekimo ir testavimo duomenų grafinis vaizdavimas. Pats komponentas bus realizuojamas kaip naujas paskaitos įrankis. Visi įrankiai paskaitose yra vaizduojami blokeliais, kurie VMA sistemoje yra suprojektuoti taip, kad būtų nesunkų pridėti naują ar modifikuoti esamą.

Šiuo atveju, mes modifikuosime jau sukurtą valdymo skydo blokelį (30 pav.). Tam buvo pakeistas blokelių komponentas „Student Model“.

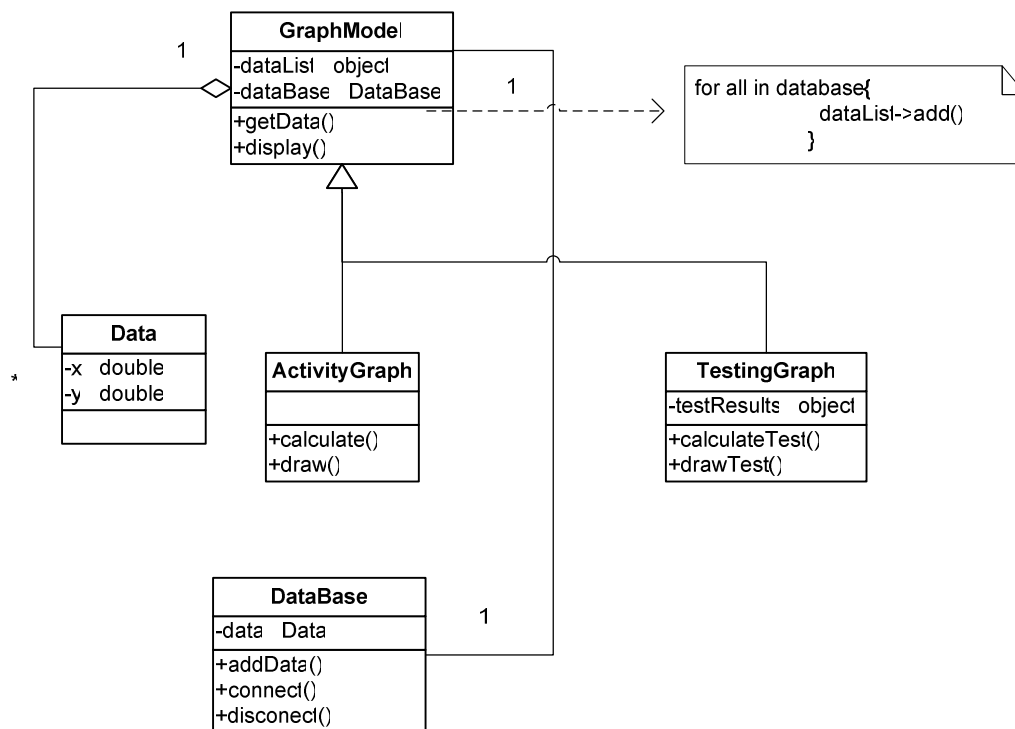


30 pav. Valdymo blokelių vartotojo sąsaja

Paveikslėlyje (30 pav.) matome nuorodą į modelio peržiūrą „Preview Model“. Spustelėjus nuorodą, galima peržiūrėti pasirinkimus, grafinę, tekstinę ir kitokią modelio informaciją. Temos grafiniame režime vaizduojamos dviem grafikais: intensyvumo ir testavimo rezultatai procentais ir

mokymosi laiko įvertinimas. Taip pat vaizduojama tekstinė informacija: pasiūlymai ir pasirinkimai. Toliau aptarsime loginę šio komponento struktūrą.

Informacijos išgavimo ir apdorojimo logiką pateikiame klasių struktūrų diagramų pavidale.



31 pav. Studento modelio klasių struktūra

Kaip matome iš diagramos (31 pav.), studento modelio duomenų bazės valdymo klasė dirba su tais pačiais duomenimis, kaip ir resurso vaizdavimo duomenų bazės klasė. Todėl protinga būtų šias klases apjungti į vieną.

Tuo tarpu kiekvienas grafinis komponentas bus realizuojamas paveldinčioje klasėje, kurios apjungiamos „GraphModel“ klase. Tai nebūtinai turi būti grafinės informacijos išvedimas, kaip jau buvo minėta anksčiau, tačiau iš diagramos galima aiškiai matyti, kad naujo komponento pridėjimas tikrai nėra sudėtingas procesas. Pakanka aprašyti naują klasę, studento modelio išvedimui panaudojant privalomas procedūras.

4.4. Skyriaus apibendrinimas

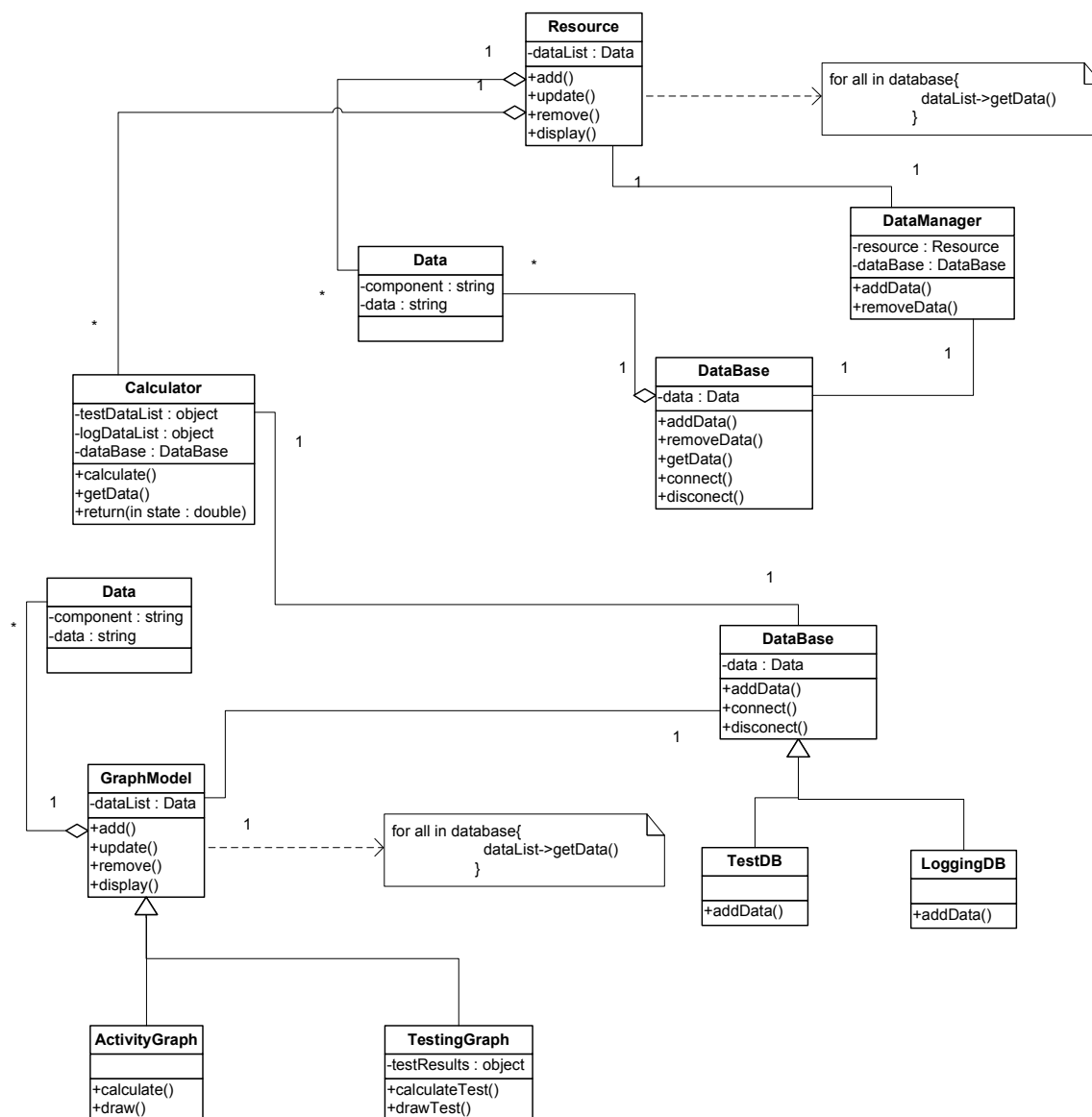
Visų pirma apibrėšime keletą reikalavimų kodui bei pačiam realizacijos procesui. Turima galvoje sistemos lankstumo ir pakartotinio kodo panaudojimo prasmė.

Siekiant, kad sistema būtų lanksti ir kodas tvarkingas, reikia atsižvelgti į plačiai paplitusias, programavime taikomas, nusistovėjusias struktūras. Šių struktūrų pritaikymas turėtų suteikti

sistemai aiškumo, o tai reikalinga, kadangi sistema yra atviro kodo. Be to, realizacijos proceso metu suprojektuotos diagramos turi būti praplečiamos ar papildomos.

Modifikacijos projektavimo metu buvo suformuotos diagramos ir reikalavimai realizacijai. Tai apima tiek loginę sistemos dalį, tiek vartotojo sąsajos reikalavimus. Kadangi kai kurie komponentai jau įgyvendinti ankstesniuose darbuose, jie nėra aprašyti. Prireikus, esamus komponentus galima bus modifikuoti realizacijos metu. Be to, lygiagrečiai šitam darbui yra vykdomi ir papildomi sistemos modifikavimo darbai, todėl turi būti atsižvelgta į juos, siekiant išvengti nesuderinamumo. Modifikacijos kol kas nebus pritaikomos naujoms VMA versijoms.

Iš šio skyriaus diagramų bei jų aprašymų galime pastebėti sąsajas tarp skirtingų komponentų, todėl pamėginsime jas apjungti į vieną, kad pamatytume bendrą modifikacijos struktūrą.



32 pav. Bendra modifikacijos struktūra

Sujungus diagramas, matomas bendras posistemės vaizdas (31 pav.). Tai nebūtinai turi sutapti su failų struktūra ir greičiausiai nesutaps. Tačiau po apjungimo mes sutaupome vieną duomenų bazės valdymo klasę.

Čia nėra galutinė realizacijos struktūros diagrama, kadangi ji labai abstrakti, realizavimo metu diagramos bus nuolat atnaujinamos.

Rezultatai

Darbo metu buvo išanalizuotos galimybės studento modelio atvaizdavimui nuotolinių studijų sistemos modifikacijoje. Buvo iškeltos problemos modelio įgyvendinimui bei nustatyti būdai ir pasirinktos priemonės modifikacijai realizuoti.

Taip pat darbo eigoje, buvo įvertintas studento modelio pritaikymo efektyvumas bei reikalingumas.

Buvo atliktas eksperimentas, siekiant nustatyti galimą SRS panaudojimą studento modelio skaičiavimams gauti, atlikta programinės įrangos galimybių analizė.

Siekiant sukurti programinį prototipą, buvo apibrėžta modifikacijos vizija, panaudojant UML diagramas – identifikuoti vartotojai, suformuluotas mokymosi procesas, organizacinis bei resursų modeliai, nustatyta sistemos tikslų hierarchija.

Galiausiai buvo sukurtas programinis kodas, failai, duomenų bazės modifikacija, įdiegimo komponentai.

Išvados

1. Daugelyje literatūros šaltinių remiamasi panašios struktūros besimokančiojo modeliu, skiriasi tik modelio sudėtingumas ir priemonės, kuriomis perteikiama besimokančiojo modelio informacija.
2. Grafinės diagramos leidžia pakankamai aiškiai ir koncentruotai atvaizduoti besimokančiajam mokymosi proceso eigą ir pasikeitimus.
3. Tam tikri programinės įrangos ribojimai neleidžia užtikrinti nuolatinio besimokančiojo sekimo informacijos saugojimo ir riboja galimybes nustatyti tikslų besimokančiojo laiko intervalą, praleistą studijuojant mokomąją medžiagą. Šis trūkumas buvo dalinai kompensuotas prisijungimų sumavimu. Nors, edukacinių požiūriu, specialisto pakeitimas sistema nėra priimtinas ir vertinamas kontraversiškai, darbe suformuluotas, specialisto sprendimų perkėlimas besimokančiajam, patvirtina tokią galimybę.
4. Darbe neišvengta tam tikrų problemų realizacijos metu, kalbant apie skaičiavimus, reikia statistinių duomenų, siekiant tinkamai parinkti koeficientus, nors ne visi iš jų labai įtakoja modelio atvaizdavimą. Sistemos prototipas paruoštas bandymui su realiais vartotojais. Modifikacijos aktualumas ir novatoriškumas leidžia kalbėti apie darbo tęstinumą.

Perspektyvos

Ateityje numatoma atlikti sistemos bandymus su realiais vartotojais. Tai labai svarbu, nes būtina įsitikinti sistemos efektyvumu bei parinkti reikiamus koeficientus. Susumavus bandymų rezultatus, bus galima pereiti į sekantį sistemos parengimo etapą – galutinę realizaciją.

Jau yra užmegzti tam tikri ryšiai su „Moodle“ atviro kodo sistemos kūrėjais, ateityje planuojama paruošti sistemos modifikaciją, kaip papildomai įdiegiamą paketą ir pateikti vartotojams. Be to, bus siekiama bendradarbiauti su sistemos „Moodle“ kūrėjais, siekiant patobulinti modifikaciją ir įdiegti adaptyvią aplinką, jau ne kaip papildomą, o kaip bendrą sistemos dalį.

Literatūra

- [1] Aroyo, L. & Dicheva, D. (2000). Domain and user knowledge in a web-based courseware engineering course, knowledge-based software engineering. In T. Hruska, M. Hashimoto (Eds.) Joint Conference knowledge-Based Software Engineering 2000 (pp. 293-300);
- [2] Baziukaitė D. (2006) Approach to an Adaptive and Intelligent Learning Environment. Advances in Computer, Information, and Systems Sciences, and Engineering Proceedings of IETA 2005, TeNe 2005 and EIAE 2005, Elleithy, K.; Sobh, T.; Mahmood, A.; Iskander, M.; Karim, M. (Eds.), XV, Springer:399-406
- [3] Baziukaitė D., Idzelytė D., Vaira Ž., Vičkus V. (2006) Virtualios mokymo(si) aplinkos adaptivi intelektualiai posistemė: koncepcinis modeliavimas ir diegimas. Technologijos mokslo darbai vakarų Lietuvoje V, KU, 141-151;
- [4] Baziukaitė D. (2005), Making virtual learning environment more intelligent: the problem of agent's mental state recognition. Liet. matem. rink., T. 45, spec. nr., 459-464;
- [5] Baziukaitė D. (2004), Making virtual learning environment more intelligent: application of Markov decision process. Liet. matem. rink., T. 44, spec. nr., 797-801;
- [6] Baziukaitė D. (2002) Adaptivios sistemos nuotolinėse studijose. Atviras ir nuotolinis švietimas žinių visuomenei, mokslinės konferencijos pranešimų tezės, Vilnius, 20-24;
- [7] Baziukaitė D., Bielskis A. A., Ramašauskas O. (2002) Applying adaptive learning principles for the e-studies. Liet. matem. rink., T. 42, spec. nr., 214-218;
- [8] Begičević N., Divjak B., Hunjak T. (2006) Development of AHP based model for decision making on e-learning implementaton. 17th International Conference on Information and Intelligent Systems IIS 2006, Varaždin, Croatia
- [9] Brusilovsky, P., Karagiannidis, C. and Sampson, D. (2004) Layered evaluation of adaptive learning systems. Int. J. Cont. Engineering Education and Lifelong Learning, Vol. 14, Nos. 4/5, pp.402–421;
- [10] Brusilovsky, P. and Tasso, C. (2004) Preface to special issue on user modeling for Web information retrieval. User Modeling and User Adapted Interaction 14 (2-3), 147-157;
- [11] Conati C and McLaren H. (2005) Data-driven Refinement of a Probabilistic Model of User Affect. Proceedings of UM2005 User Modeling: Proceedings of the Tenth International Conference, Lecture Notes in Computer Science, Volume 3538/2005, Springer Berlin / Heidelberg;
- [12] Conati C., Merten C, Muldner K., and Ternes D. (2005) Exploring Eye Tracking to Increase Bandwidth in User Modeling. Proceedings of UM2005 User Modeling: Proceedings of the Tenth International Conference. Lecture Notes in Computer Science, Volume 3538/2005, Springer Berlin / Heidelberg;
- [13] Conati, C. and Bunt, A. (2004) Student Modeling for Open Learning Adaptive Hypermedia. To appear in Proceedings of the First International Workshop on Authoring of Adaptive and Adaptable Educational Hypermedia, in conjunction with WBE 2004, International Conference on Web-Based Education, Innsbruck, Austria;

- [14] Conati C. and Manske M. (2005) Modelling Learning in Educational Games. Proceedings of AIED 05, Proceedings of the 12th International Conference on AI in Education, Amsterdam, July 19-23;
- [15] Cristea A. I. (2004) What can the Semantic Web do for Adaptive Educational Hypermedia. International Peer-Reviewed On-line Journal "Educational Technology and Society" 7(4), Special Issue on Ontologies and the Semantic Web for E-learning, IEEE, LTSC, 2004, pp 40-58;
- [16] Cristea, A., De Bra, P. (2002) ODL Education Environments Based on Adaptivity and Adaptability. Proceedings of the AACE ELearn'2002 conference, pp. 232-239;
- [17] Dagger, D., Wade, V. (2005) Personalisation for All: Making Adaptive Course Composition Easy, IBM CASCON, Dublin, Ireland;
- [18] Dimitrova, V., Denaux, R., Aroyo, L., (2005) OWL-OLM: Interactive Ontology-based Elicitation of User Models. Workshop on Personalisation for the Semantic Web PerSWeb05 at 10th International Conference on User Modeling , Edinburgh, UK, 23-29;
- [19] Dimitrova, V. & Mazza, R., (2005) Visualising Student Tracking Data to Support Instructors in Web-Based Distance Education. 9th IEEE Conference on Information Visualisation IV05 , London, UK, 6-8;
- [20] Dimitrova, V., Denaux, R., Aroyo, L. (2005) An Approach for Ontology-based Elicitation of User Models to Enable Personalization on the Semantic Web. 14th International World Wide Web Conference , Chiba, Japan;
- [21] Dimitrova V., Buffi V., Lugano (2003) An Adaptive Meta-model for E-learning. School of Computing, University of Leeds;
- [22] Dolog P. and Schaafer M. (2004) Learner Modeling on the Semantic Web. L3S Research Center, University of Hannover, Expo Plaza 1, 30539 Hannover, Germany;
- [23] Dufresne, A., Paquette, G. ExploraGraph (2000) A flexible and adaptive interface to support distance learning. Proceedings of the Ed-Media'2000, pp. 269-274., AACE, Montreal;
- [24] Eriksson H. E. and Penker M. (2000) Business Modeling with UML: Business Patterns at Work. 2000 m.;
- [25] Flor G. De La (2004) User Modeling & Adaptive User Interfaces, ILRT Research Report Number: 1085;
- [26] Henze N. (2005) Personal Readers: Personalized Learning Object Readers for the Semantic Web In: C. K. Looi, G. McCalla, B. Bredeweg and J. Breuker (eds.), Artificial Intelligence in Education, IOS Press, Amsterdam, Netherland, pp. 274-281;
- [27] Henze N., Baumgartner R., Herzog M. (2005) The Personal Publication Reader: Illustrating Web Data Extraction, Personalization and Reasoning for the Semantic Web. ESWC 2005: (515-530);
- [28] Henze N. (2005) E-Learning in the Semantic Web: The Personal Learning Object Readers. Proceedings of 2nd Joint Workshop on Cognition and Learning through Media-Communication for Advanced e-Learning 2005 (JWSJG05), Tokyo, Japan;
- [29] Henze N. and Abel F. (2005) User Awareness and Personalization in Semantic Portals. Proceedings of 4th International Semantic Web Conference (ISWC2005), Galway, Ireland.;

- [30] Henze N., Ghita S. and Nejd W. (2005) Task Specific Semantic Views: Extracting and Integrating Contextual Metadata from the Web. Proceedings of 1st Workshop on "The Semantic Desktop - Next Generation Personal Information Management and Collaboration Infrastructure" (SemDesk2005), Galway, Ireland;
- [31] Jameson, A. & Schultheis, H., (2004) Assessing Cognitive Load in Adaptive Hypermedia Systems: Physiological and Behavioral Methods. In W. Nejd & P. De Bra (Eds.), Adaptive hypermedia and adaptive web-based systems: Proceedings of AH 2004 (pp. 225–234), Berlin: Springer;
- [32] Jameson, A. (2001) User-Adaptive and Other Smart Adaptive Systems: Possible Synergies. Proceedings of the First EUNITE Symposium, Tenerife. Extended abstract of invited presentation;
- [33] Jameson, A. (2006) Adaptive Interfaces and Agents. In J. Jacko & A. Sears (Eds.), Human-computer interaction handbook (2nd ed.). Mahwah, NJ: Erlbaum. In press;
- [34] Jameson, A., Bohnenberger, T., Jacobs, O., & Aslan, I. (2005) Decision-Theoretic Planning Meets User Requirements: Enhancements and Studies of an Intelligent Shopping Guide. In H. Gellersen, R. Want, & A. Schmidt (Eds.), Pervasive computing: Third international conference (pp. 279–296), Berlin: Springer;
- [35] Jameson, A. (2005) User Modeling Meets Usability Goals. In L. Ardissono, P. Brna, & A. Mitrovic (Eds.), UM2005, User Modeling: Proceedings of the Tenth International Conference (pp. 1–3); Berlin: Springer;
- [36] Jedlitschka, A. & Althoff, K.-D. (2003) Case-Based User Modeling in an Experience Management System. KI (1)2003, 46-49;
- [37] Lee Y. and Chong ST Q. (2002) Collaborative and Sharable Learning in Multi-agent Systems. Journal of Interacting with Computers, 5(1), pp. 33- 55;
- [38] Lin T., Kinshuk & McNab P. (2005). Supporting Inductive Reasoning in Adaptive Virtual Learning Environment. In Uskov V. (Ed.), Proceedings of the 4th IASTED International Conference on Web-Based Education (WBE 2005) (February 21-23, 2005, Grindelwald, Switzerland), Anaheim, CA, USA: ACTA Press, 562-566;
- [39] Paneva D., Zhelev Y. (2007) Models, Techniques and Applications of E-learning Personalization. Information Technologies & Knowledge Volume 1/2007, 244-251;
- [40] Razmerita L., Angehrn A., Nabeth T. (2005) On the Role of User Models and User Modeling in Knowledge Management Systems. Centre for Advanced Learning Technologies, Fontainebleau, France;
- [41] Razmerita L., Brna P., Cooper B., (2001) Marching to the wrong distant drum: Pedagogic Agents, emotion and student modeling. Workshop on Attitude, Personality and Emotions in User-Adapted Interaction in conjunction with User Modeling 2001, Sonthofen, Germany;
- [42] Razmerita, L., Angehrn, A., Nabeth, T., Roda, C. (2004) InCA: An Intelligent Cognitive Agent-based Framework for Adaptive and Interactive Learning. Conference on Cognition and Exploratory Learning in Digital Age, IADIS CELDA 2004, Lisbon, Portugal;
- [43] Razmerita L., Roda C., Angehrn A., Nabeth T., (2003) Using conversational agents to support the adoption of knowledge sharing practices. Interacting with Computers, Special Issue on Intelligence & Interaction in Community-based Systems, Elsevier, pp. 57-89;

- [44] Razmerita, L., Angehrn, A., Maedche, A. (2003) Ontology based user modeling for Knowledge Management Systems, Proceedings of the User Modeling Conference, Pittsburgh, USA, Springer-Verlag, pp.213-217;
- [45] Rutkauskienė D. (2003) Nuotolinis Mokymasis. Kaunas „Tachnologija“ 2003m.;
- [46] Masthoff, J. (2004). Selecting news to suit a group of criteria: An exploration. *Proceedings of the Fourth Personalized TV workshop*, associated with AH04;
- [47] Mazza R. (2003) Visualising Student Tracking Data to Support Instructors in Web-Based Distance Education. Faculty of Communication Sciences University of Lugano;
- [48] Sinan Si' Alhir (2003) Learning UML, Publisher O'Reilly, 2003 m.;
- [49] Specht M., Kravcik M., (2004) Authoring Adaptive Courses – ALE Approach. In: Proc. of the WBE 2004 Conference, Innsbruck;
- [50] Trnková J., Rößling G., Mühlhäuser M. (2003) L4 - A sophisticated Adaptive E-Learning Laboratory. In AELM: Workshop on Adaptive E-Learning and Metadata, (241-245).

Iliustracijos















1 pav.	Peter Brusilovsky siūloma adaptyvaus mokymosi organizavimo schema [9]	11
2 pav.	Grace de la Flor siūlomas paprasčiausias vartotojo modelis [25]	13
3 pav.	Besimokančiojo modelis panaudotas sujungiant dvi mokymuisi skirtas sistemas [18]	13
4 pav.	Grafinis besimokančiojo modelio vaizdavimas nuotolinių studijų sistemoje [19] (žr. prieduose)	15
5 pav.	Modelis, remiamas besimokančiojo tikslų, tikslų įgyvendinimo ir emocijų modeliu [16]	16
6 pav.	Sistemos tikslų hierarchija [2]	17
7 pav.	Įverčių nustatymas	23
8 pav.	Įverčių rūšiavimas ir normalizavimas	24
9 pav.	Temomis suskirstyta mokomoji medžiaga	24
10 pav.	Įverčiai parinkti pagal testų rezultatus	25
11 pav.	Testo rezultatų normalizacija	25
12 pav.	Įverčiai parinkti pagal mokymosi laiką	25
13 pav.	Temos rezultatų normalizacija	25
14 pav.	Įverčiai parinkti pagal atliktas užduotis	26
15 pav.	Temos rezultatų normalizacija	26
16 pav.	Galutiniai rezultatai	27
17 pav.	Dinaminis modelis	28
18 pav.	Nuotolinės sistemos ir jos organizacinės aplinkos “Raiškūs paveikslėlis”	32
19 pav.	Nuotolinių studijų tikslų medis	35
20 pav.	Mokymo(si) proceso modelis nuotolinių studijų sistemoje	37
21 pav.	Detalizuotas mokymo(si) proceso modelis	38
22 pav.	Nuotolinių studijų organizacinis modelis	40
23 pav.	Sprendimų priėmimo procesas	43
24 pav.	Esybių ryšių duomenų bazės fragmento diagrama	46
25 pav.	VMA resursų klasių struktūra [3]	47
26 pav.	Resurso modifikacijos klasių struktūra	48
27 pav.	Failų sistema	48
28 pav.	Spalvų perėjimas	50
29 pav.	Resurso vaizdavimo komponento klasių struktūra	50
30 pav.	Valdymo blokelių vartotojo sąsaja	51

31 pav.	Studento modelio klasių struktūra	52
32 pav.	Bendra modifikacijos struktūra	53
33 pav.	Resursų išdėstymo pagal temas langas	64
34 pav.	Resursų užpildymo langas dėstytojo režime	65
35 pav.	Besimokančiojo modelio pasirinkimų atvaizdavimo komponentas	65
36 pav.	Besimokančiojo modelio resursų grafinis atvaizdavimo komponentas	66
37 pav.	Besimokančiojo modelio pasiūlymų atvaizdavimo komponentas.....	66
38 pav.	Besimokančiojo modelio resursų grafinis atvaizdavimo komponentas	67
39 pav.	Besimokančiojo modelio pasiūlymų atvaizdavimo grafinis komponentas	67
40 pav.	Duomenų bazė papildyta komponento lentele	67
41 pav.	Duomenų bazė papildyta resursų „sekimo“ lentele.....	67
42 pav.	Padidintas, grafinis besimokančiojo modelio vaizdavimas nuotolinių studijų sistemoje	
[19]	68	

Priedai

Prieduose pateikiami pagrindiniai realizacijos komponentai bei diskas su realizuota modifikacija. Disko turinys: kodas, dokumentacija, duomenų bazė.

1 priedas – vartotojo sąsajos komponentai.

Curriculum	
1	Skaičiavimo sistemos
1.1	Dešimtainė sistema  Dešimtainė sistema 39.5 %  Dešimtainė sistema
1.2	Dvejetainė sistema  Dvejetainė sistema 12.64 %  Dvejetainė sistema
1.3	Veiksmai su dvejetainiais skaičiais  Veiksmai su dvejetainiais skaičiais 29.94 %  Veiksmai su dvejetainiais skaičiais  Tempti/Mesti veikla
1.4	Šešioliktainė sistema  Šešioliktainė sistema  Šešioliktainė sistema 17.29 %
1.5	Dešimtainių skaičių vertimas į kitas sistemas  Dešimtainių skaičių vertimas į kitas sistemas  Dešimtainių skaičių vertimas į kitas sistemas 12.64 %
1.6	Teigiami ir neigiami skaičiai  Teigiami ir neigiami skaičiai 31.65 %  Teigiami ir neigiami skaičiai
2	Asemblerio kalbos ypatumai
2.1	Asemblerio kalbos ypatumai  Asemblerio kalbos ypatumai 12.64 %  Asemblerio kalbos ypatumai  Kryžiažodis

33 pav. Resursų išdėstymo pagal temas langas

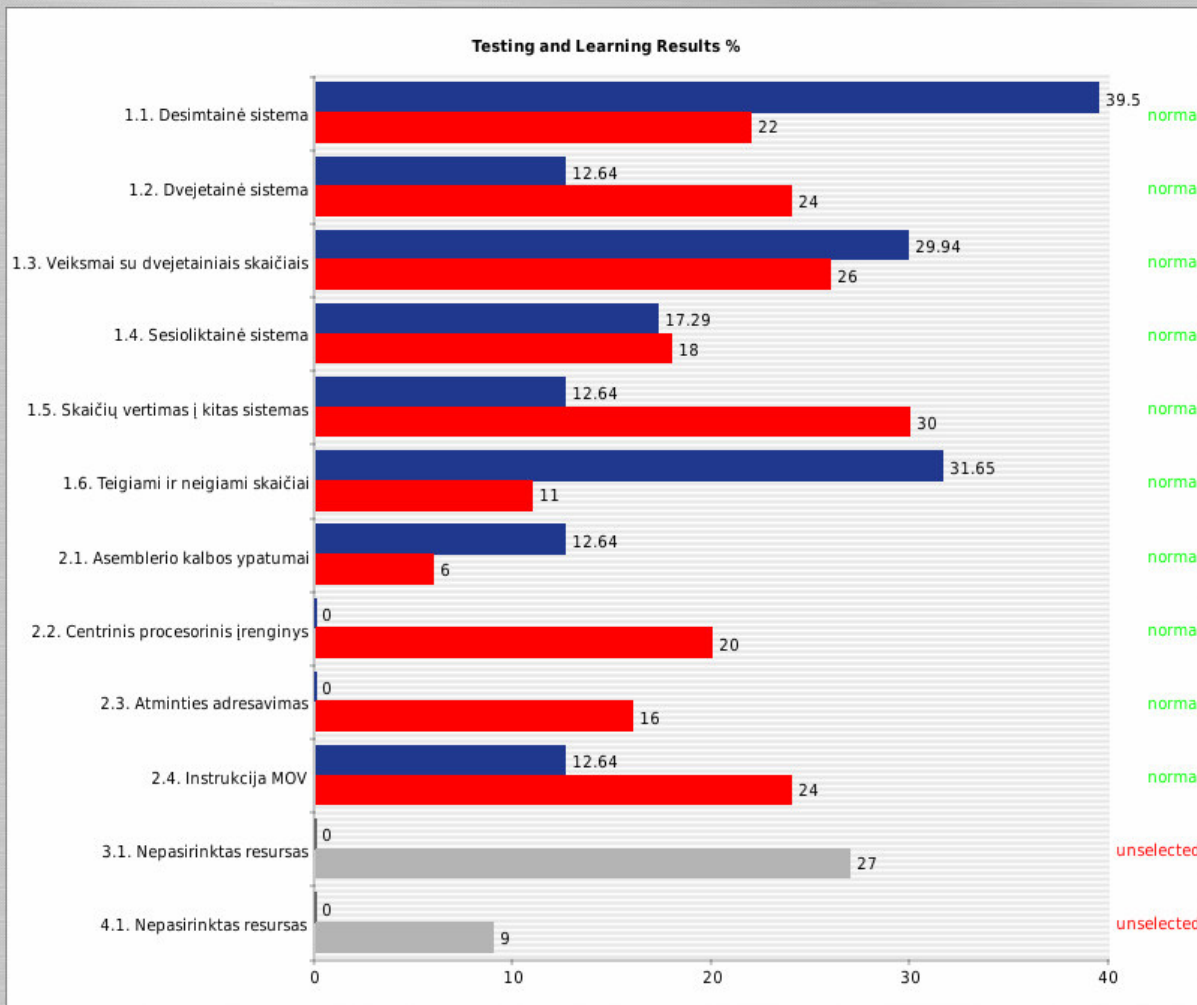


34 pav. Resursų užpildymo langas dėstytojo režime



35 pav. Besimokančiojo modelio pasirinkimų atvaizdavimo komponentas

Testing and learning results



36 pav. Besimokančiojo modelio resursų grafinis atvaizdavimo komponentas

Suggestions for Testing and learning results

The Worst Result!!!

1.5. Skaičių vertimas į kitas sistemas

[Change](#)

You Sould Change Learning Preferences Or Study Extra Literature For Following Chapters!!!

1.1. Desimtainė sistema

[Change](#)

You Sould Change Your Learning Plan For Following Chapters!!!

No Sugestions...

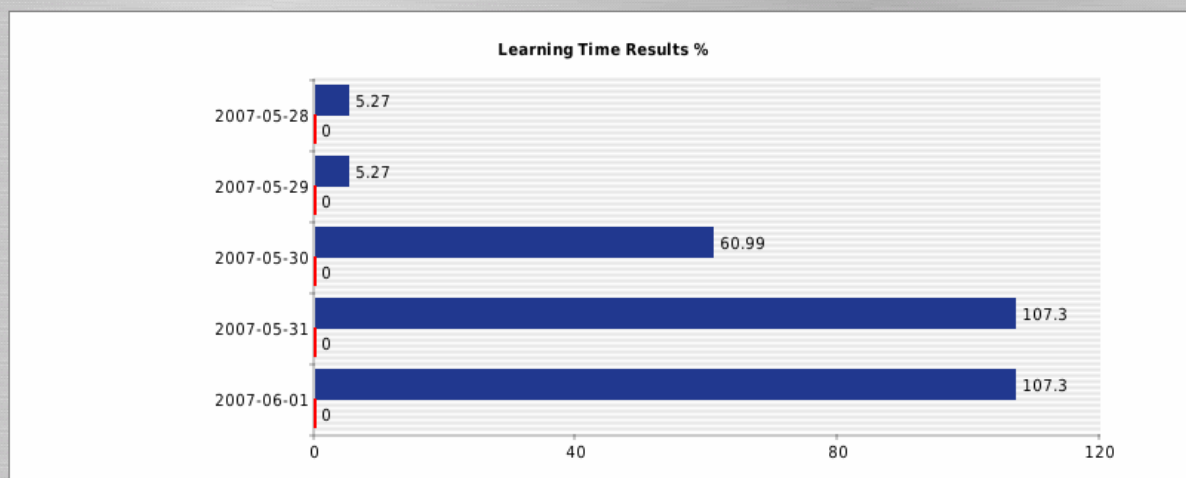
You Sould Think About Removing Following Chapters from Learning Plan because of the achieved results!!!

2.2. Centrinis procesorinis įrenginys

[Change](#)

37 pav. Besimokančiojo modelio pasiūlymų atvaizdavimo komponentas

Learning Time Results



38 pav. Besimokančiojo modelio resursų grafinis atvaizdavimo komponentas

Suggestions for Learning Time Results

You Should read more of these resources!!!

No Sugestions...

39 pav. Besimokančiojo modelio pasiūlymų atvaizdavimo grafinis komponentas

2 priedas – duomenų bazės komponentai.

	Field	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	id	int(11)			No		auto_increment
<input type="checkbox"/>	resourceid	int(11)			No		
<input type="checkbox"/>	name	varchar(255)	cp1257_bin		No		
<input type="checkbox"/>	source	varchar(255)	cp1257_bin		No		
<input type="checkbox"/>	text	text	cp1257_bin		No		
<input type="checkbox"/>	type	int(11)			No		
<input type="checkbox"/>	ordering	int(11)			No		

40 pav. Duomenų bazė papildyta komponento lentele

	Field	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	id	bigint(20)			No		auto_increment
<input type="checkbox"/>	course	int(11)			No		
<input type="checkbox"/>	userid	int(11)			No		
<input type="checkbox"/>	resourceid	int(11)			No		
<input type="checkbox"/>	quantity	int(11)			No		
<input type="checkbox"/>	access_date	varchar(10)	utf8_lithuanian_ci		No		

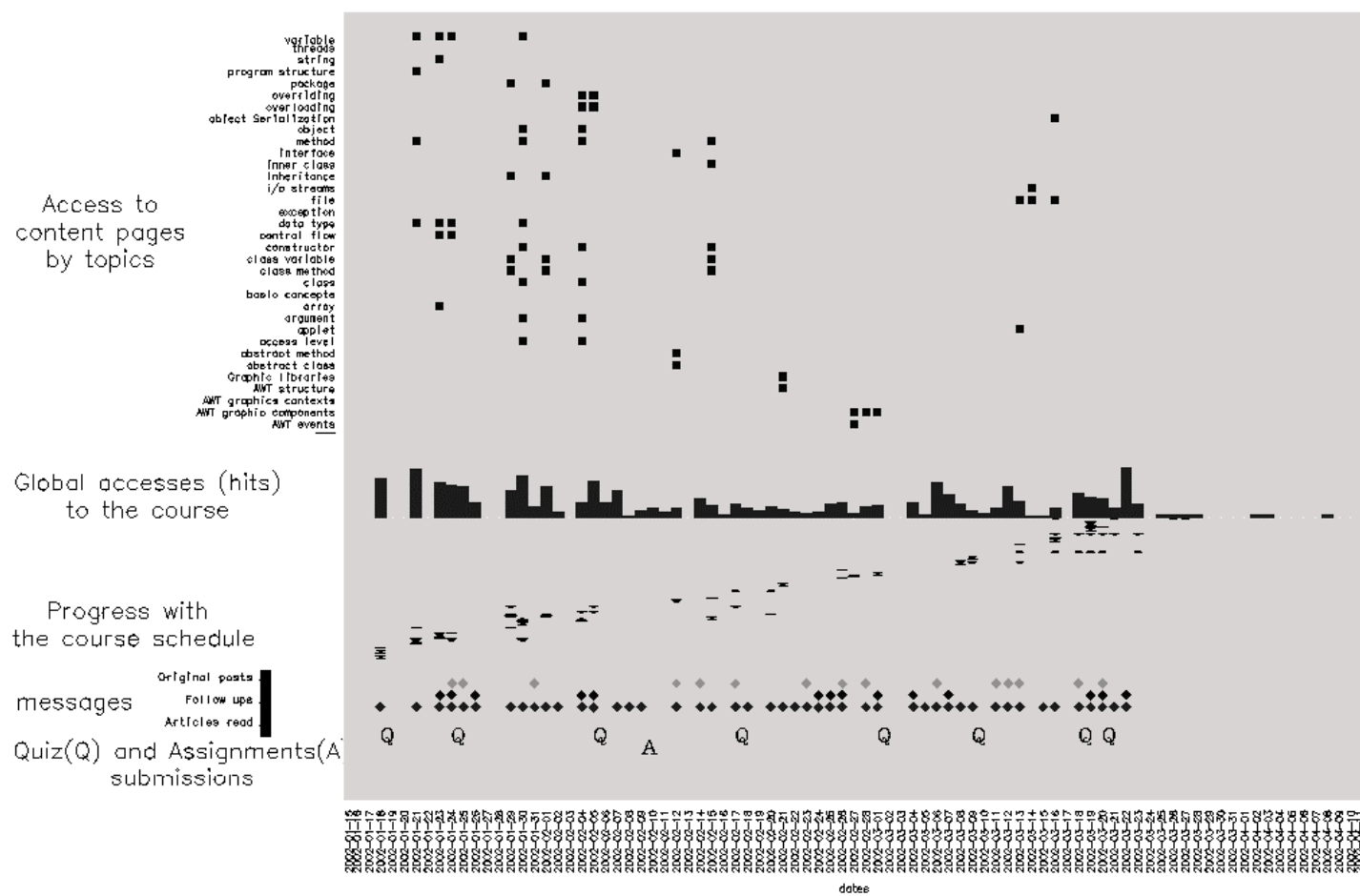
41 pav. Duomenų bazė papildyta resursų „sekimo“ lentele

3 priedas – padidintas paveikslėlis (4 pav.).

Summary of student's behaviours

from 2002-01-15 to 2002-04-11

Student: "francesco"



42 pav. Padidintas, grafinis besimokančiojo modelio vaizdavimas nuotolinių studijų sistemoje [19]

3 priedas – kompaktinis diskas.

