

Vilniaus universiteto
Komunikacijos fakulteto
Informacijos ir komunikacijos katedra

Vaidas Liubinas,
Informacijos sistemų vadybos magistro studijų programos studentas

**E. MOKYMOŠI APLINKŲ INTEGRACIJA Į
AKADEMINES INFORMACIJOS SISTEMAS**

Magistro darbas

Vadovas doc. dr. Povilas Abarius

Vilnius, 2011

Bakalauro / Magistro darbo lydraštis

Pildo bakalauro / magistro baigiamojo darbo autorius

VAIDAS LIUBINAS

(bakalauro / magistro baigiamojo darbo autoriaus vardas, pavardė)

E. MOKYMOŠI APLINKŲ INTEGRACIJA Į AKADEMINES INFORMACIJOS SISTEMAS

(bakalauro / magistro baigiamojo darbo pavadinimas lietuvių kalba)

INTEGRATION OF E-LEARNING ENVIRONMENTS INTO ACADEMIC INFORMATION SYSTEMS

(bakalauro / magistro baigiamojo darbo pavadinimas anglų kalba)

Patvirtinu, kad bakalauro / magistro baigiamasis darbas parašytas savarankiškai, nepažeidžiant kitiems asmenims priklausančių autorių teisių, visas baigiamasis bakalauro/ magistro darbas ar jo dalis nebuvo panaudotas kitose aukštosiose mokyklose.

(bakalauro / magistro baigiamojo darbo autoriaus parašas)

Sutinku, kad bakalauro/ magistro baigiamasis darbas būtų naudojamas neatlygintinai 5 metus Vilniaus universiteto Komunikacijos fakulteto studijų procese.

(bakalauro / magistro baigiamojo darbo autoriaus parašas)

Pildo bakalauro / magistro baigiamojo darbo vadovas

Bakalauro / magistro baigiamąjį darbą ginti

(įrašyti – leidžiu arba neleidžiu)

(data)

(bakalauro / magistro baigiamojo darbo vadovo parašas)

Pildo instituto/ katedros, kuriojančios studijų programą, reikalų tvarkytoja

Bakalauro / magistro baigiamasis darbas įregistruotas

(instituto/ katedros, kuriojančios studijų programą, pavadinimas)

(data)

(instituto/ katedros reikalų tvarkytojos parašas)

Pildo instituto/ katedros, kuriojančios studijų programą, vadovas

Recenzentu skiriu

(recenzento vardas, pavardė)

(data)

(instituto/ katedros vadovo parašas)

Pildo recenzentas

Darbą recenzuoti gavau.

(data)

(recenzento parašas)

REFERATAS

Liubinas, Vaidas

Li614 E. mokymosi aplinkų integracija į akademines informacijos sistemas : magistro darbas / Vaidas Liubinas ; mokslinis vadovas doc. Povilas Abarius ; Vilniaus universitetas. Komunikacijos fakultetas. Informacijos ir komunikacijos katedra. – Vilnius, 2011. – 79 [7] lap. : iliustr. – Mašinr. – Santr. angl. – Bibliogr.: p. 76-79 (52 pavad.).

UDK 004.414.23 : 378

Reikšminiai žodžiai: akademinės informacijos sistemos, e. mokymosi aplinkos, integracija, integravimo strategija ir architektūra, autentifikavimas, duomenų federacija, integracijos mechanizmas.

Magistro *darbo objektas* – akademinės informacinės sistemos. *Darbo tikslas* – suprojektuoti ir sukurti akademiinių informacinių sistemų ir e. mokymosi aplinkos integracijos procesų algoritmą (mechanizmą). Pagrindiniai *darbo uždaviniai* - apžvelgti ir išanalizuoti e. mokymosi aplinkų naudojamą technologijas ir architektūras, apžvelgti ir išanalizuoti pagrindinius akademiinių informacinių sistemų technologinius sprendimus, išanalizuoti galimus integracijos būdus ir pateikti e. mokymosi aplinkų ir akademiinių informacinių sistemų integravimo proceso strategiją ir architektūrą, naudojamų komponentų, technologijų aprašymą apibrėžiant integracijos mechanizmą.

Šiame darbe, išanalizavus e. mokymosi aplinkas, jų naudojamą technologijas ir architektūras, padarytos išvados, kad asinchroninio tipo architektūros aplinkos, dėl nesudėtingo diegimo ir atviro kodo programinių paketų pasirinkimo yra paprasčiausia ir pigiausia e. mokymosi organizavimo forma, turinti daugiau galimybių integruotis į kitas informacijos sistemas dėl galimybės modifikuoti pradinį programinio paketo kodą.

Akademinės informacijos sistemos Lietuvoje yra dar naujas reiškinys. Mažuma aukštųjų mokyklų turi visiškai funkcionuojančias akademines informacines sistemas, dauguma yra įdiegimo procese, todėl tik nedaugelis aukštųjų mokyklų yra suformulavusios informacijos sistemų tikslus ir uždavinius, sistemose yra skirtingos informacijos pateikimo formos ir jų gausa, vieningo informacijos pateikimo modelio nėra, vartotojų autentifikavimas pertekliniai, besidubliuojantys ir sunkiai valdomi. Išanalizavus Lietuvos aukštųjų mokyklų naudojamą ar įdiegiamą akademines informacijos sistemas, darbe išskirti pagrindiniai informacijos sistemos veiklos funkcijos ir

uždaviniai: apskaitos ir vartotojų sąskaitų valdymas, kontrolė, informacijos kaupimas, studijų planavimas ir vykdymas, komunikacija ir sklaida.

Integravimo strateginių sprendimų analizė patvirtino, kad mišrus integravimo būdas yra dažniausias, nes informacijos sistemos dažniausiai integruojamos ir dalykiniu, ir techniniais aspektais. Integracijos architektūrinį sprendimą lemia informacijos sistemų organizavimo sprendimų visuma. Išanalizavus integracijai atlikti duomenų lygmenyje taikomus tris pagrindinius metodus, galima teigti, kad duomenų federacijos metodas yra saugiausias duomenų atžvilgiu, paprasčiausias ir reikalaujantis mažiau sąnaudų. Integracijos lygmens pasirinkimą lemia organizacijos poreikiai ir finansinės galimybės.

Atlikus sistemų ir integracijos procesų analizę, daromos išvados, kad e. mokymosi aplinkos ir akademinės informacijos sistemos integracija leistų sukurti kokybinę studijų valdymo, administravimo ir studijų tikslų įgyvendinimo aplinką. Federacinis integracijos architektūrinis sprendimas ir duomenų federacijos metodo taikymas sukuria draugiškos aplinkos įvaizdį, į kurią galima integruoti kitas informacijos sistemas ir paslaugas. Vartotojų registravimosi funkcijų integravimas leidžia išvengti vartotojų duomenų dubliavimo ir pertekliško, palengvina valdymą, o vieno prisiregistravimo metodo naudojimas palengvina vartotojo darbą skirtingose informacijos sistemose ir programinių paketų aplinkose.

Remiantis šiame darbe analizuotais integracijos procesais, apibrėžtas integravimo mechanizmas ir sudarytas jo algoritmas. Šis algoritmas suteiks galimybę nuosekliai, kryptingai ir racionaliai atlikti darbus prieš integracijos procesų realizavimą.

Magistro darbas gali būti naudingas akademinėms ir verslo institucijoms, kurios turi informacijos sistemas, naudojami e. mokymosi aplinkomis ir nori ar planuoja jas integruoti į vieną informacijos sistemą.

TURINYS

ĮVADAS.....	9
1. ELEKTRONINĖ MOKYMOSI APLINKA	11
1.1. Elektroninio mokymosi technologijų tipai	12
1.2. Elektroninio mokymosi technologijų klasifikacija	14
1.3. Elektroninio mokymosi aplinkų sistemų architektūros	17
1.3.1. Asinchroninio tipo sistemų architektūra	17
1.3.2. Sinchroninio tipo sistemų architektūra	19
1.3.3. Mišraus tipo sistemų architektūra	20
1.4. Elektroninio mokymosi aplinkų informacinės technologijos	21
1.4.1. Studentų veiklos įrankiai	22
1.4.2. Administravimo įrankiai	23
1.4.3. Žinių vertinimo įrankiai	25
1.4.4. Turinio kūrimo įrankiai	25
1.4.5. Programiniai paketai	27
2. AKADEMINĖS INFORMACIJOS SISTEMOS	30
2.1. Akademinės informacijos sistemos Lietuvoje	32
2.1.1. Vilniaus universiteto informacinės sistemos atvejis	32
2.1.2. Kauno technologijos universiteto akademinės informacijos sistemos atvejis	34
2.1.3. Vilniaus Gedimino technikos universiteto informacijos sistemos atvejis	36
2.1.4. Kitos institucijos	37
2.1.5. Akademiinių informacijos sistemų bendrumai	38
2.2. Akademinės informacijos sistemos veiklos ir uždaviniai	39
3. INTEGRACIJA	42
3.1. Integravimo strateginiai sprendimai	43
3.2. Integravimo architektūriniai sprendimai	44
3.3. Duomenų integravimas	46
3.3.1. Duomenų federacijos duomenų apjungimo metodas	48
3.4. Programinių paketų integravimas	54
4. E. MOKYMOSI APLINKOS INTEGRAVIMAS Į AKADEMINES INFORMACIJOS SISTEMAS	57
4.1. Integravimo paradigma	57
4.2. Integravimo strateginiai ir architektūriniai sprendimai	58

4.3.	Integruojamų sistemų duomenys	60
4.4.	Vartotojų registravimosi funkcijų integravimas	62
4.4.1.	LDAP katalogas	62
4.4.2.	SSO galimybė	66
4.5.	Integravimo mechanizmo algoritmas	70
	IŠVADOS	72
	Integration of e-learning environments into academic information systems (summary)	74
	Bibliografinių nuorodų sąrašas	76

Santrumpų sąrašas

- BPEL – biznio procesų valdymo kalba (angl. Business Process Execution Language);
- CAS – centrinė atpažinimo sistema, vieno prisiregistravimo metodo protokolas (angl. Central Authentication Service);
- CourseLM – elektroniniams mokymo kursams aprašyti skirta aprašymo kalba, realizuota XML pagrindu;
- ETL - programinė įranga, kuri padeda konsoliduoti skirtingus duomenis, su esančius skirtingose formose ar formatuose (angl. extract, transform, and load);
- HTTP – hiperteksto perdavimo protokolas (angl. Hypertext Transfer Protocol);
- IMAP - elektroninio pašto serverio protokolas (angl. Internet Message Access Protocol);
- IMS – mokymosi objektų standartas (angl. Instructional Management Systems);
- LDAP – supaprastintos kreipties į katalogus protokolas (angl. Lightweight Directory Access Protocol);
- LieMSIS – Lietuvos mokslo ir studijų informacijos sistema;
- LOM - mokymo objektų meta duomenys (angl. Learning Object Metadata);
- NNTP – tinklo žinučių perdavimo protokolas (angl. Network News Transfer Protocol)
- OMG – tarptautinis, verslo technologijų ir integracijos standartų konsorciumas (angl. Object Management Group);
- OpenLDAP – atviro kodo supaprastintos kreipties į katalogus protokolas (angl. Open source Lightweight Directory Access Protocol);
- OSP - atviro kodo įrankis, skirtas skaitmeniniams įrašams apie studijas kurti (angl. Open Source Portfolio);
- POP3 – trečios versijos protokolas, naudojamas elektroninių laiškų gavimui iš serverio. (angl. Post Office Protocol 3);
- SCORM - bendro turinio objektų nuorodų modelis (angl. Sharable Content Object Reference Model);
- SOA – į paslaugas orientuota architektūra (angl. Service-Oriented Architecture);
- SOAP – informacijos, skirtos internetui ir perdavimui tinklais, struktūrizavimo protokolas (angl. Simple Object Access Protocol);
- SQL – programavimo sąsaja ir programavimo kalba, skirta informacijos tvarkymui reliacinių duomenų bazių sistemose (angl. Structured Query Language);
- SSL – kriptografinis protokolas, skirtas informacijos, sklindančios internete apsaugojimui šifruojant (angl. Secure Sockets Layer);
- SSO - vieno prisiregistravimo metodas (angl. Single Sign-On);

WebDAV - protokolas, sukurtas HTTP protokolo pagrindu, pridėjus tam tikrus patobulinimus, skirtus grupei vartotojų atlikti veiksmus su bylomis, esančiomis nutolusioje sistemoje (serveryje), tuo pačiu sukuriant išpūdį, tartum jie būtų sistemoje, su kuria tuo metu dirba vartotojas. (angl. Web-based Distributed Authoring and Versioning);

WSDL – web paslaugų aprašomoji kalba (angl. Web Services Description Language).

ĮVADAS

Šiuo metu naudojama nemažai informacijos sistemų. Pagrindinės jų funkcijos - pasitelkiant informacines technologijas, taupyti žmogiškuosius išteklius ir brangų žmogaus laiką renkant, susisteminant, kaupiant ir pateikiant norimais pjūviais reikalingą ir savalaikę informaciją, tenkinančią vartotojo informacinius poreikius. Informacinė sistema - materialinė sistema, galinti priimti, kaupti, saugoti, keisti, naudoti, skleisti informaciją [13]. Informacinės sistemos gelbsti, ieškant ir priimant teisingus ir sėkmingus sprendimus. Informacinė sistema dažnai būna vienas iš optimaliausių vartotojo problemų sprendimų būdų, susiduriant su dideliais informacijos kiekiais. Modernios informacinės sistemos – tai optimaliausias būdas apdoroti visą mus supančią informaciją.

Nemažai informacijos sistemų naudoja ir aukštosios mokyklos. Pirmosios informacijos sistemos akademinėje aplinkoje, vienokia ar kitokia forma, jau naudojamos daugiau nei dešimtmetį. Tai sudėtingi informaciniai dariniai, sudaryti iš skirtingais laikotarpiais kurtų, skirtingus vartotojų poreikius tenkinančių, posistemų. Jos kaupia, susistemina ir teikia reikiamą informaciją skirtingų poreikių ir lygmenų vartotojams: dėstytojai tvarko studijų modulių aprašus, veda dalykų įvertinimus, studijų administratoriai talpina su studijų tvarka susijusius dokumentus, kiti padaliniai talpina apklausas ir informaciją, susijusią su studijomis ir socialiniais studentų reikalais, studentai turi galimybę matyti vertinimus, studijų planą, studijuojamų dalykų aprašus ir t.t. Informacija pateikiama susisteminta, nustatomi vartotojų lygmenys. Prieiga prie informacijos sistemos autentifikuota. Šios informacinės sistemos, kurios aptarnauja visą studijų procesą ir su jais susijusią informaciją, vadinamos akademinėmis informacijos sistemomis. Dažniausiai tai pagrindinė informacinė sistema akademinėje institucijoje, tačiau gali būti ir kitų.

Lietuvos universitetuose nuo 1996 metų pradėti kurti nuotolinių studijų centrai, papildę tradicinį studijų modelį nuotolinėmis studijomis arba e. kursais kaip pagalbine mokymo - mokymosi priemone. Tokių centrų pagrindinis uždavinys buvo sukurti e. mokymosi pagrindus, sukuriant studijoms reikalingus modulius, užpildančius elektroninę mokomąją aplinką. Tokia aplinka suprantama kaip palankių ir kiekvienam studentui internete prieinamų mokymosi ir technologinio ugdymo sąlygų sudarymas - mokytis patogiu laiku, patogioje vietoje, nuolat atnaujinamoje mokymo aplinkoje šiuolaikiniais metodais. Pirmosios elektroninės mokymosi sistemos buvo traktuojamos kaip priemonė nuotoliniam mokymuisi pagerinti. Dabar ši technologija naudojama ir kaip būtinas mokymosi išteklius įvairioms mokymo formoms. Mokomosios medžiagos internete prieiga aktuali visiems studentams, nors išskiriama, kad ji aktualiausia išėstinių studijų studentams, tobulinantiems kvalifikaciją, besimokantiems savarankiškai.

Mokomoji aplinka internete – tai viena iš priemonių visokeriopa padėti studentams mokytis. Mokomoji aplinka skirta visiems besimokantiems, ne lygu ar tai būtų nuosekliųjų, nenuosekliųjų, nuolatinių ar iššestinių studijų studentas. Šiuo metu sukurta nemažai e. mokymosi aplinkų todėl jų pasirinkimas yra sudėtingas uždavinys bet kuriai mokymo ir mokymosi institucijai. Visgi verta pastebėti, kad visos mokymosi aplinkos yra panašios. Taip yra todėl, kad kuriant jas siekiama tų pačių tikslų. Tačiau kadangi kūrėjai turi savo idėjas ir filosofiją, tai atsispindi ir jų sukurtuose produktuose. Kiekvienoje aplinkoje galima rasti medžiagos pateikimo, žinių patikrinimo bei vertinimo, bendravimo ir bendradarbiavimo įrankius. E. mokymosi aplinkos yra mokymo informacinės sistemos su visomis informacijos sistemai būdingais požymiais.

Tačiau šios dvi aukščiau paminėtos informacinės sistemos nėra vienintelės informacijos sistemos, kuriomis naudojasi aukštosios mokyklos akademinės bendruomenės nariai. Institucijose dažniausiai naudojamos dokumentų valdymo, finansų apskaitos ir valdymo sistemos, bibliotekų duomenų bazės, katalogai ir kt.

Nagrinėjant informacinių sistemų efektyvaus realizavimo akademinėje terpėje klausimus, viena iš pagrindinių problemų tampa skirtingų informacinių sistemų sąveika, duomenų dubliavimas, duomenų perteklius ir kt. Be to, skirtingos informacijos sistemos turi savo autentifikacijos mechanizmus ir vartotojas turi įsiminti didelį kiekį identifikacinių duomenų komplektų, norėdami naudotis visomis informacijos sistemomis.

Darbo objektas – akademinės informacinės sistemos. Darbo tikslas – suprojektuoti ir sukurti akademinę informacinių sistemų ir e. mokymosi aplinkos integracijos procesų algoritmą (mechanizmą).

Uždaviniai:

1. Apžvelgti ir išanalizuoti e. mokymosi aplinkų naudojamą technologijas ir architektūras.
2. Apžvelgti ir išanalizuoti pagrindinius akademinę informacinių sistemų technologinius sprendimus.
3. Apžvelgti ir išanalizuoti galimus integracijos būdus.
4. Pateikti e. mokymosi ir akademinę informacinių sistemų integravimo proceso strategiją ir architektūrą, naudojamų komponentų, technologijų aprašymą.
5. Apibrėžti integracijos mechanizmą.

1. ELEKTRONINĖ MOKYMOSI APLINKA

Elektroninis mokymasis (toliau – e. mokymasis) tampa viena svarbiausių mokymosi priemonių. Kadangi vis daugiau mokslo organizacijų pereina prie elektroninio mokymosi, vis daugiau naudingos, skirtingos, informacijos yra platinama internetu ir vidiniuose kompiuterių tinkluose. Iššūkiu tampa problema, kaip suformuoti universalią elektroninio mokymosi aplinką visiems besinaudojantiems elektroniniu mokymusi.

Elektroninio mokymosi aplinkas (toliau – e. mokymosi aplinkos) galima apibrėžti kaip kompiuterines, santykinai atviras, aplinkas, kurios leidžia grupinius susibūrimus ir dalyvių tarpusavio sąveiką, teikia galimybę pasiekti labai įvairius mokymosi išteklius. E. mokymas apibrėžiamas kaip informacijos ir komunikacijos technologijų priemonėmis praturtintas mokymas, kai pats mokymosi procesas iš dalies yra perkeliamas į virtualią erdvę. Virtuali erdvė – tai informacijos ir komunikacijos technologijų priemonėmis suformuota, ir dažniausiai su internetu tapatinama, terpė [47]. E. mokymosi aplinka turėtų būti skiriama nuo nedidelių, atskirų programų, naudojimo, kur studentas turi atskirą, ribotą kompiuterinę mokymosi erdvę ar aplinką, atskirtą nuo auditorijos mokymosi aplinkos, kur įvairios technologijos naudojamos kaip pagalbinės įprastiems užsiėmimams [51].

Informacinių ir komunikacinių technologijų progresas pateikia naują paradigmą bendro mokymosi procese, kuri vadinama kompiuteriais paremtu mokymusi bendradarbiaujant (angl. Computer Supported Collaborative Learning). Pasak mokslininko P. Restos (2007) mokymasis bendraujant elektroninėje erdvėje yra dinamiškas procesas, kuriame technologijos palengviną žinių ir kūrimo patirties dalijimąsi per tarpusavio sąveiką ir grupinio mokymo procesus tarpdisciplininių ir tarptautinių mokslinių tyrimų srityse.

Remiantis šia teze, bet kuri kompiuterinė technologija skatinanti bendradarbiauti elektroninėje erdvėje, prisideda prie mokymosi bendradarbiaujant. Tokio mokymo privalumai yra tai, kad generuojama daugiau idėjų grupėse, naudojamos kolektyvinio naujų idėjų svarstymo priemonės, vyrauja dalyvavimas diskusijos metu ir pan. Toks mokymasis suteikia galimybę stebėti individualų mokymosi procesą vykstantį tam tikrų vartotojų grupės erdvėse, diskusijose, vertinant patiems, t.y. taikant savikontrolę, ir vertinant kuratoriams. Tuo pačiu tai skatina studentus motyvuotai mokytis.

E. mokymosi aplinkoje dėmesys labiau krypsta į bendradarbiavimo priemones, į galimybes mokymo procesą vykdyti kartu ir individualiai, bet kartu naudotis tais pačiais resursais. Pačios aplinkos turi daug panašumų ir perėmusios nemažai gerų kompiuterizuoto mokymo ir programų sistemų bruožų. Visgi e-mokymosi mintis kur kas gilesnė. Tai aplinkos, kurios apjungia geriausias

kompiuterių tinklo ir interneto savybes – jos leidžia „daug su daug“ ryšius (n:m) tarp studentų ir kuratorių, skatina bendradarbiauti ir diskutuoti [36].

Kiekviena e. mokymosi aplinka turi struktūrą, kurią, dažniausia bendrais bruožais, sudaro mokomosios medžiagos teikimo, bendravimo ir bendradarbiavimo elementai, kontrolės ir studentų veiklos peržiūros priemonės. Be to e. mokymosi aplinkos jungia skirtingus organizacijų duomenis. Didelė duomenų šaltinių, skirtų elektroninio mokymosi sistemoms, įvairovė kelia didelius iššūkius skirtingų duomenų integravimui. Semantinis ir sintaksinis tarpusavio veikimas yra labai svarbus duomenų integravimui. Semantinis tarpusavio veikimas yra žinių lygio tarpusavio veikimas, kuris pagalbinei veiklai sudaro galimybę aprėpti semantinius konfliktus, kylančius dėl numanomų reikšmių, perspektyvų ir prielaidų skirtumų, tokiu būdu sukuriant semantiškai suderinamą informacinę aplinką, paremtą aptartomis idėjomis skirtinguose veiklos organizmuose [21]. Sintaksinis tarpusavio veikimas yra pritaikymo lygio tarpusavio veikimas, kuris leidžia daugiopiepiams programinės įrangos komponentams veikti kartu, nors jų įdiegimo kalbos, sąsajos įtaisai ir vykdymo programos yra skirtingos. Sintaksinis tarpusavio veikimas pateikia technologinį sprendimą, tuo tarpu, semantinis tarpusavio veikiamumas pateikia semiotinius, kalbinius, filosofinius ir socialinius sprendimus [21].

Toliau panagrinėsime kokie gali būti e. mokymosi technologijų tipai ir kaip jos grupuojamos.

1.1. Elektroninio mokymosi technologijų tipai

Nagrinėjant e. mokymosi technologijas, svarbu gerai suprasti e. mokymosi aplinkos sistemos struktūrą. Kalbant apie e. mokymosi sistemas, duomenų šaltiniai yra žymiai platesni nei duomenų bazės. Nepakanka dėmesio skirti tik duomenų bazės schemas integravimui. E. mokymosi aplinka yra žymiai lankstesnė. Ji turi remtis atitinkamais standartais (pvz. SCORM ar kt.). Turimi duomenų šaltiniai dažniausiai apima duomenų bazes, žinių bazes, tradicines bylas, tinklo puslapių bylas, elektroninio pašto pranešimus, žemėlapius ir vaizdinius, video ir garso įrašus. E. mokymosi aplinka turi palaikyti turinio pateikimo, bendravimo, bendradarbiavimo, vertinimo funkcijas ir įrankius. Daugėjant prieinamų informacijos šaltinių, funkcijų ir įrankių skaičiui, svarbi tampa problema įvairiarūšių duomenų šaltinių naudojimo ir funkcionalumo integravimų į kitas sistemas. Duomenų rinkimas ir integravimas yra esminis reikalavimas e. mokymosi aplinkoms. E. mokymosi aplinkoje vyksta visas ugdymo procesas: pateikiamas kurso ar modulio turinys, bendraujama diskusijų forumuose, pokalbiuose ir elektroniniu paštu, įgytos žinios bei gebėjimai tikrinami kompiuteriniais testais, įvertinami automatinėmis priemonėmis ir t.t. [22].

Visgi funkcionalios e. mokymosi aplinkos sukūrimas neatsiejamas nuo taikomųjų e. mokymosi technologijų.

E. mokymosi technologijos yra įvairių tipų, nelygu kaip jas grupuosime. Tai atviro kodo ir licencijuota programinė įranga, specializuotos (skirta konkrečiai e. mokymosi organizavimo funkcijai) ir bendrojo pobūdžio e. mokymosi aplinkos, skirtos tam tikram mokymosi modeliui realizuoti (pvz. asinchroninis, sinchroninis, mišrus), kursų turinio valdymo sistemos, mokymo valdymo sistemos ir pan. Programinės įrangos gamintojai e. mokymosi įrankius ir komponentus interpretuoja skirtingai, todėl kartais tik atrodo, kad sistemos yra skirtingos. Programinė įranga, skirta nuotoliniam mokymuisi, dažniausiai yra sudaryta iš kelių, mišrių, funkcinų komponentų (pavyzdžiui, viename produkte gali būti kursų kūrimo įrankiai ir mokymo valdymo sistemos elementai). Šiuo metu yra sukurta daug e. mokymosi aplinkų programinės įrangos, skirtos mokyti nuotoliniu būdu ir auditorijose.

Aplinkos gali būti skirstomos įvairiai ir tai dažniausiai priklauso nuo to, kokius skirstymo kriterijus pasirinksi. Literatūroje dažniausiai pateikiamas skirstymas pagal licencijos tipą:

- *Atviro kodo programos* dažniausiai yra nemokamos, nereikia pirkti licencijų įdiegiant ir naudojant. Kitas privalumas yra tai, kad šios aplinkos yra atviro kodo. Tai reiškia, kad mokant programuoti, galime šias programas patobulinti naujais įrankiais ar funkcijomis, pataisyti bei papildyti atrastas klaidas ir tokiu būdu pritaikyti programas savo poreikiams. Žinoma, yra nemaža įvairių klaidų tikimybė ar nesklandaus darbo. Populiariausios atviro kodo aplinkos yra Moodle, Sakai, ATutor, Fle3, Baazar ir t.t.

- *Licencijuojama programinė įranga* dažniausiai yra mokama ir kuriama komercinių organizacijų. Didžiausias trūkumas yra tai, kad įranga yra nepigi, kainuoja ne tik jų įdiegimas, bet dažniausiai kainuoja ir naudojimosi licencijos, o taip pat ir naujinimai. Privalumai būtų tai, kad tokios programų aplinkos standartizuotos, pakankamai funkcionalios, stabiliai veikiančios, paprastai naudojamos ir nesudėtingai įdiegiamos. Jos nereikalauja papildomų žinių jas eksploatuojant, nebūtinai ir informacinių technologijų žinios. Populiariausios licencijuojamos e. mokymosi aplinkos tinkančios šiam tipui yra Blackboard, LearningSpace, FirstClass ir kt.

- *Institucinė programinė įranga*. Be anksčiau minėtų programinės įrangos tipų gali būti naudojama ir programinė įranga, kurią mokymo institucijos ar organizacijos sukuria pagal savo poreikius ar adaptuoja grupinio darbo sistemas mokymosi paslaugai teikti organizacijos viduje. Dažniausiai organizacijos inicijuoja turinio valdymo sistemų kūrimą (pvz. web aplinkos ar intraneto), kur gali pageidauti matyti ir tam tikrus mokymosi aplinkos elementus darbuotojų mokymams organizuoti. Šios funkcijos gali išsiplėtoti iki e. mokymosi aplinkų. Tačiau pačios organizacijos šios programinės įrangos neplatina, o naudoja tik savo reikmėms.

Dažniausiai turinio valdymo sistemos skirtos žiniatinklio svetainėms kurti ir jų turiniams modifikuoti. Daugelis jų būna žiniatinklinės (turiniui tvarkyti pakanka naršyklės), tačiau yra ir reikalaujančių specialios klientų programinės įrangos (pvz. Lotus Notes). Turinio valdymo sistemomis galima tvarkyti svetainės turinį, nesirūpinant vidiniu jo pateikimo mechanizmu (vartojama ženklinimo kalba, scenarijais ir kt.) [13]. E. mokymosi aplinkų pagrindą ir sudaro turinio valdymo sistemos, kurios turi specialių priemonių mokymosi turiniui kurti, jam tvarkyti ir jo kūrėjams bendrauti.

Šiose sistemose dėmesys yra skiriamas nevienalaikio mokymo plėtojimui ir pristatymui. Turinio valdymo sistemos paprastai yra skirtos mokslo institucijoms, ypač aukštojo mokslo institucijoms. Šiose sistemose yra standartinės turinio kūrimo priemonės, paprastai pateikiami įvairūs vadovavimo besimokantiesiems, vertinimo ir proceso sekimo lygiai. Komercinės e. mokymosi aplinkos leidžia naudotis visomis įmanomomis mokomosios medžiagos pateikimo, bendradarbiavimo, savikontrolės, protokolavimo ir kitomis galimybėmis dėstytojų ir studentų patogumui, tačiau jų kaina yra didelė, be to, periodiškai tenka mokėti už jų atnaujinimą. Nemokamos (atvirojo kodo) e. mokymosi aplinkos daugiau dėmesio dažniausiai skiria konkrečiam e. mokymosi tikslui realizuoti.

Lietuvoje naudojamose e. mokymosi aplinkose yra sudėta tūkstančiai kursų, tačiau jų autoriai paprastai naudoja tik dalį aplinkos galimybių, dažniausiai mokomosios medžiagos valdymą ir testus [41].

E. mokymosi aplinkos tipas gali riboti integravimo procesus tiek, kiek pati aplinkos sistema yra atvira. Tačiau bet kuris e. mokymosi aplinkos tipas gali būti integruotas į kitą informacijos sistemą. Toliau pažvelgsime į kokias klases e. mokymosi technologijų skirstomos.

1.2. Elektroninio mokymosi technologijų klasifikacija

E. mokymosi technologijos pagal didžiausią Europos Žmogiškųjų išteklių valdymo ir plėtros institutą (angl. The Chartered Institute of Personnel and Development) [48], gali būti klasifikuojamos į penkias kategorijas:

1. Virtualios mokymosi aplinkos (angl. Virtual Learning Environments). Virtualios mokymosi aplinkos yra mokymo vykdymo ir valdymo platformos. Platforma šiuo atveju yra bendrinis terminas, kuris nusako programinės įrangos sistemą, įtraukiant sistemos architektūrą, operacinę sistemą ir programavimo kalbas – viskas sudaro platformą, kuri specialiai adaptuota mokymui ir mokymuisi. Virtualios mokymosi aplinkos dar vadinamos mokymosi valdymo aplinkomis, tačiau kai kurie kiti terminai, kurie yra naudojami, išreiškia daugiau jų specifškumą, kaip kad besimokančio valdymo sistema (angl. Learner Management System; sistema

manipuliuojanti informacija apie besimokančiuosius), mokymosi turinio valdymo sistema (angl. Learning Content Management System; mokymo organizavimo sistema – talpinimo saugyklos, kursų ir jų komponentų talpinimo ir naudojimo sistema) ir virtualios mokymosi klasės (angl. Virtual Classroom; priemonės mokymams rengti vartotojams, esantiems skirtingose vietose). Virtualios mokymosi aplinkos gali apimti ir kai kurias technologijas, kurios nurodomos žemiau nagrinėjamoje klasifikacijoje.

2. Kūrimo įrankiai (angl. Authoring Tools). Bent kiek įgijus programavimo įgūdžių galima sukurti nesudėtingą internetinį puslapį su įvairiu turiniu. Kūrimo įrankiai suteikia galimybę sukurti tokį patį rezultatą net ir neturint programavimo žinių. E. mokyme kūrimo įrankiai leidžia kurti vartotojui mokymosi turinį, pasinaudojant minimaliais šablonais, kur autorius gali matyti, ką jis įvedė, taip pat tai kaip tą įvestą informaciją matys besimokantysis. Kūrimo įrankiai taip pat leidžia sukurti „mokymosi objektus“, kurie gali būti kaupiami, atkuriami ir panaudojami kituose kursuose ar kuriant studijų medžiagą.

3. Bendravimo įrankiai (angl. Collaborative Tools). Bendravimo įrankiai leidžia bendrauti su kitais besimokančiais ir dirbti kartu svarstant įvairias problemas ar klausimus. Tai gali būti wiki, blog'ai ir diskusijų forumai.

4. Vertinimo įrankiai (angl. Assessment Tools). Vertinimo technologijos būna nuo paprastų apklausos įrankių iki sudėtingos klausimynų programinės įrangos, analizuojančios atsakymus. Tai objektyvių metodų apdorojimas, tokių kaip klausimai su atsakymais, teisingi – neteisingi atsakymai, „karšto taško“ klausimais (kai reikia pažymėti tam tikrą sritį diagramoje, žemėlapyje ar kt.) ir t.t. Visgi jie riboti įvertinant laisvus teksto atsakymus ar esė dėl didelio ir sudėtingo atsakymo diapazono.

5. Specializuota programinė įranga (angl. Specialist software). Ši kategorija apima visas tas technologijas, kurios neįeina į pirmąsias keturias. Tai vis atsirandančios naujos technologijos. Čia įeina simulatoriai, žaidimai, studijavimo ir peržiūrėjimo įrankiai, trimatė panoraminė vaizdo gavimo programinė įranga, virtualūs pakaitalai ir kitos technologijos, kurios padeda studijuoti.

Šios penkios bendrinės e. mokymosi technologijos apima visus e. mokymosi procesuose naudojamus įrankius ir komponentus.

E. mokymosi aplinką sudaro duomenys arba mokymosi medžiaga. Mokymosi medžiagai galima priskirti spausdintinę, garsinę, vaizdinę ir elektroninę informaciją. Tačiau šiandieninėse nuotolinėse studijose beveik visa naudojama mokymosi medžiaga yra elektroninė, o pats nuotolinio mokymosi procesas dažniausia realizuojamas informacinių technologijų, o dar tiksliau - internetinių technologijų (žiniatinklio, internetinių pokalbių, garso ir vaizdo konferencijų) pagalba. Visa elektroninė informacija perduodama kompiuterinėmis ryšio priemonėmis ir panaudojama mokyme.

Bet kurią elektroninę informaciją galima panaudoti organizuojant mokymą, skirsis tik tos medžiagos panaudojimo efektyvumas ir pritaikomumas.

Kuriant ir aprašant e. mokymosi modulius atsiranda būtinybė derinti juos su egzistuojančiais standartais. Egzistuoja eilė projektų ir standartų, kurių pagalba yra aprašomi e. mokymo ir mokymosi kursai (IMS, SCORM, LOM, CourseML ir kt.), kurie įtakoja e. mokymosi aplinkas. Kiekvieno skirtingo standarto ar kitokio elektroninio mokymo kurso kūrimo projekto taisyklių taikymas sunkina elektroninės informacijos sklaidą, nes jie gali netikti vienai ar kitai elektroninio mokymo terpei. Tačiau skirtingų standartų naudojimas sunkina e. mokymosi aplinkų integraciją su kitomis informacijos sistemomis. Be to, mokymo turiniui rengti naudojamos įvairios priemonės, padedančios kurti ir rengti elektroninę mokymo medžiagą konkrečiai elektroninio mokymo aplinkai. Priemonės pasirinkimas priklauso ne tik nuo mokymosi dalyvių poreikių, tačiau ir nuo elektroninės mokymo aplinkos, jos galimybių integruoti tam tikrus duomenų formatus.

Informacinės technologijos, naudojamos nuotoliniam mokymuisi, gali būti daugiau ar mažiau specializuotos, tenkinančios vienos ar kitos srities poreikius. Vis dėlto labiausiai paplitęs e. mokymosi programinės įrangos klasifikavimas pagal jos bendrą funkcionalumą. Nuo funkcionalumo priklauso e. mokymosi teikimo ir valdymo galimybės. Pagal D. Belyk, J.Schubert ir J.Baggaley [2] galima skirti tokias programinės įrangos grupes:

- Asinchroninės kursų valdymo sistemos;
- Synchroninės sistemos;
- Mokymo valdymo sistemos;
- Kitos specializuotos sistemos. Apima dvi autorių nurodytas grupes – bendrinių produktų (angl. Total solutions) ir specializuotus įrankius (angl. Related Tools).

Asinchroninės sistemos. Asinchroninės kursų valdymo sistemos (angl. course management systems) - tai tarnybinių stočių programinė įranga, skirta sukurti, pateikti ir valdyti asinchroninius mokymosi procesus. Šiose sistemose realizuojami pasirinkti mokymosi metodai, ir dažniausiai jos turi vartotojų aplinkos ir mokymosi priemonių apribojimų. Tačiau kursų kūrėjai, kuratoriai turi turėti minimalius techninius ir meninius gebėjimus, kurių prireikia norint realizuoti kurso medžiagą.

Tai didžiausia virtualių mokymosi terpių grupė, kuri pirmiausia ir buvo taikyta mokymo įstaigoms. Šios grupei priskiriama programinė įranga: Learning Manager, LearningSpace, TopClass, Desire2Learn.

Synchroninės sistemos. Synchroninės - tai tarnybinių stočių programinė įranga, skirta organizuoti vaizdo ir garso konferencijas, bendravimą realiuoju laiku. Dažniausiai jos naudojamos informacijai pateikti, konsultacijoms, rečiau - bendradarbiauti.

Šios grupės programinė įranga: LaunchForce, LearnLinc, LearningSpace – Virtual Classroom, eLecta Live.

Mokymo valdymo sistemos (angl. Learning management systems). Tai programinė įranga, skirta valdyti nuotolinio mokymosi procesus didelėmis apimtimis. Dažniausiai mokymosi moduliai kuriami naudojant kitą programinę įrangą. Šios sistemos turi labai dideles informacijos pateikimo ir valdymo galimybes. Jos gali būti akivaizdžiai brangesnės nei asinchroninės sistemos, nes yra skirtos didelėms kompanijoms ar didelį finansavimą turinčioms mokymo įstaigoms. Šios grupės programinė įranga: IBM Learning Management System, Blackboard, Generation 21, Saba Learning Enterprise, WBT Manager.

Kita įranga. Programinė įranga, skirta internetiniams kursams kurti, bendradarbiauti. Tai testų kūrimo ir valdymo sistemos. Čia taikomi skirtingi įrankiai, kurių įdiegimas ar teisingas naudojimas gali reikalauti ir programavimo žinių. Kaip pavyzdžiai tai programiniai paketai Authorware, First Class, Pathware, PlaceWare, Questionmark, Trainersoft, WebBoard, TestTool ir kt.

Programinė įranga, patenkanti į kurią nors iš šių grupių, atlieka tam tikras funkcijas arba pateikia įrankius tam, kad galima būtų įgyvendinti tam tikrus tikslus. Šie įrankiai nebūtinai priklauso tik vienai programinės įrangos grupei, jie gali priklausyti kelioms grupėms iš karto. Kartais sunku vienareikšmiškai priskirti vieną ar kitą sistemą tam tikrai grupei, todėl nėra griežtų ribų tarp šių grupių. Sistema priskiriama tai grupei, kurios funkcijų atlieka daugiausiai.

1.3. Elektroninio mokymosi aplinkų sistemų architektūros

E. mokymas gali būti realizuojamas trimis pagrindinėmis informacijos sistemų architektūros formomis [14]: asinchronine, sinchronine ir mišria (angl. blended learning). Jos skiriasi ne tik funkciniu, bet ir technologiniu požiūriais.

1.3.1. Asinchroninio tipo sistemų architektūra

Terminas *asinchroninis* apibrėžiamas kaip vykstantis savu laiko ritmu, nederinamu su kitais tuo pat metu vykstančiais susijusiais procesais [13].

Pagrindiniai asinchroninės architektūros privalumai yra šie [33]:

- Spartesnis reagavimas iš sąsajos procesų (paprastai iš internetinio puslapio), kuris suvokiamas kaip greitesnė užsakovo sistema;
- Lengviau atliekamas apkrovos subalansavimas (angl. load balance);

- Sistemos gedimų prevencija;
- Sujungtų sistemų trikdžių prevencija.

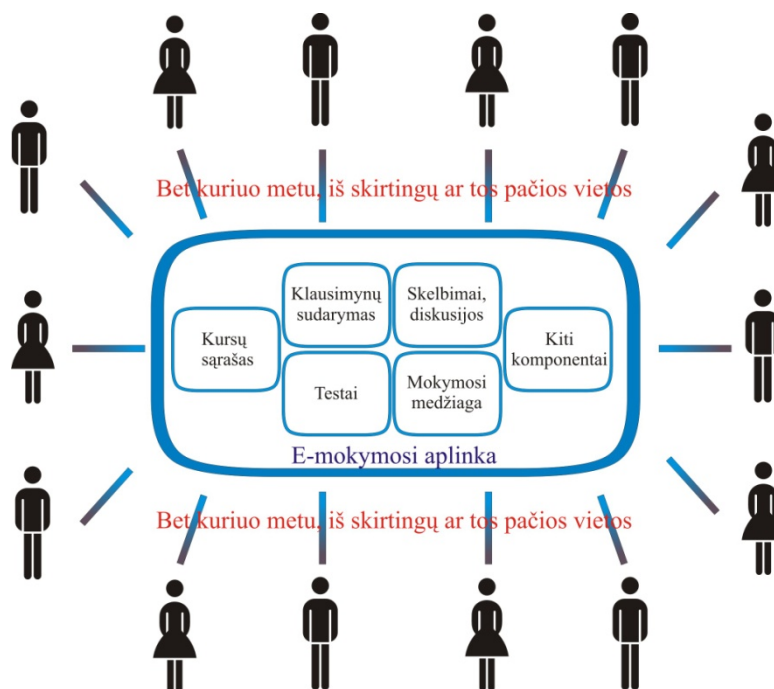
Kiekvienas iš aukščiau minėtų privalumų yra asinchroninio modelio rezultatas naudojant taikomas programas. Procesas laikomas asinchronišku, kai suformuojama tam tikra veiksmų seka, kurią sudaro sprendimų laukiančios užklauskos. Užklauskos siunčiamos vykdymui naudojant tarpines sekas vietoje tiesioginio kreipimosi į ankstesnę ar kitą proceso žingsnį.

Vienas esminių asinchroninio tipo sistemos privalumų yra greitesnis reagavimo laikas, t.y. kai vartotojui naudojantis tam tikra kompiuterine sistema, nereikia laukti, kol daugybė procesų nuorodų bus įvykdytos. Asinchroninėje sistemoje, gavus atsakymą, kad užduotis įvykdyta, laiko užtrunkama tiek, kiek užtrunkama nusiųsti nurodymus kitai proceso užduočiai.

Asinchroninėje sistemoje užklauskų seka reikalinga daugybei tarpinių užklauskų žingsnių saugoti. Kai baigiamas vienas tarpinis žingsnis, pereinama prie kito. Kai kitas tarpinis žingsnis yra pasirengęs kitai užklausiai, jis ima iš sąrašo kitas užklauskas. Apkrovos balansavimą tokiose sistemose galima pasiekti didinant sistemos užklauskų apdorojimo galimybes kiekviename tarpiniame žingsnyje.

Asinchroninės architektūros dėka sistema gali veikti be trikdžių, prognozuojant procesų sutrikimus, kurie gali pasitaikyti, atsiradus sistemos klaidoms. Sistemos užklauskų apdorojimų galimybių didinimas kiekviename tarpiniame žingsnyje leidžia lanksčiai valdyti sistemos apkrovos balansavimą, taip pat veikti sistemai be trikdžių. Jei techninės ar programinės įrangos gedimas pašalinamas viename proceso etape, laukianti užklausa šiam etapui bus tik pastatyta į eilę, kol paslauga bus atstatyta. Tai neturės jokios realios įtakos ankstesniam proceso etapui, net jei bendra proceso trukmė greičiausiai padidės dėl trukdžio. Jei prieš tai minėtas sistemos apkrovos balansavimas bus realizuotas, tikėtina, kad procesas šiame etape paprasčiausiai bus lėtesnis, tačiau nesustabdytas. Ta pati funkcija gali būti naudojama taikant užklauskų sugrupavimą, sancaupų formavimą (angl. clustering).

E. mokymas, paremtas asinchronine sistemos architektūra, yra tokia studijų forma, kai besimokantieji studijuoja teikiamą medžiagą bet koku pavidalu (e. moduliai, e. kursai, nuorodos į medžiagą internete, CD, konspektai, knygos ir kt.), bet kuriuo jiems patogiu laiku. Studentai turi galimybę bendradarbiauti tarpusavyje bei su kuratoriumi, tačiau į jų užklauskas greičiausiai bus atsakyta ne tuoj pat, o tik tada, kai kolega ar kuratorius prisijungs prie kurso. Tad šioje sistemoje trūksta tik tiesioginio bendravimo tarp vartotojų.



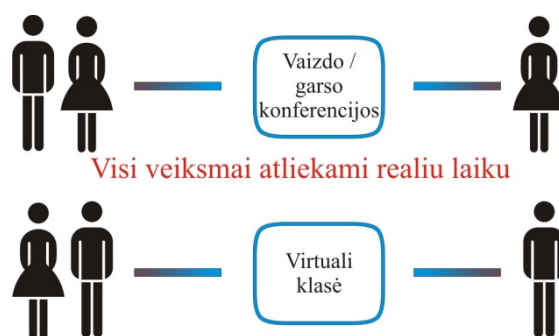
1 paveikslas. Asinchroninio tipo sistema funkcinio požiūriu

1.3.2. Sinchroninio tipo sistemų architektūra

Terminas *sinchroninis* apibrėžiamas kaip vykstantis tuo pačiu laiko ritmu, kaip ir kiti tuo pat metu vykstantys susiję procesai [13]. Sinchroninėse sistemose vartotojas gauna atsakymą, kai visos operacijos (pavyzdžiui, pateikiant užsakymą) yra įvykdytos.

Sinchroninės bendravimo priemonės yra tekstiniai pokalbiai, pokalbiai balsu, vaizdo ryšys. Sinchroninės bendradarbiavimo priemonės: pateiktys, naršymas drauge, bendras programos lango valdymas, bendros diagramų kūrimo priemonės, žaidimai tinkle.

E. mokymas, vykdomas sinchroninės sistemos architektūros pagrindu, grįstas garso ir / arba vaizdo konferencijų rengimu ir dalyvavimu jose. Tokiu atveju besimokantieji turi sutartu laiku būti ten, kur yra priimami signalai iš studijos ar dėstytojo darbo vietos, iš kurios jis skaito paskaitą. Dažniausiai besimokantieji turi ryšį su dėstytoju ir gali su juo bendrauti realiu laiku.



2 paveikslas. Sinchroninio tipo sistema funkcinio požiūriu

Sinchroninis mokymasis gali būti realizuojamas keliais būdais [41]:

- Vaizdo/garso konferencijos, kurios gali būti transliuojamos iš specialios video konferencijų studijos, naudojant profesionalią techniką, gali būti specialiai įrengta studija, nors dabartinės technologijos leidžia tai atlikti bet kurioje tinkamoje vietoje. Šios vaizdo – garso konferencijos pagal sujungimo pobūdį skirstomos į kelias grupes: 1) *taškas – taškas* vaizdo konferencija, kurioje dalyvauja du nutolę taškai; 2) *daugiataškės* vaizdo konferencijos, kurios sujungia tris ir daugiau vietų. Jų metu gali būti naudojama specializuota techninė ir programinė įranga; 3) *klasės – auditorijos* vaizdo konferencijų sistemos tipas leidžia bendrauti įvairaus dydžio žmonių grupėms. Kokybiškai transliacijai reiktų profesionalios vaizdo konferencijų įrangos tiek transliuojančioje pusėje, tiek priimančioje signalą.

- Darbo vietos vaizdo konferencijos tinka individualiems ar nedidelių grupių kontaktams, žmonės gali dalyvauti vaizdo konferencijoje prie kompiuterių savo darbo vietose. Tokiu atveju vaizdo konferencijos gali vykti su minimalia vaizdo konferencijų įranga (pvz. internetinė kamera, ausinės su mikrofonu ar kompiuteryje įmontuotu mikrofonu). Internetinė kamera, pritvirtinta prie kompiuterio monitoriaus arba įmontuota į patį kompiuterį, filmuoja darbo vietą šalia kompiuterio, o įmontuotas mikrofonas perduoda balsą.

1.3.3. Mišraus tipo sistemų architektūra

Mišrus mokymasis, dar vadinamas hibridiniu arba įvairiarūšiu mokymusi – tai švietimo srityje tvirtai susiformavęs mokymosi metodas. Nuotolinių studijų organizavime tai pakankamai nauja forma – mišri, kuri apjungia asinchroninį ir sinchroninį mokymosi modelius. Techniniu ir technologiniu požiūriu ji pakankamai sudėtinga, tačiau edukologiniu požiūriu yra labai efektyvi [34]. Pagrindinė mintis – tai pasirinkimas. Mišraus mokymo esmė - galimybė pasirinkti patogiausius metodus pagal nustatytus mokymo poreikius.

Mišraus mokymo forma gali būti dvejopa [24; 25]:

- mišri forma apibūdinama kaip asinchroninių e. kursų teikimas ir akivaizdiniai susitikimai „akis į akį“ konsultacijoms;
- mokomosios medžiagos teikimas asinchroninių e. kursų pavidalu ir paskaitų skaitymas bei konsultacijos audio/video konferencijų (sinchronine) forma.

Mišraus mokymo komponavimas ir mokymosi veiklos seka, susiję su:

- įvairius teikimo būdus: realiu laiku, nerealiu laiku, sinchroninis, asinchroninis;
- įvairiomis mokymosi architektūromis: juslinis, nurodomasis, valdomas atrandant ir tiriamasis mokymas;

- įvairiais mokymo metodai, ir daugialypiškumu, kurių pagalba parenkamas mokymosi stilius;

- įvairių socialinio mokymosi dinamika, kurios turi įtakos motyvacijai.

Mišrų mokymosi metodą galima realizuoti apjungiant mokymosi valdymo sistemą (arba kursų valdymo sistemą) ir virtualią klasę. Svarbu, kad būtų palaikomas tas pats standartas (pvz. SCORM). Mišraus tipo architektūra yra komponentinė, kurios pagrindą sudaro mokymosi valdymo sistemos komponentės, o šių komponentų darni veikla, leidžia išspręsti įvairias e. mokymosi organizavimo ir valdymo problemas. Tokios sistemos funkcionalumas paskirstomas tarp daugelio aplikacijų.

Mišraus tipo sistemų pagrindinės savybės būtų šios [25]:

- multi platforminė architektūra, kuri prisitaiko prie besikeičiančių poreikių ir, pabrėžtina, neprisiria prie konkrečių technologijų;

- suteikia galimybę mokytis neprisijungus prie tinklo;

- bendravimo ir bendradarbiavimo priemonių laisvas pasirinkimas.

Mišraus tipo sistemų architektūros savybės:

- nėra geografinių barjerų, sistema pasiekama iš bet kur ir bet kur;

- kadangi nereikalingas stacionarus mokymosi inventorių tai mokymas yra dinamiškesnis resursų atžvilgiu;

- greitis. Greitesnė reakcija į poreikius, galima naudoti šablonus kursų kūrimui kas spartina jų sukūrimą, greitesnė organizacija.

Iš trijų minėtų sistemų architektūros tipų mums šiame darbe aktualiausia yra asinchroninio tipo sistema. Pagrindinė ir populiariausios e. mokymosi aplinkoms realizuoti yra naudojama būtent asinchroninio tipo architektūra.

1.4. Elektroninio mokymosi aplinkų informacinės technologijos

Renkantis e. mokymosi aplinką sudėtinga būtų jas įsidiesti ir išbandyti, todėl priimtinesnis būdas jas palyginti yra pasinaudoti jau esančiais įrankiais. E. mokymosi aplinkų palyginimo darbų nėra daug, dažniausiai jie yra uždari ir skirti tik tai organizacijai arba organizacijų nariams. Sunku rasti ir įrankių palyginimo pavyzdžių, tačiau vienas įrankis *Edutools*, leidžia palyginti skirtingas e. mokymosi aplinkas per jų savybes. Šioje dalyje bus akcentuojama technologinė e. mokymosi aplinkų dalis.

Edutools priemonę sukūrė C2T2 (angl. The Centre for Curriculum, Transfer & Technology) ir mokslininkas Bruce Landon iš Douglas koledžo Britų Kolumbijoje, Kanadoje [50]. Edutools šiuo metu priklauso WCET (angl. the Western Cooperative for Educational Telecommunications) ir yra

sprendimų priėmimo pagalbinis įrankis mokslo institucijoms. Edutools apsiriboja produktais, skirtais aukštojo mokslo nuotoliniam mokymui.

Šiuo metu Edutools viena populiariausių prieinamų dialoginė priemonė, skirta e. mokymosi ir bendro mokymo taikymo sričių peržiūrai ir analizei.

Lietuvoje šiuo metu plačiausiai naudojamos e. mokymosi aplinkos yra turinio valdymo sistemų pagrindu sukurtos asinchroninės sistemos, tokios kaip Moodle (naudoja beveik visi universitetai ir kolegijos), Sakai (Šiaulių universitetas). Prieš keletą metų aktyviai buvo naudojama Blackboard e. mokymosi aplinka, tačiau šiuo metu akademinė visuomenė atsisakė jos dėl finansinių aspektų, bet apžvelgiant e. mokymosi aplinkas darbe remsimės ja kaip pavyzdine komercine e. mokymosi aplinka. Lietuvoje naudojamos sinchroninės sistemos, tokios kaip LearningSpace ir FirstClass. Tačiau šiame darbe analizuosime plačiausiai naudojamas e. mokymosi aplinkas, sukurtas turinio valdymo sistemų pagrindu. Analizuojant ir lyginant labiau kreipsime dėmesį į sistemų architektūrinius ir technologinius aspektus.

Analizuosime e. mokymosi aplinkas pagal programinės įrangos pavadinimą. Lyginsime populiariausias, turinio valdymo sistemų pagrindu kurtas, asinchronines e. mokymosi aplinkas, kurios Lietuvoje geriausiai žinomos arba naudojamos: The Blackboard Academic Suite (Release 8.0), Sakai 2.3 ir Moodle 1.9. Tačiau atliekant analizę, labiau akcentuosime tuos komponentus, kurie yra aktualūs vykdant e. mokymosi aplinkos integracijos procesus į kitas informacijos sistemas.

1.4.1. Studentų veiklos įrankiai

Studentų veiklos įrankiai apima tokius komponentus kaip darbą grupėse, studento aplanką ir bendrą tinklą. E. mokymosi aplinkos leidžia kurti tam tikro dydžio atsitiktinio dydžio grupes, kuriose dažniausiai leidžiama turėti savo vidinius diskusijų forumus, pokalbių lentas ar lentas. Grupės gali turėti savo išteklius, bendrus failus, o Sakai leidžia kurti ir atskirą projekto erdvę. Kiekvienai grupei gali būti skiriama konkreti užduotis ar veikla. Grupės galima stebėti atskirai. Studentai gali kurti virtualius klubus, grupes pagal pomėgius ir studijas, tokiu būdu organizuojant atskirus bendrus tinklus. Studentai iš įvairių kursų gali bendrauti su visais pokalbių kambariuose ar diskusijų forumuose.

Blackboard aplinka leidžia studentams kurti asmeninį pagrindinį puslapį kiekvienam kursui, kuriame gali kaupti visus kurso darbus. Be to, studentas gali dalintis savo aplanku su kitais vartotojais. Sakai e. mokymosi aplinkoje yra naudojamas OSP (angl. Open Source Portfolio), kuris yra laikinasis įrankis, turintis vartotojo aplanko funkcionalumą. Moodle aplinkoje vartotojas savo

profilyje per „blog“ gali kaupti kurso darbus, juos viešinti ir atlikti kitus veiksmus. Privati erdvė apsaugo studentą nuo dokumentų praradimų, kurie gali būti reikalingi vėlesniais laikotarpiais.

1.4.2. Administravimo įrankiai

Analizuojant e. mokymosi aplinkas svarbūs yra administravimo įrankiai, kurie nusako kaip vartotojas patenka į e. mokymosi sistemą.

Autentifikavimo procesas tai tapatybės nustatymas. Autentifikavimas yra svarbus norint įsitikinti, kad tai tikrai tas vartotojas yra prisijungęs. Vartotojui suteikiamos prieigos, kur gali būti apibrėžta kaip jis gali naudotis e. mokymosi aplinka. Autentifikavimas leidžia išvengti nesusipratimų atsiskaitymų laikotarpyje, be to apsaugo vartotojų profilius ir privačias erdves nuo praradimų ar neleistinių kitų veiksmų. Administratoriai gali leisti svečio prieigą prie visų kursų. Moduliai gali būti viešai prieinami arba sukurta apsaugota prieiga prie atskirų kursų su vartotojo vardu ir slaptažodžiu.

E. mokymosi aplinkos leidžia atlikti tapatybės tikrinimą per išorinę LDAP tarnybą – vartotojų duomenų saugyklą. Autentifikavimo procesui galima atlikti pasinaudojant: Kerberos saugos protokolu, Shibboleth standartu tapatybės nustatymui, centrine atpažinimo sistema CAS (angl. Central Authentication Service). Vartotojo sąsaja su tarnybine stotimi gali būti apsaugota koduojant ryšio kanalą SSL protokolu. Slaptažodžiai dažniausiai koduojami sistemos duomenų bazėje.

E. mokymosi aplinka suderinama su daugialypiais organizaciniais ir virtualiais vienetais pagal serveryje esančią konfigūraciją. Galimas autentifikavimas iš antrinio šaltinio kai nepavyksta iš pirminio.

Toliau pateiksime kitas Blackboard, Sakai ir Moodle e. mokymosi aplinkų administravimo įrankių galimybių reikšmes pagal atskiras savybes.

1 lentelė. Kitos administravimo įrankių savybės

Programinė įranga Savybė	Blackboard	Sakai	Moodle
Autentifikavimas	Galimas tapatybės nustatymas per Active Directory, Microsoft.NET	Tapatybės nustatymas galimas asinaudojant IMAP, POP3 ir saugų NNTP protokolus.	Galimas tapatybės nustatymas per Active Directory, dirba su LDAP
Kursų autorizacija	Prieiga pagal nustatytas vartotojų teises. Teisės pagal paslaugų teikėjo nustatymus. Administratoriai gali sukurti neribotą skaičių organizacinių		

	vienetų ir vartotojų su užsakytomis specifinėmis priemonėmis prie kursų ir įrankių, priskirti prieigų teises pagal organizacijos vienetus ar skyrius aprašytus serveryje. Kuratoriams ar studentams galima paskirti skirtingus vaidmenis skirtinguose kursuose.		
Registracijos integracija	<p>Administratoriai gali perduoti studentui informaciją dvikrypčiu būdu per sistemą. Programinė įranga palaiko duomenų apsikeitimą su studentų informacijos sistemomis per įvykiais pagrįstą sąsają. Programinė įranga palaiko integraciją su SCT Baner, SCT Luminis, Datatel, PeopleSoft 8 ar individualiomis integracijomis su kitomis IS. Suderinama su studentų duomenų IMS Enterprise specifikacija. Sistema palaiko SOAP pagrįstų duomenų integraciją.</p>	<p>Sąrašo įrankis (Roster tools): aprūpina visų vartotojų sąrašą, sąsajoje su atskiro vartotojo profiliu. Sąrašo įrankis dar naudoja naują programavimo sąsają privatumo valdymui. Profilio įrankis: profilio įrankis yra vieta, kur vartotojas gali tvarkyti savo asmeninius duomenis sprendžiant ką viešinti, o ko ne. Galima informacijos sinchronizacija su organizacijoje esančiu asmens profiliu. Profilio įrankis gali teikti informaciją apie bet kokią vartotoją sistemoje.</p>	<p>Administratoriai gali perduoti studentui informaciją dvikrypčiu būdu per sistemą.</p>
	<p>Kuratoriai studentus prideda / registruoja į kursus rankiniu būdu arba leidžia jiems registruotis patiems. Administratoriai gali pridėti studentus į sistemą naudojant suformuotą teksto rinkmeną. Vartotojas gali nurodyti kuri informacija yra vieša, kuri privati.</p>		
Prieglaudos paslaugos	<p>Produkto tiekėjas siūlo prieglaudos sprendimą. Klaidų kontrolė, priežiūra visa parą visus metus, 10-20 Gb talpa, atsarginių kopijų darymas įvykiu sekimas ir signalizavimas apie problemas ir kt.</p>	<p>Svečių prieglaudą ir paslaugų tarnybas Sakai ima iš Sakai komercinių padalinių arba iš kitų organizacijų.</p>	<p>Svečių prieglauda, paslaugų tarnybos, kurios imamos iš komer. organizacijų. Visas prieglaudų paslaugas galima naudoti, kurias teikia Moodle partneriai.</p>

Kursų autorizacija leidžia kontroliuoti kursu besinaudojančius vartotojus, leidžia apsaugoti autorines kurso teises nuo neteisėtų veiksmų. Prieigos kontrolė leidžia sekti studentų aktyvumą ir stebėti jų veiksmus.

Registracijos integracijos lyginimas mums rodo, kad aplinkos turi galimybių integruotis į informacines sistemas. Integracija su studentų informacinėmis sistemomis gali būti realizuojama per esamus aplinkų papildinius arba per paslaugų teikėjų registravimo programas. Informacija tarp turinio valdymo sistemos ir institucinių informacinių sistemų yra suderinama per IMS organizacijų specifikaciją taikomą studentų duomenims.

1.4.3. Žinių vertinimo įrankiai

Testavimo įrankiais galima kurti kontrolės ir savikontrolės užduotis, testus, įvertinti juos. Žinių vertinimo įrankiai turi daug galimybių, organizuojant testus ir jų pagalba valdant, gauti atgalinį ryšį. Kursų kuratoriai turi plačias galimybes valdyti kursą, sudėliojant jų studijavimo sekas, ar medžiagos sekas, pagrįstą datomis. Taip pat kuratoriai valdo vertinimus, pranešimus, įvykių datas ir kt. Dauguma e. mokymosi aplinkų turi testavimo įrankius, be to, gali automatiškai sudaryti statistines ataskaitas apie studentų testavimo rezultatus. Tačiau viena aplinka nesuteikia galimybės parinkti tik tam tikram studentui pritaikytus egzamino klausimus.

E. mokymosi aplinka suteikia galimybę stebėti studentų veiklą ir tam teikia papildomas priemones, kurių pagalba galima gauti pranešimus apie studentų darbo laiką ir datas, jungiantis prie kurso turinio. Kuratorius gali matyti kiekvieno studento įrašą ir sekti jo eigą. Vartojimo statistika gali būti sujungta per kursus ar per organizaciją.

Visose analizuojamose e-mokymosi aplinkose sukurta užduotis automatiškai sukuria skiltį vertinimų lentelėje. Kuratorius gali nustatyti vertinimo skales ir vertinimo žymes, detalizuoti vertinimo skiltis. Vertinimų lentelės gali būti eksportuojamos į išorines elektronines lenteles.

Vienas svarbiausių duomenų integracijos proceso tikslų yra galimybė perkelti vertinimo rezultatus į kitą informacijos sistemą.

Matome, kad e. mokymosi aplinkos turi galimybę eksportuoti vertinimus į išorines bazes. Išlieka tik suderinamumo klausimas dėl to: kada, kokių formatu ir kurie duomenys turėtų būti perkelti.

1.4.4. Turinio kūrimo įrankiai

E. mokymosi aplinkoje turinio kūrimo įrankiai turi plačias galimybes kurti, naudoti ir redaguoti e. mokymosi turinį, dalintis juo per objektų saugyklas, kurios priskirtos sistemos arba

konkreto organizacinio vieneto, visiems e. mokymosi aplinkos vartotojams. E. mokymosi aplinkos programinė įranga turi galimybę suformuoti šabloną naujiems kursams kurti. Pats kurso turinys gali būti teikiamas naudojant WebDAV protokolą (angl. Web-based Distributed Authoring and Versioning). Pabrėžtina, kad dirbant su turinio kūrimo įrankiais galima naudotis plačiomis turinio vizualizacijos galimybėmis, be to, kuratoriai gali sukurti linijines studijavimo sekas, organizuoti hierarchinius kursus, pamokas ir temas. Visose e. mokymosi aplinkose kursų kūrėjai gali naudoti daugialypės terpės elementus, tokius kaip Flash technologija ir kt.

Planuojant integraciją svarbu žinoti, kokiais formatais aprašyti duomenys. E. mokymosi aplinkų atveju dažniausiai naudojamasi metaduomenų ar duomenų aprašų standartais. Lentelėje x išvardinti skirtingų e. mokymosi aplinkų palaikomi standartai.

2 lentelė. Turinio kūrimo įrankiai

Programinė įranga / Savybė	Blackboard	Sakai	Moodle
Standartų palaikymas	SCORM 1.2; SCORM 1.3; IMS standartai; Metaduomenų žodynas v1.2.1, IMS Content Packaging 1.1.2, IMS Enterprise Specification 1.01. Sistema turi įrankius, kurie leidžia turinio perkėlimą tarp skirtingų programinės įrangos versijų.	IMS Content Packaging 1.1.4 IMS QTI 1.2.1 SCORM 1.2 IMS Common Cartridge - Sakai palaiko importavimą iš Common Cartridge medžiagos. Melete Lesson Builder palaiko IMS CP.	AICC; IMS Content Packaging 1.1.3; IMS Content Packaging 1.1.4; IMS QTI 1.2.1; IMS Enterprise 1.1; SCORM 1.2; SCORM 1.3
Pagalba neįgaliesiems	Programinė įranga atitinka WAI WCAG 1.0 lygio nurodymus ir kitus aktus.		

E. mokymosi standartai šiuo metu labai aktuali ir svarbi e. mokymosi aplinkų sritis. Mokymo valdymo sistemų atsiradimų ir galimybių į ją importuoti įvairiose aplinkose sukurtus elektroninius kursus, paskatino būtent e. mokymosi standartų kūrimas, jų pripažinimas ir standartų palaikymas, kuriant e. mokymosi aplinkas. Vieningo standarto buvimas leistų lengviau migruoti mokymosi medžiagai tarp skirtingų aplinkų, o ir integravimo procesas supaprastėtų.

Pagalba neįgaliesiems ypač aktuali, suteikiant galimybę naudotis informacijos sistemomis visiems bendruomenės nariams, ypač turintiems specialiųjų poreikių, tačiau šis aspektas kuriant sistemas dažnai pamiršamas.

1.4.5. Programiniai paketai

3 lentelė. Techninė ir programinė įranga

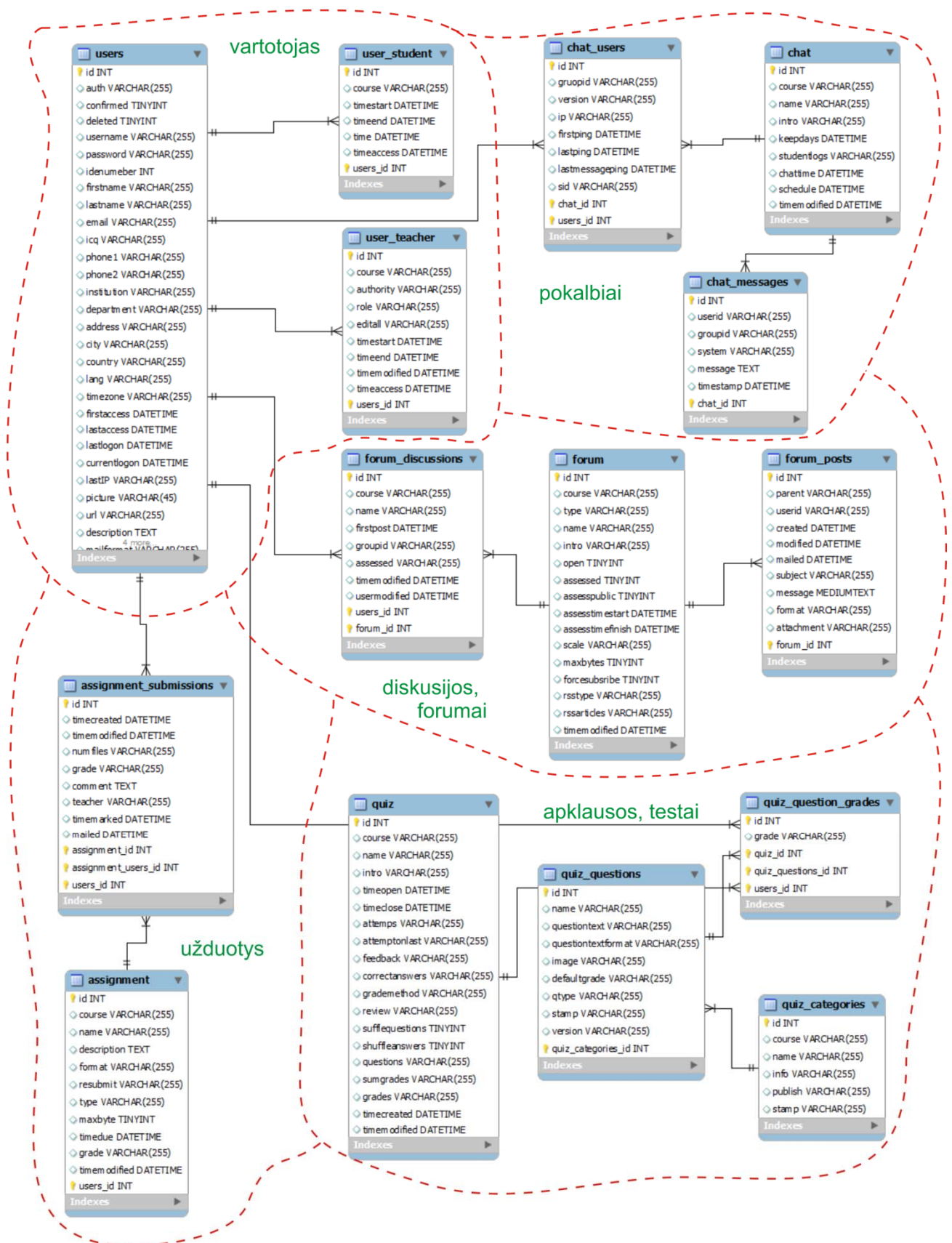
Programinė įranga / Savybė	Blackboard	Sakai	Moodle
Naršyklė	Suderinama su visomis naršyklėmis, tačiau rekomenduojama IE.	Suderinama su visomis naršyklėmis, tačiau tam tikros funkcijos nedirbs gerai ar visai nedirbs su Safari ir IE Mac. Turi būti įdiegtas Javascript įskiepis.	Suderinama su visomis naršyklėmis.
Duomenų bazė	Sistema suderinama su Oracle, MS SQL Server, MySQL	Sistema suderinama su Oracle, MySQL. Aplinkai reikalinga tik viena duomenų bazės sistema, tačiau ji palaiko kitų sistemų darbštalius.	Sistema suderinama su Oracle, MS SQL Server, MySQL, PostGreSQL.
UNIX serveris	Suderinama su visomis versijomis		
Windows serveris	Suderinama su visomis versijomis		
Kaina / licencijavimas	Metinis licencijos mokestis įstaigai arba konsorciui. Gali būti taikomas metinis mokestis pagal vartotojų skaičių.	Sakai yra nemokamas produktas. Licencijavimas pagal Educational Community License.	Moodle yra nemokamas produktas. Licencijavimas pagal GPL licenciją.
Atviras kodas		Sukurta pagal atviro kodo licencijas	Sukurta pagal atviro kodo licencijas

Visos aplinkos turi platų diapazoną trečių šalių programinės įrangos kūrėjų modulių, kurie skirtingai gali išplėsti e. mokymosi aplinkos funkcionalumą.

Atlikus analizę galima tvirtinti, kad visos nagrinėtos asinchroninės e. mokymosi aplinkos, kursų valdymo sistemos (Blackboard, Sakai ir Moodle), funkcinio požiūriu yra beveik lygiavertės. Todėl kalbant apie integracijos procesus galima pasinaudoti viena iš jų kaip pavyzdine.

E. mokymosi aplinkos sistemos duomenų struktūrą bei ryšius tarp duomenų patogiau vaizduoti esybių ryšių diagrama. Tai statinis sistemos modelis. Esybių ryšių diagramai (3 paveikslas) realizuoti pasinaudosime e. mokymosi aplinka Moodle, kuri bus naudojama kaip tipinis pavyzdys sistemų integracijai. Analizuojant e. mokymosi aplinkos dalykinę sritį, esybių ryšių

diagrama vaizduoja logines dalykinės srities informacijos grupes bei ryšius tarp šių grupių. Diagramoje vizualizuojami ne visi sistemos komponentai bei ryšiai, tačiau pagrindiniai ir esminiai, kurie gali sąveikauti su kitomis sistemomis arba gali jas papildyti. Vizualizuojant naudotasi Moodle sistemos duomenų lentelėmis bei jų statiniais ryšiais. Raudona punktyrinė linija nurodo priklausomumą komponentų blokui. Remiantis dalykinės srities modeliais, vėliau gali būti braižomos detalios esybių ryšių diagramos, aprašančios fizinę sistemos duomenų bazės struktūrą. Tačiau tai nėra šio darbo uždavinys.



3 paveikslas. Modifikuota Moodle esybių ryšių diagrama

2. AKADEMINĖS INFORMACIJOS SISTEMOS

Akademinės informacijos sistemos Lietuvoje naudojamos jau daugiau nei dešimtmetį. Jų kūrimo ir diegimo etapai buvo labai ilgi, pačias sistemas sudaro daug skirtingų modulių, dažniausiai projektuotų ir rašytų skirtingų kūrėjų skirtingais laikotarpiais. Todėl kartais jų funkcionalumas nėra tenkinantis vartotojų poreikius, o kai kurių funkcijų, ypač analitinių, vartotojai neturi visai. Tiesa, jas vadinti informacijos sistemomis, kurios visapusiškai tenkintų akademinės institucijos poreikius, yra gana sudėtinga. Šioje srityje nėra specializuotų programinės įrangos sprendimų, todėl sistemas tenka projektuoti ir kurti pačioms institucijoms.

Nuo 2001 m. buvo vykdoma programos “Informacinės technologijos mokslui ir studijoms” paprogramė “Lietuvos mokslo ir studijų informacijos sistema (LieMSIS)”. Jos tikslai buvo – sukurti mokslo ir studijų informacijos sistemą, skirtą padidinti studentų, pedagogų, mokslininkų bei administratorių darbo efektyvumą [30]. Vienas iš uždavinių buvo sukurti akademinę informacinę sistemą, tinkančią visiems Lietuvos universitetams ir kolegijoms. Tačiau 2011 metais šis projektas buvo pripažintas kaip nepavykęs ir jo vykdymas nutrauktas nesukūrus nieko tinkamo.

Šiuo metu Lietuvoje dauguma aukštųjų mokyklų naudoja akademinės informacijos sistemas. Smulkesnę informaciją apie informacijos sistemas gauti pakankamai sudėtinga, nes informacijos sistemų valdytojai tokios informacijos neviešina. Aukštųjų mokyklų akademinės informacijos sistemas bendrai bruožais aptarsime žemiau. Kitos Lietuvos aukštosios mokyklos naudoja vidinius tinklus su turinio valdymo sistema, arba sinchronines e. mokymosi aplinkas (pvz. VDU ir Kauno kolegija First Class programinę įrangą). Naudojami papildomi atskiri moduliai (pvz. dokumentų valdymo) tam tikroms funkcijoms realizuoti ir skleisti informaciją vartotojams. Prieiga prie vidinių tinklų yra apribota ir informaciją apie tai kas ten vyksta galima atsekti tik iš antrinių informacijos šaltinių arba centrų, aptarnaujančių vidinius tinklus, ataskaitų. Tačiau, ar galima laikyti vidinius tinklus akademinėje institucijoje akademinę informacijos sistema kelia pagrįstų abejonių. Intranetas - vidinis tinklas, naudojantis interneto protokolus duomenims persiųsti, galintis turėti ryšį su internetu. Dažniausiai jį susikuria įmonė arba žinyba savoms reikmėms [13]. Intranetas yra organizacijos vidinė sistema, leidžianti organizacijai elektroninėmis priemonėmis efektyviai valdyti vidinius organizacijos procesus. Intranetu dažnai vadinama ir tokia sistema, kur vidiniame tinkle organizacija saugo ir disponuoja svarbia informacija, arba sistema, naudojama kaip nuolatinė privaloma organizacijos darbuotojo darbo vieta. Intranetas pritaikytas skirtingiems ir unikaliems organizacijos poreikiams tenkinti. Tačiau pats intraneto naudojimas akademinėje institucijoje nereiškia, kad institucija turi akademinę informacijos sistemą. Akademinė informacijos sistema turi turėti atitinkamą turinį ir tenkinti akademinės institucijos poreikius.

Akademinės informacijos sistemos apibrėžimo nėra. Akademinės organizacijos įvairiai apibūdina savo akademinės informacijos sistemas. Vieni akcentuoja pagalbą studentams, kiti apibrėžia platesnį paslaugų ratą, kur jau kalbama ir apie studijų administravimą, finansų kontrolę, studentų įrašus ir kt. Interpretuojant informacijos sistemos apibrėžimą, galima būtų teigti, kad akademinė informacijos sistema tai materialinė sistema, galinti priimti, kaupti, saugoti, keisti, naudoti, skleisti akademinės organizacijos informaciją ir ją valdyti. Galima numanyti, kad akademinėje informacijos sistemoje turėtų būti personalizuotos aplinkos ir paslaugos, skirtos akademinėms institucijoms, darbuotojams bei studentams kasdieninei veiklai, studijoms bei savitarnai palaikyti. Šiuolaikinės informacijos sistemų tendencijos rodo, kad informacija interaktyvioje aplinkoje turi atsirasti ir būti naudojama visų pirma tų, kurie betarpiškai su ja susiję, ir ten, kur ji sukuriama. Tik tuomet bus galima išvengti tarpininkų įvedant informaciją, popierinės ir elektroninės informacijos dubliavimo, perkelti į virtualiąją terpę daugiau realiųjų gyvenimo procesų, neapsiribojant apskaita [30].

Paprogramės LiemSIS uždaviniuose tiksliai ir aktualiai buvo apibrėžtas akademinės informacijos sistemos poreikis ir uždaviniai, nurodant, kad „būtina sukurti akademinę LieMSIS posistemę, kuri, visų pirma, būtų orientuota į tiesioginius mokslo ir studijų procesų dalyvius: studentus, pedagogus, mokslininkus (portfolio principas), sukuriant jiems individualias aplinkas. Portfolio studentams būtų kaupiami studijų rezultatai, metodinė medžiaga, jiems būtų pateikiamos užduotys bei priimami atsiskaitymai, studentai formuotų savo individualius studijų planus. Dėstytojams portfolio taptų aplinka, kurioje jie fiksuotų studentų atsiskaitymų rezultatus, rengtų metodinę medžiagą, teiktų užduotis studentams. Mokslininkams portfolio leistų saugoti tyrimų medžiagą ir ją apdoroti, ieškoti bei kaupti literatūrą ir pan.“ Galima teigti, kad šis poreikis išliko aktualus ir šiandien, nes problemos išliko, o posistemė nebuvo sukurta. Vieningos akademinės informacijos sistemos idėjos autoriai turėjo įžvalgą, kad ši, sukurta, akademinė posistemė taptų „aplinka, kuria galėtų naudotis ne tik universitetų ir kolegijų studentai, bet ir kiti Lietuvos gyventojai ir įmonių darbuotojai, norintys persikvalifikuoti ar patobulinti kvalifikaciją. Tuo pačiu būtų sudaromos prielaidos nuotolinių ir tradicinių studijų procesų supanašėjimui ir suartėjimui, praplečiant Lietuvos gyventojų galimybes dalyvauti mokymosi visą gyvenimą procese“. Tad dar 2001 metais, kuriant akademinės informacijos sistemos koncepciją jau buvo žvelgiama į ateitį, kad e. mokymosi aplinkos gali susiliesti su akademinėmis informacijos sistema.

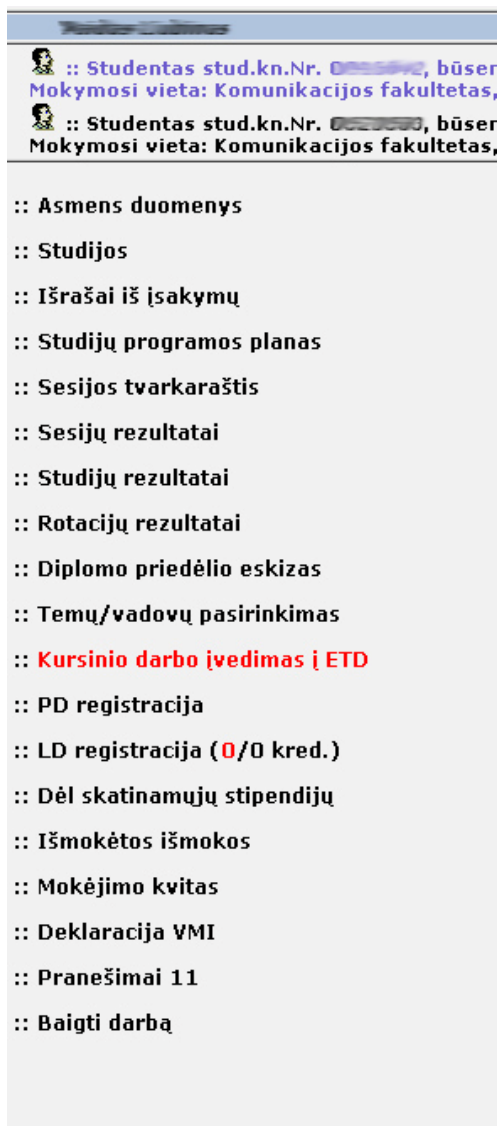
2.1. Akademinės informacijos sistemos Lietuvoje

2.1.1. Vilniaus universiteto informacinės sistemos atvejis

Vilniaus universiteto informacinę sistemą sudaro atskiros duomenų bazės ir sistemos, kurios federacijos architektūriniu principu jungiamos į vieną aplinką, naudojant karkasą (angl. frame). Informacinių technologijų taikymo centras nurodo, kad „administruoja ir prižiūri Studijų, Bibliotekos, Personalo, Finansų apskaitos, Dokumentų valdymo, Intraneto (VU vidinio tinklalapio), Rašto darbų, Kompiuterinio egzaminavimo (KES), Elektroninio plagiato aptikimo (EPAS), Priėmimo į VU pagrindines ir vientisąsias studijas, antrosios (magistrantūros) ir trečiosios (rezidentūros) pakopos studijas per atskirus etapus bei bendrąjį priėmimą į 44 aukštųjų mokyklų duomenų bazes ir kompiuterių programas“. Kiekvienas vartotojas turi autorizuotą prieigą prie visų jo darbinę veiklą atitinkančių kompiuterių programų ir duomenų bazių. Be to, kiekvienas VU IS naudotojas, turi galimybę peržiūrėti informaciją apie save (savitarnos programos) ir susipažinti su informacija, kuri skirta tik tam vartotojui. Pavyzdžiui, studento aplinka teikia tokią informaciją, kaip matyti 4 paveiksle.

Matyti, kad yra naudojamos skirtingos duomenų bazės ar moduliai, tam kad suformuoti studento individualią aplinką. Paskyros „Studijos“, „Studijų programų planas“, „Sesijos tvarkaraštis“, „Sesijų rezultatai“, „Studijų rezultatai“, „Rotacijų rezultatai“, „Diplomo priedėlio eskizas“ akivaizdžiai susijusios su Studijų bazės moduliais. „Temų/vadovų pasirinkimas“, „PD registracija“ ir „LD registracija“ siejami su Studijų ir Personalo moduliais ar duomenų bazėmis, o paskyros „Dėl skatinamųjų stipendijų“ – Studijų ir Finansų moduliai. Finansų reikalams priskiriamos ir „Išmokėtų išmokų“, „Mokėjimo kvitų“ ir „Deklaracija VMI“ paskyros. O visi duomenys apie studentą, su visais jį lydinčiais duomenimis, pradedant asmens kodu, vardu, pavarde ir baigiant baigta vidurinio išsilavinimo įstaiga bei vertinimais, ateina iš studentų priėmimo sistemos, mūsų atveju iš Priėmimo į Lietuvos aukštąsias mokyklas informacijos sistemos. Paskyrą „Pranešimai“ formuoja fakulteto ar universiteto direkcijos pranešimai.

Prisijungimas prie informacinės sistemos autorizuotas. Kiekvienas vartotojas turi unikalų identifikatorių ir slaptažodį. Yra trys vartotojų lygmenys: administratoriai, dėstytojai/darbuotojai ir studentai.

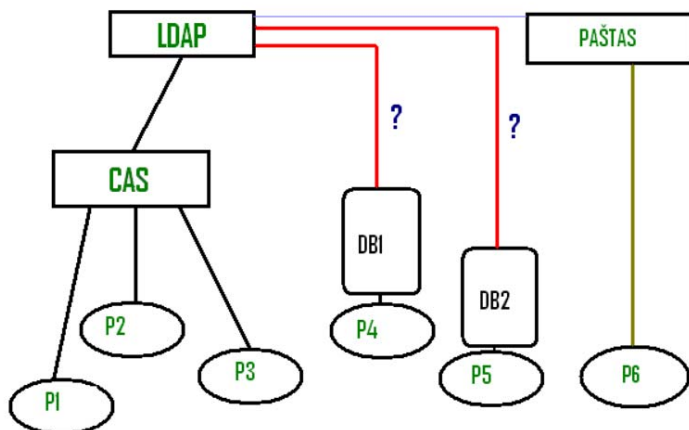


4 paveikslas. Studento pasirinkimo sąrašas.

Vartotojų įrašai saugomi lentelėse ir LDAP tarnybos saugykloje. Vartotojo įrašas sistemoje atsirandą sukuriant jam elektroninį paštą, tada, duomenų perdavimo posistemės pagalba, duomenys apie vartotoją perduodami į informacinę sistemą. Web technologijų pagrindu yra sukurta autentifikacijos sistema. Tačiau esant tokiai autentifikacijos schemai gana nepatogus vartotojų valdymas, duomenys dubliuojami perkeliant lenteles į skirtingas sistemas, nors patys vartotojai to ir nejaučia. Galimas ir duomenų neatitikimas skirtingose sistemose, nes jų redagavimas nėra uždraustas.

Studento atveju reikia žinoti keletą prisijungimo identifikatorių. Jungiantis prie informacijos sistemos ir prie elektroninio pašto naudojamas studento bilieto numeris ir slaptažodis, kurį reikia maždaug kas tris mėnesius keisti. Tuo tarpu jungiantis prie bevielio tinklo identifikatorius yra elektroninis paštas, o naudojantis bibliotekos paslaugomis – asmens kodas. Tai nėra patogu vartotojams.

5 paveiksle atvaizduojamoje schemos matyti, kad VU vartotojų duomenys yra saugomi LDAP tarnybos vartotojų saugykloje ir duomenų bazių lentelėse (DB1, DB2). LDAP tarnyba replikuoja su pašto sistema. Schemoje dar matyti atskira autentifikavimo sistema – Centrinė atpažinimo sistema, kuri ko gero turėtų perimti visas autentifikavimo pareigas visoms sistemoms.



5 paveikslas. Bendroji vartotojų tapatybės valdymo schema VU. Paimta iš A.Grigaliaus ir R.Sabaliausko pranešimo LITNET konferencijoje 2010 m.

VU informacinių technologijų taikymo centras ieško sprendimų dėl autentifikacijos suvienodinimo, bandomi įvairūs sprendimai ir greitai laiku galima tikėtis patogesnio naudojimo administruojančiam personalui, o vartotojams patogios autentifikacijos VU sistemose.

Informacijos sistema suprojektuota ir įdiegta Oracle duomenų bazių valdymo sistemų pagrindu. Ateityje planuojama plėtoti šio galingo duomenų bazių įrankio naudojimą.

2.1.2. Kauno technologijos universiteto akademinės informacijos sistemos atvejis

Kauno technologijos universitetas akademinę informacijos sistemą kūrė ilgus metus bandant įvairius sprendimus. KTU sistema veikia nuo 1996 m., o duomenys sistemoje nuo 1993 m. akademinės veiklos. Akademinė informacijos sistema buvo projektuojama ir realizuota Oracle kompanijos duomenų bazių valdymo sistemų pagrindu.

KTU akademinėje informacijos sistemoje naudojamas trijų lygių vartotojų prisijungimas: studentams, darbuotojams ir fakultetų raštinėms. Studento prisijungimo vardas yra studento bilieto numeris. Visi trys lygiai yra saugomi atskirose lentelėse duomenų bazėje. Duomenys apie naujai įstojusius studentus perkeliama iš Priėmimo į Lietuvos aukštąsias mokyklas informacijos sistemos.

Akademinėje informacijos sistemoje vartotojų duomenims suteikiami vaidmenys ir paskirstomi į grupes.

AKADEMINĖ INFORMACIJOS SISTEMA

Prisijungimas studentams:

Studento bilieto numeris:

Slaptažodis: *

* Studentai jungdamiesi pirmą kartą vietoj slaptažodžio turi įvesti asmens kodą.
Esant neaiškumams, kreiptis į savo fakulteto raštinę

Prisijungimas darbuotojams:

Tabelio numeris:

Slaptažodis:

Prisijungimas fakultetų raštinėms:

Prisijungimo vardas:

Slaptažodis:

6 paveikslas. Trijų lygių prisijungimas prie akademinės informacijos sistemos.

Dėstytojas, prisijungęs prie akademinės informacijos sistemos, gali rinktis savo studijų modulių programas ir atlikti visus galimus veiksmus: sukurti naują, redaguoti esančius, papildyti, taisyti. Be to, dėstytojui pateikiamos dokumentų formos, ruošimo nurodymai, modulio užpildymo pavyzdžiai bei kita pagalba.

Pagrindiniame lange dėstytojas gali matyti modulius pasirinkusių studentų sąrašus. Taip pat sukurti žiniaraščiai studentų vertinimams įvesti.

Studentui pateikiama informacija (7 paveikslas): kiekvienų mokslo metų studijų planas, kur matyti semestras, modulio kodas ir pavadinimas, kreditų skaičius, dėstomoji kalba, forma, studijų modulio atestavimo duomenys, įvertinimas. Kitame lange pateikiamas trumpas modulio aprašymas. Yra galimybė planą atspausdinti.

Be to, studentui teikiama informacija apie stipendijas ir studijų įmokas, universiteto įsakymai įtakojantys studento studijas, visi vertinimai ir kt. Šioje skiltyje galima projektuoti papildomas nuorodas ir į kitą informaciją informacinių sistemų integracijos atveju.

Studento individualus studijų planas

Pagrindinis meniu Apsilungti

2009/2010 m.m.
XXXXX Vardenis Pavardenis
Informatikos fakultetas
Studijų programa: M404DA01 Programų sistemų inžinerija

Individualus studijų planas 2009/2010 m.m.

Rudens semestras (03)

Modulio kodas ir pavadinimas	forma, kalba, kreditai	įvertinimai, data
T120M010 Informacinės technologijos projektavimo vadyboje	01 L 4 IS 09	2003.01.23
T120M012 Programinės įrangos įdiegimo tyrimas (praktika)	02 L 12 IS 09	2003.01.23
T120M025 Programų kūrimo priemonės	01 L 4 IS 10	2003.01.23

atsiskaityta: 20 kr.

Pavasario semestras (04)

Modulio kodas ir pavadinimas	forma, kalba, kreditai	įvertinimai, data
T000M038 Magistro baigiamasis darbas	01 L 12 IS 10	2003.06.10
T120M013 Programų priežiūra ir tobulinimas	01 L 4 IS 08	2003.05.30
T120M019 Programų kokybės analizė	01 L 4 IS 09	2003.05.29

atsiskaityta: 20 kr.

Dėl plano keitimo ir pastebėtų neatitikimų kreipkitės į fakulteto raštinę

[Atsispausdinti planą](#)

T120M010 Informacinės technologijos projektavimo vadyboje
Kodas: T120M010
Mokslų šakos kodas: T120
Mokslų šakos pavadinimas: Sistemų inžinerija, kompiuterių technologija
Studijų programa: Magistro
Pavadinimas: Informacinės technologijos projektavimo vadyboje
Pavadinimas anglų k.: Information Technologies in Project Management
Aukštoji mokykla: Kauno technologijos universitetas
Būtinasis pasirengimas: Projektavimo valdymas
Anotacija: Kompiuterinių technologijų ir bendradarbiavimo sistemų naudojimas projektavimo vadyboje. Projekto informacinės sistemos sukūrimas. Kompiuterinių grupinio darbo sistemų naudojimas. Rizikos/ kainos analizės sistemos sukūrimas. Kokybės valdymas. Pakeitimų valdymo sistema. Išlaidų apskaičiavimo priemonės. Grupės veiklos planavimas ir valdymas, naudojant IT. Dokumentų valdymas. Licenzijų rengimas. Produkto rėmimas (reklamoms sukūrimas ir organizavimas). Efektyvios pateiktys.
Anotacija anglų k.: Module develops the skills in new technologies, that enables effective project planning and management. Groupware. Electronic data interchange. Online database of project establishment. Financial management. Document management. Effective presentations.
Modulio grupė: 5 Specialaus lavinimo
Studijų sritis: 5.Fiziinių, biomedicinos, technologijos mokslų studijos (išskyrus nurodytąsias 6,

7 paveikslas. KTU akademinės informacijos sistemos studento studijų plano vaizdas.

2.1.3. Vilniaus Gedimino technikos universiteto atvejis

Vilniaus Gedimino technikos universitetas naudoja informacinę sistemą „Alma Informatica“. Universitetas ją pradėjo eksploatuoti prieš dešimtmetį ir sėkmingai plėtoja.

Dirbti su VGTU informacine sistema galima dviejų rūšių programomis: įprastinėmis ir internetinėmis.

- Įprastinės - tai programos, kurios turi būti įdiegtos vartotojo kompiuteryje.
- Internetinės - tai programos, prie kurių jungiamasi internetinės naršyklės programomis (Internet Explorer, Firefox ar pan.).

Universiteto informacinės sistemos duomenų bazę ir duomenis tvarko visos universiteto tarnybos: mokslo (mokslinės veiklos, mokslo sutarčių, doktorantūros duomenys), personalo (darbuotojų duomenų), studijų (studijuojančiųjų duomenų, pedagoginio krūvio, studijų programų ir dalykų, pažangumo, absolventų ir baigimo dokumentų, studentų paskolų, baigiamųjų darbų, paskaitų ir egzaminų tvarkaraščių, praktikos sutarčių ir bazių, studijų sutarčių duomenys), finansų (bendrabučio lėšų ir studijų įmokų duomenys), ekonomikos (sutarčių, sąskaitų, išmokų studentams duomenų), ūkinės (bendrabučių gyventojų, patalpų, telekomunikacinių duomenys), užsienio ryšių (tarptautinio bendradarbiavimo, tarptautinių dvišalių sutarčių duomenys), teisės (teisės aktų duomenys) direkcijos bei priėmimo komisijos. Šių tarnybų pateikti duomenys sudaro Vilniaus Gedimino technikos universiteto informacinės sistemos duomenų bazę.

Šiuo metu veikiančių internetinių programų pavyzdžiai būtų studentų vertinimo programa "sesija" bei studijų programų peržiūros programa.

Studentas, prisijungęs prie informacijos sistemos „Alma Informatica“ gali:



8 paveikslas. Studento prisijungimo prie „Alma Informatica“ langas

- įvertinti universitete dėstomus modulius ir jų dėstymo kokybę,
- suformuoti praktinio rengimo sutartį,
- užsisakyti pažymą apie įmokas už studijas arba gautą stipendiją,
- atspausdinti mokesčio už studijas kvitą,
- peržiūrėti įmokas,
- peržiūrėti studijų programą,
- peržiūrėti įsakymus,
- peržiūrėti įvertinimus (nurodyto semestro arba visus),
- peržiūrėti paskaitų tvarkaraštį,
- peržiūrėti egzaminų tvarkaraštį (tik nurodyto semestro),
- užpildyti ir atsispausdinti baigiamojo darbo anotaciją.

Informacinėje sistemoje „Alma Informatica“ sukaupti duomenys prieinami vartotojams pagal vartotojų lygmenis: dėstytojai, darbuotojai, studentai ir užsienio ryšiu.

2.1.4. Kitos institucijos

Mykolo Romerio universitetas naudoja informacinę sistemą „Mano studijos“, o Lietuvos Sveikatos mokslo universitetas informacinę sistemą „LSMUSIS“. Viešos informacijos apie šias informacines sistemas nėra. Galima tik identifikuoti autentifikavimui reikalingus duomenis. Mykolo Romerio universiteto informacinės sistema naudoja elektroninio pašto autentifikacijos duomenis, o Sveikatos mokslų universitetas leidžia jungtis su vartotojo vardu arba tabelio numeriu bei slaptažodžiu. Informacines sistemas turi susikūrę ir naudoja Tarptautinė teisės ir verslo aukštoji

mokykla, Verslo ir vadybos akademija. Tačiau viešos informacijos, kokius duomenis ir galimybes turi vartotojai, nėra.

Klaipėdos, Šiaulių, Vilniaus pedagoginis universitetai, Vilniaus dailės ir Lietuvos kūno kultūros akademijos bei Vilniaus kolegija naudoja arba pradeda naudoti modifikuotas Kauno technologijos universiteto akademinės sistemos kopijas. Pagrindas šiam teiginiui yra vienodas prisijungimo prie informacinės sistemos langas (kaip Kauno technologijos universiteto) bei internetinio adreso priklausomybė - ktu.lt ir liemsis.lt domenų valdytojai yra Kauno technologijos universitetas. Tad galima spėti, kad šių akademinėjų informacinių sistemų funkcionalumas yra toks pat kaip ir Kauno technologijos universiteto akademinės informacijos sistemos.

2.1.5. Akademinėjų informacijos sistemų bendrumai

Akademinės informacijos sistemos Lietuvoje gana naujas reiškinys. Visus su studijomis susijusius procesus apimančios akademinės sistemos kol kas nėra sukurtos. Uždavinius kuriant tokias sistemas formuoja aukštųjų institucijų poreikiai, tačiau didelę įtaką jų kūrimui turi ir išoriniai veiksniai: teisinės bazės kaita, finansavimas, studijų formos bei jų realizavimo būdų kaita ir t.t. Be to, kinta ir technologijos bei galimybės pasiekti akademinės informacijos sistemas per įvairius įtaisus (pvz. telefonai, i-Pad, delninkai ir t.t.).

Žvelgiant į nagrinėtus akademinės informacijos sistemų atvejus, didelių skirtumų nepastebėta. Dauguma informacinių sistemų realizuojamos Oracle duomenų bazių sistemų aplinkoje ir įrankiais. Dažniausiai naudojami karkaso (angl. framework-oriented) tipo programiniai sprendimai, kurios sudaro programinių modulių rinkinys ir technologija, kai iš atskirų modulių sukuriama vientisa, vartotojo poreikius atitinkanti informacinė sistema. Visos informacinės sistemos pasipildo naujais studentų duomenimis iš Priėmimo į Lietuvos aukštąsias mokyklas informacijos sistemos.

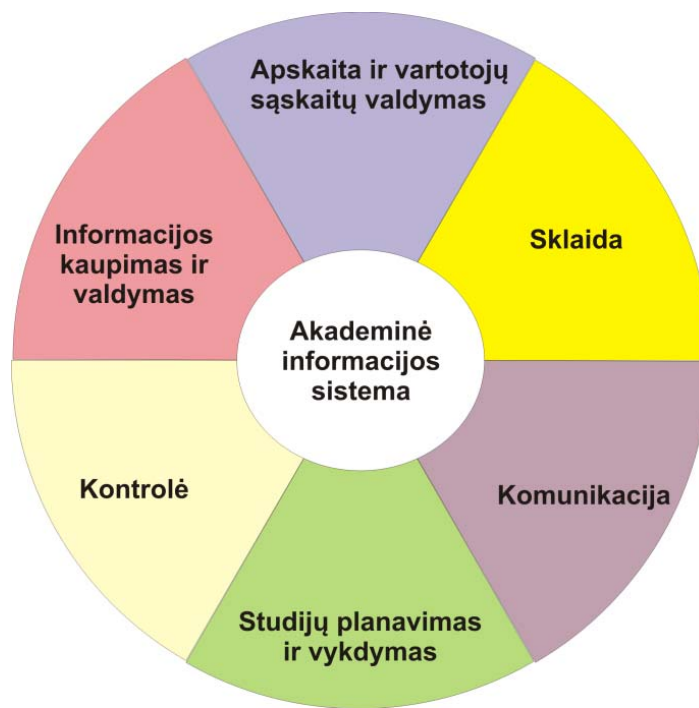
Skirtumai matomi informacijos pateikimo gausoje, informacijos pateikimo formose.

Žvelgiant į naudojamas autorizacijos schemas matyti, kad jos yra skirtingos ir dažniausiai informacijos sistemos turi perteklinių vartotojų duomenų lentelių, t.y. naudojama ne viena autentifikavimo sistema.

2.2. Akademinės informacijos sistemos veiklos ir uždaviniai

Kiekviena akademinė institucija, planuodama įsigyti ar kurti akademinę informacijos sistemą, numato, kokius poreikius jos turi tenkinti, kokias funkcijas turi atlikti bei kokie uždaviniai jai keliami. Pateikti standartinių, visoms institucijoms tinkančių reikalavimų sąrašą yra sudėtinga.

Nors akademinė institucijų tikslai tokie patys, tačiau jos gali rinktis skirtingus studijų proceso vykdymo metodus bei formas, gali būti vykdomos specifinės studijų programos, reikalaujančios atitinkamų studijų proceso organizavimo niuansų ir kt. Galima apibrėžti bendras akademinės sistemos funkcijas ir uždavinius ir jų vartotojus.



9 paveikslas. Akademinės informacijos sistemos pagrindinės veiklos ir uždaviniai

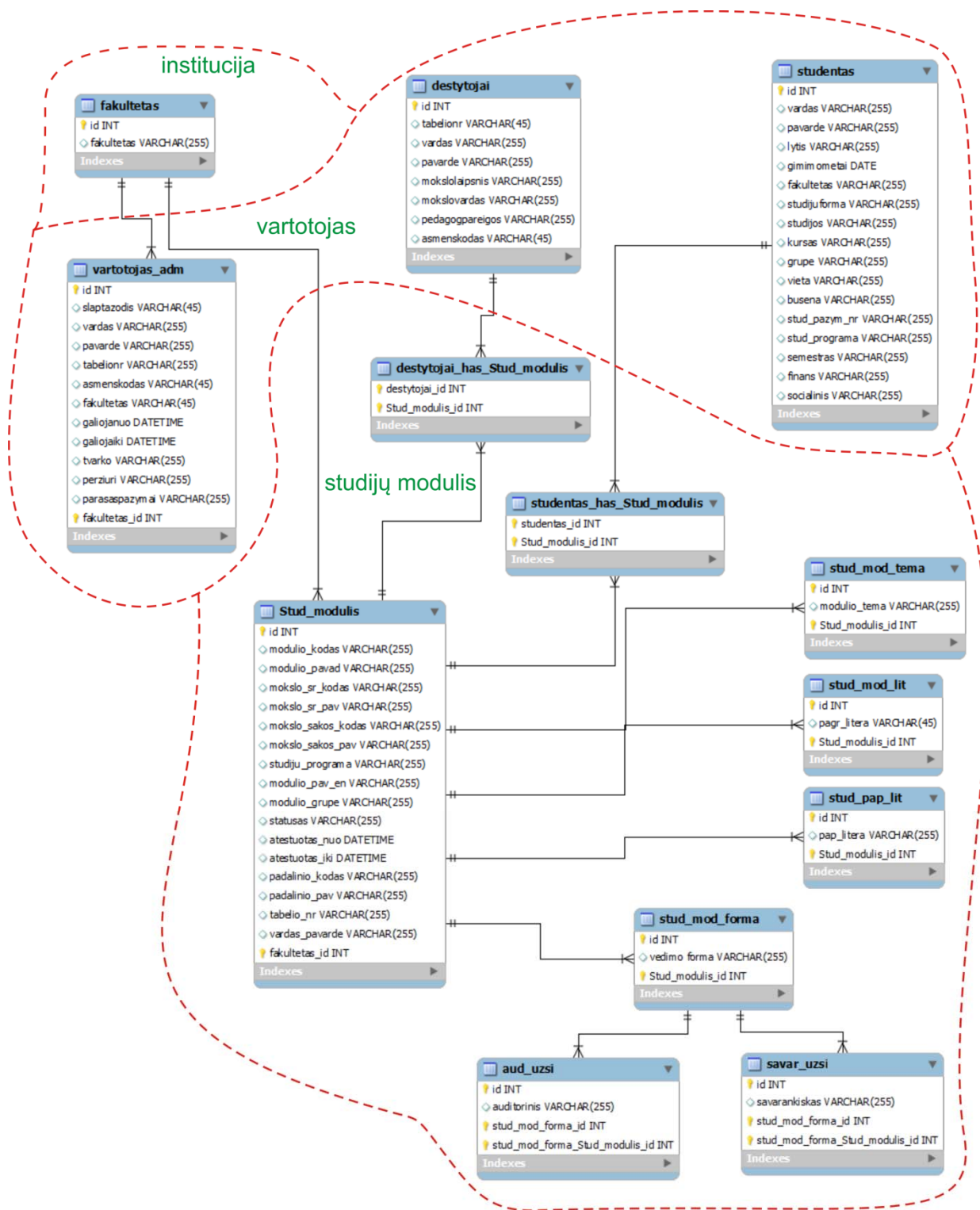
Pagrindinės akademinės informacijos sistemos veiklos funkcijos ir uždaviniai:

- Apskaitos ir vartotojų sąskaitų valdymo funkcija. Kiekvienais metais registruojami visi naujai įstoję studentai, duomenų bazė papildoma naujais įrašais. Esantys įrašai gali būti tvarkomi, redaguojami, perkeliama ir kt. Vykdoma dėstytojų duomenų apskaita.
- Kontrolės funkcija – duomenų bazėje fiksuojami dalykų egzaminų įvertinimai, matomi akademinė skolų turintys studentai, formuojami rotacijos rezultatai ir kt. Kaupiami duomenys ir apie dėstytojų atestaciją.
- Informacijos kaupimo funkcija – kaupiami duomenys apie akademinės institucijos studentus ir dėstytojus. Baigusių (ir nebaigusių) studijas studentų duomenis galima saugoti neribotą laiką atsižvelgiant į poreikius. Duomenys nuolat papildomi, atrenkami pagal tam tikrus kriterijus (studentas, darbuotojas, absolventas ir kt.). Ataskaitų formavimas pagal poreikius, kurias galima peržiūrėti, spausdinti, eksportuoti į kitas informacijos sistemas. Kaupiami duomenys apie studijų programas, kuriuos galima būtų perduoti suinteresuotoms struktūroms (pvz. Studijų ir mokymo programų registru, Studijų kokybės vertinimo centrui ir kt.).

- Studijų planavimo ir vykdymo funkcija. Studijų modulių kūrimas, tvarkymas, priežiūra, naujinimas, apskaita. Dėstytojų tarififikacijos. Studijų tinklelio sudarymas ir kt. uždaviniai.
- Komunikacijos funkcija. Komunikacija el. paštu, diskusijos, forumai.
- Sklaidos funkcija. Informacijos perdavimas visais vartotojų lygmenimis pagal poreikį.

Akademinės informacijos sistemos pagrindinė dalis yra studijos ir vartotojai, kurie susiję su studijų organizavimu. Negalima teigti, kad kiti komponentai nesvarbūs, tačiau be šių dviejų jokie procesai nėra įmanomi. Todėl pagrindinę akademinės aplinkos sistemos duomenų struktūrą bei ryšius tarp duomenų pabandydysime vaizduoti esybių ryšių diagrama įtraukiant tik pagrindines dalis. Esybių ryšių diagramai realizuoti naudosime pagrindinius sistemų laukus, kurie yra svarbiausi ir identifikuojami peržiūrint analizuotas akademinės informacijos sistemas. Remiantis dalykinės srities modeliais, vėliau gali būti braižomos detalios esybių ryšių diagramos, aprašančios fizinę sistemos duomenų bazės struktūrą. Tačiau tai nėra šio darbo uždavinys.

Raudona punktyrinė linija nurodo priklausomumą komponentų blokui.



10 paveikslas. Akademinės informacijos sistemos esybių ryšių diagrama

3. INTEGRACIJA

Internetas yra nauja aplinkos forma, kuri pakeičia verslo ir rinkos pobūdį, pertvarkydama ryšius tarp šių dviejų sričių. Šiuo metu visi perspektyviniai verslai turi technologines galimybes ir aktyviai vykdo veiklą Interneto erdvėje. Daugelis jų atlieka absoliučiai visas operacijas internete, kiti – dalį jų. Informacinės sistemos tampa vis sudėtingesnės, jos susideda iš daugelio heterogeninių dalių, jas sudaro daug komponentų, kurie galimai buvo sukurti skirtingais metodais ir skirtingu laiku. Sukurti monolitinę informacinę sistemą beveik neįmanoma. Todėl visada buvo poreikis integruoti informacines sistemas su jau esančiomis programinės įrangos taikomosiomis programomis ir pritaikyti jas naujoms faktinėms sąlygoms. Verta pažymėti, kad funkcionalumo išskirstymas po posistemės sudaro prielaidas sistemos lankstumui. Informacinių technologijų sprendimai buvo kuriami atsižvelgiant į šias problemas, o programinės įrangos gamintojų dėmesys buvo skirtas aukštos kokybės priemonių, skirtų informacinių sistemų integravimui, plėtojimui. Šiuo metu daugelyje bendrovių programinės įrangos sprendimų plėtojimo procesas nebėra specifinis procesas - tai viena iš strategijos dalių, kuriai taip pat reikia esamų taikomųjų programų integravimo. Integravimą reiktų suprasti kaip sudėtinių dalių (posistemų) jungimą į vieną visumą.

Jei informacinė sistema integruota, vartotojai gali stebėti visos sistemos elgseną, t.y. vartotojai naudojami vieninga sąsaja, jiems visiškai nerūpi, kaip dirba atskiros posistemės ar sistemos moduliai. Čia reiktų nepamiršti ir skirtingo integracijos laipsnio. Sistemų integracija galima keliais aspektais [37]:

- dalykiniu aspektu, kai siekiama sistemą padaryti skaidresnę ir paprastesnę vartotojui bei geriau palaikyti verslą;
- techniniu aspektu, kai siekiama supaprastinti jos aptarnavimą ir priežiūrą bei padidinti jos darbo našumą.

Atsižvelgiant į dinamišką informacinių technologijų kaitą, į didėjantį taikomųjų programų integravimo poreikį, į naujas informacines tiesiogines sistemas bei elektroninio verslo modelius, reikia naujų sprendimų, paremtų integracija bei tarpusavio sąveikavimu.

Integravimo procesas turi būti nagrinėjamas tiek metodiniu, tiek ir technologiniu požiūriu. Integruojant informacinę sistemą, pagal dr. A. Lupeikiene, reikia pasirinkti:

- Integravimo paradigmą;
- Strategiją;
- Integravimo architektūrą;
- Priemones;
- Platformą;

- Integravimo mechanizmus.

Nagrinėjant integravimo procesus šiame darbe plačiau apžvelgsime integravimo strateginius ir architektūrinius sprendimus, panagrinėsime duomenų integravimą, apžvelgsime aplikacijų integravimo principus. Tačiau nenagrinėsime priemonių ir platformų, nes tai turėtų būti siejama su konkrečių sistemų integracija ir jų integravimo mechanizmų kūrimu.

3.1. Integravimo strateginiai sprendimai

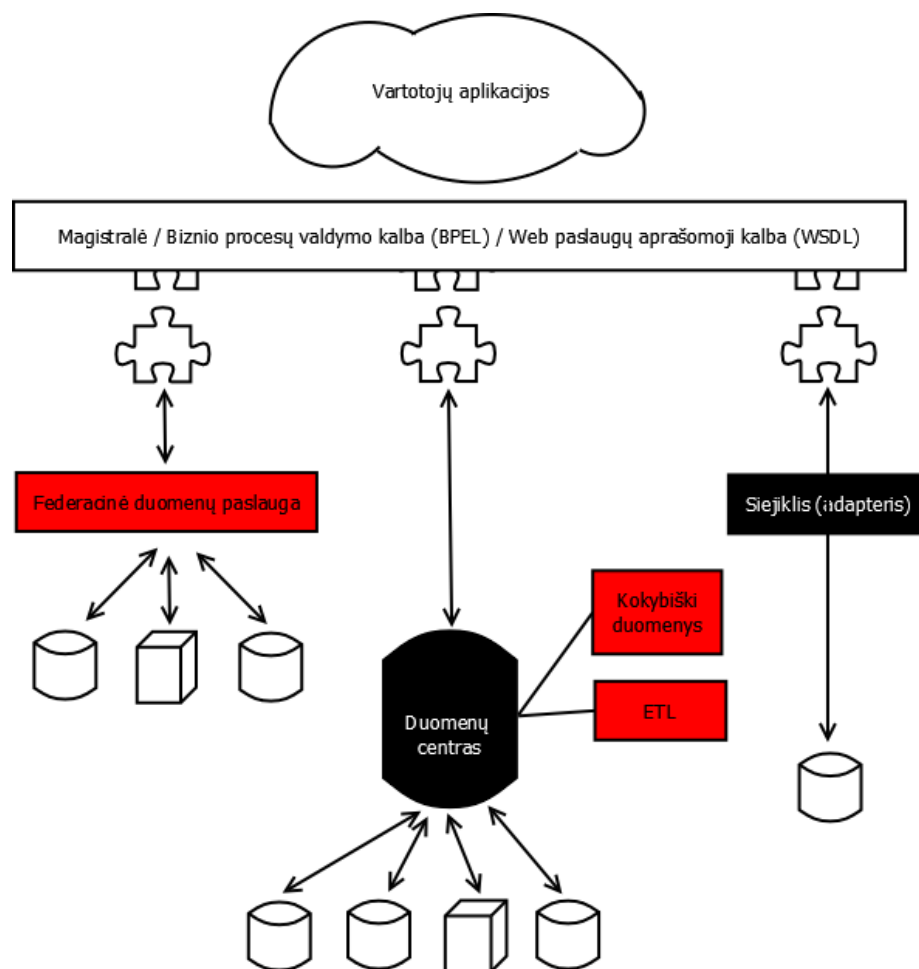
Egzistuoja ne viena informacinių sistemų kūrimo metodika, nusakanti, kaip projektuoti ir diegti kompiuterizuotas informacines sistemas, apimant technologijas ir naudotinus standartus. Tačiau metodikos nepateikia išsamių ir teoriškai pagrįstų IS integravimo su verslo sistema problemos sprendimų [1].

Vienas iš naudojamų sprendimų yra kai kurių ryšių tarp specifinių sistemų, kurios buvo apjungtos, suregulavimas. Tačiau šį ryšį sunku sukurti ir išlaikyti didžiojoje sistemų dalyje organizacijos lygiu. Integravimo tikslas yra priversti šias sistemas dirbti vienoje loginėje sistemoje. Taikomųjų programų integravimas įvairiose posistemėse arba organizacijoje gali būti pasiekiamas trimis kryptimis: vertikalia, horizontalia ir mišria [17].

Vertikalų integravimą sudaro taikomųjų paslaugų, kurios apjungtų esamas taikomas programas bei esamus duomenis, su naujomis taikomosiomis programomis, naujomis platformomis ar paslaugomis, sukūrimas. Tai reiškia, kad informacinės sistemos posistemės integruojamos veiklos, informacijos apdorojimo ir technologiniu lygmenimis. Visi trys lygmenys turi būti suderinti tarpusavyje.

Horizontalų integravimą sudaro taikomųjų paslaugų, kurios apjungtų jau esamas taikomas programas ir duomenis, esančius jose, sukūrimą. Tai komponentų integravimas viename abstrakcijos lygmenyje. Pavyzdžiui, kai integruojami informacijos procesai, kurie aprašyti skirtingais standartais, apjungiant tiesioginio įsigijimo sistemą su sąskaitybos sistema. Integravimą galima realizuoti naudojant taikomosios programos integravimo priemones organizacijos lygmeniu.

Mišrus (arba hibridinis) integravimas – tai vertikalaus ir horizontalaus junginys. Mišrus yra sudarytas tiek iš esamų, tiek naujų taikomųjų programų integravimo. Integravimą sudaro taikomųjų programų bei duomenų integravimas, naudojant atitinkamus saugumo standartus ir metodus.



11 paveikslas. Mišraus (hibridinio) integravimo sprendimo architektūra

Paprastai vertikali ir horizontali integravimo strategijos glaudžiai susijusios. Retai kada galima integruoti sistemą dalykiniu aspektu, neintegruojant jos techniniu aspektu.

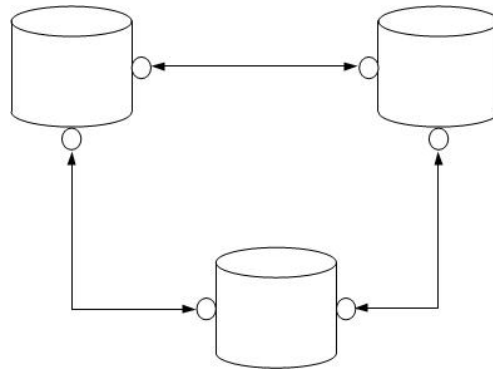
3.2. Integravimo architektūriniai sprendimai

Informacinių sistemų architektūra lemia visumą sprendimų apie informacijos sistemų organizavimo aspektus. Dr. A. Lupeikienė, cituojant M. Shaw ir D. Garlan, teigia, kad sistemos architektūra yra susijusi su sudėtinėmis dalimis ir jų jungimu į sudėtingas struktūras, su valdymo struktūromis, komunikavimo protokolais, funkcionalumo priskyrimu sudėtinėms dalims, su projektavimo alternatyvų pasirinkimu [1].

Galimi tokie architektūriniai sprendimai:

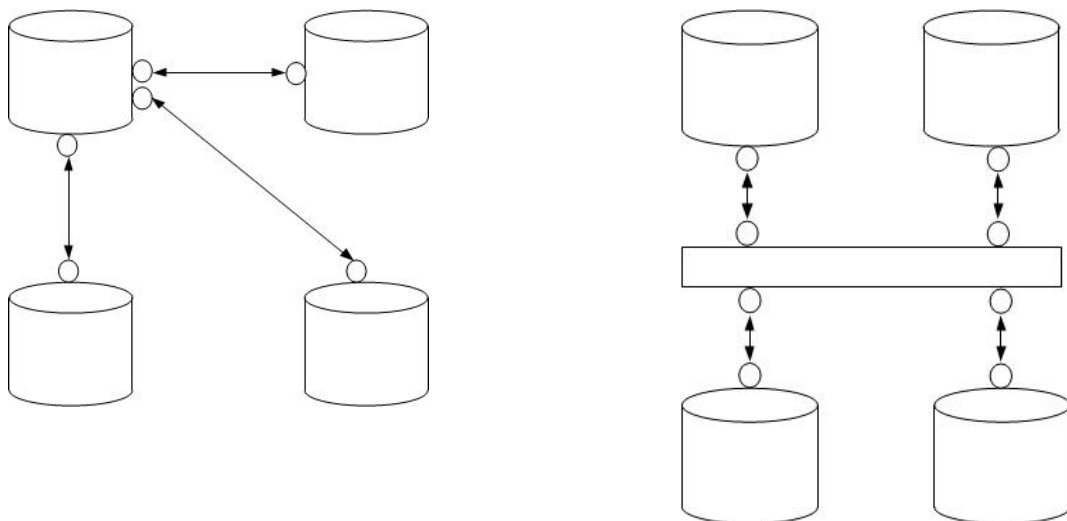
- Tiesioginis;
- Naudojant tarpininkus;
- Federacija.

Tiesioginio sujungimo (13 paveikslas) atveju sistemos sudėtinės dalys yra tiesiogiai sujungtos viena su kita.



12 paveikslas. Tiesioginio sujungimo atvejis

Integruojant sistemas per *tarpininkus* (13 paveikslas), sumažinamas sąsajų tarp sistemos sudėtinių dalių skaičius. Tačiau bene svarbiausias tarpininkų naudojimo privalumas – jų papildomas funkcionalumas ir teikiamos paslaugos.



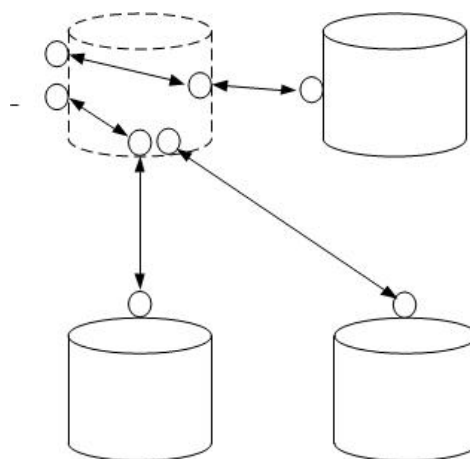
13 paveikslas. Sistemų integracijos per tarpininkus atvejai: naudoja bendrą sritį ir magistralę.

Sistemos integracija naudojant tarpininkus gali vykti keliais būdais:

- sistemos komponentai naudoja vieną bendrą darbo sritį.
- informacijos brokeris. Sistema dirba principu „klausimas-atsakymas“. Gaunama užklausa tarpininkui, tarpininkas suranda atsakymą ir grąžina įvykdytą užklausa.
- magistralė. Visi sistemos komponentai sujungti per informacinę magistralę ir sąveikauja per ją.

Tarpininkai, atitinkantys $m:1$ ir $1:m$ topologiją, paprastai vadinami brokeriais, o $n:m$ kardinalumą – magistralėmis. Vienareikšmės ribos tarp šių dviejų tarpininkų tipų nėra. Tarpininkai vadinami magistralėmis arba brokeriais priklausomai nuo to, kokiuose architektūros sluoksniuose jie naudojami (atsižvelgiama į nusistovėjusias tradicijas), kokie techniniai sprendimai pasirenkami jiems realizuoti, kokie konkretūs protokolai naudojami.

Federacija – tai architektūrinis sprendimas, kai tarpininkas užtikrina prieigą prie heterogeninių sistemos elementų ir centralizuotai tvarko jų teikiamas paslaugas. Tai yra, sukuriamas tarpininkas, kuris veikia kaip visos sistemos įgaliotinis, o komponentai yra savarankiški elementai, kuriems integravimas nekelia jokių papildomų reikalavimų. Kas yra viduje vartotojas nemato, jam pateikiama tik sistemos sąsaja. Visus vidinius ypatumus mato tik tarpininkas. Tarpininkas vartotojui pateikia sudėtinę sąsają, kurią sudaro skirtingų sistemų ir jų paslaugų rinkinys. Esant poreikiui iš sudėtinės sąsajos bet kuri sistemos dalis gali būti išimta arba papildyta naujomis bet kurios kitos sistemos dalimis, tačiau vartotojas matys tik atsirandančias nuorodas į paslaugas. Federacija kitaip dar vadinama sistemos integravimu naudojant nuorodas [8].



14 paveikslas. Sistemų federacijos atvejis

Virtualią integruotą sistemą sudaro nuorodų į realius sistemos komponentus visuma. Federacija leidžia jungti užklausas keliems komponentams, taip sudarydama įvaizdį, kad užklausa gali realizuoti viena paslauga.

3.3. Duomenų integravimas

Duomenų integravimas nurodo integravimą duomenų bazės lygiu arba duomenų migracijos iš vienos sistemos į kitą, arba bendrų duomenų saugyklų, kurias naudotų kelios sistemos, sukūrimą. Integravimą sudaro duomenų ištraukimas, perkėlimas bei patvirtinimas šaltinių sistemose, o taip pat

ir duomenų perkėlimas į tikslines sistemas arba bendras duomenų saugyklas. Sukuriamos duomenų saugyklos ir sukuriami mechanizmai, leidžiantys įvairiems sistemos komponentams bendrai naudotis saugyklose saugomais duomenimis. Tai įgyvendinama niekaip nekeičiant informacijos apdorojimo procesų. Duomenims integruoti taikomi šie pagrindiniai metodai:

- duomenų federacija;
- duomenų jungimas;
- duomenų išrinkimas.

Duomenų federacija – tai integruotas požiūris į išskirstytus duomenis, išvengiant duomenų pertekliškumo [7]. Jis įgyvendinamas kuriant tarpininką, kuris užtikrina vieningą prieigą prie heterogeninių duomenų šaltinių. Tarpininkų vaidmenį paprastai atlieka duomenų bazių valdymo sistemos, kurios jungia duomenų šaltinius į vieningą federaciją ir tvarko užklausų srautus.

Duomenų federacijos atveju sukuriama viena saugykla, kurioje saugomi ne duomenys, o metaduomenys (informacija apie duomenis ir jų buvimo vietą). Technologija leidžia sukurti vieną standartizuotą požiūrį į skirtingus duomenis ir „paslėpti“ daugelį skirtingų duomenų saugyklų.

Duomenų jungimo atveju naudojami specializuoti įrankiai duomenims išrinkti, transformuoti ir išsaugoti vienoje centralizuotoje vietoje. Kitaip sakant, sukuriama nauja duomenų saugykla, į kurią duomenys surenkami iš įvairių sistemos posistemų. Svarbiausias šio būdo trūkumas – sunku sukurti vieną duomenų modelį, tinkantį viešiesiems vartotojams.

Duomenų išrinkimo atveju kuriamos specializuotos programos, kurios išrenka duomenis, saugomus skirtingų posistemų saugyklose, juos transformuoja, jungia ir pateikia vartotojams ar kitiems sistemos komponentams. Šis būdas visada gali būti įgyvendintas, tačiau reikalauja daug sąnaudų ir nėra lankstus, nes pakitus duomenų struktūroms, reikia modifikuoti specializuotas programas.

Atrodytų, kad duomenų federacija, duomenų jungimas ir duomenų išrinkimo atvejai yra skirtinguose poliuose. Duomenų federacija veikia kaip duomenų sutraukimas iš daugelio šaltinių į vieną aplinką, o duomenų jungimas veikia kaip duomenų perkėlimas į vieną centrą. Duomenų išrinkimo atveju kuriamos atskiros posistemės. Tačiau realybėje šie visi minėti atvejai gali būti naudojami kartu imant geriausias jų savybes organizacijos duomenų centro struktūrose. Iš tiesų, pragmatiškos organizacijos pradeda nuo paslaugų integracijos „nuo taško link taško“ ir tik vėliau, esant poreikiui, pereina prie duomenų jungimo ar išrinkimo metodų, jei atsiranda toks poreikis. Duomenų jungimo ir duomenų išrinkimo atvejais sąnaudos ir kaštai yra dideli, nes reikia kurti naujas duomenų saugyklas arba specializuotus įrankius duomenų išrinkimui. Išskyla klausimas, kokie duomenys turi būti perkeliami į saugyklą, o kurie nereikalingi ir ar jų neprireiks vėliau. Sukurti vieną duomenų modelį sudėtingas darbas ir dažnai tampa neįmanomas. Todėl svarstant

duomenų integravimo klausimą ir renkantis jų integracijos metodą verta rinktis pragmatišką žingsnį - duomenų federaciją. Panagrinėsime plačiau duomenų federacijos metodą.

3.3.1. Duomenų federacijos duomenų apjungimo metodas

Naudojant duomenų federacijos metodą, vartotojas mato tik vieną sąsają. Žvelgiant iš vartotojo pusės, kai yra taikomas duomenų federacijos metodas, vieta, kur yra duomenys, neturi jokios reikšmės. Tokia pati situacija ir duomenų nukreipimo atvejais – vartotojui nereikės žinoti, kokia programavimo kalba dominuoja arba kokia programinė sąsaja yra naudojama informacinėje duomenų bazėje. Pavyzdžiui, jeigu naudojama SQL (Struktūrinė užklausų kalba), visiškai nėra svarbu, kokių SQL dialektu pateikiamas šaltinis. Vartotojui nereikia žinoti, kaip duomenys yra saugomi fiziškai, kokia jų priklausomybė ir jų fragmentavimo aspektai, taip pat ar jie yra kopijuojami ar teikiami tiesiogiai iš šaltinio, kokie naudojami jungimo į tinklą protokolai ir kt.

Vartotojo sąsaja duomenų federacijos serveryje gali sietis tik su vienu virtualiu duomenų šaltiniu. Nenaudodama duomenų federacijos struktūros, užklausa turi bendrauti su daugialypiais šaltiniais individualiai per skirtingas sąsajas ir skirtingus protokolus. Tyrimai parodė, kad duomenų federacijos metodo naudojimas padeda ženkliai sutrumpinti plėtros laiką kai reikia integruoti daugialypius šaltinius [7].

Daugumai vartotojų dažniausiai reikia to paties – integruotos informacijos. Viena vertus, kiekvienas vartotojas pats apjungia informaciją iš skirtingų šaltinių. Kita vertus, integruotas informacijos telkinys yra sukuriamas vieną kartą, o papildomas daugybę kartų ir laikoma vienoje vietoje taip sukuriant vieną tašką, per kurį atliekami visi pakeitimai. Tai sumažina priežiūros ir vėlesnių patobulinimų išlaidas.

Duomenų federacijos metodo įdiegimas, skiriant didelį dėmesį pažangioms duomenų apdorojimo technologijoms, daugeliu atvejų turi geresnio darbo savybes. To priežastis yra tai, kad pagrindinis duomenų federacijos serveris gali optimaliai paskirstyti darbo krūvį tarp paties savęs ir įvairių šaltinių. Duomenų federacijos serveris nustatys, kurio darbo krūvio dalis ar segmentas atliekamas efektyviausiai reaguojant į gautas užklausas ir optimizuos atsakymo laiką.

Pritaikius duomenų federacijos metodą konkrečiam integravimo scenarijui, šio specifinio apjungto prieinamumo rezultatas gali būti pateiktas kaip paslauga daugeliui vartotojų, kuriems ji reikalinga. Be to, apjungus duomenis lengviau juos valdyti.

Vykstantys įvairių įmonių ar organizacijų susijungimai ir išsigijimai dažnai pareikalauja, kad būtų integruoti nesulyginami duomenų šaltiniai į jungtinę duomenų visumą. Sprendimas dėl to, kaip sukurti vieningą sistemą ir aplinką, dažnai priimamas išbandant schemą rankiniu būdu bei įvertinant priemonių, patirties, erudicijos bei organizacijos kultūros aspektus. Laikas, pastangos ir išlaidos,

susijusios su integracija, gali viršyti numatomą apjungimo naudą, jei naudojamosi tradicinėmis, paveldėtomis, struktūromis. Sukūrus aplinką, pagrįstą struktūriniu informacijos paslaugų metodu, kuris įgyvendinamas pačioje paslaugoje, gali būti ilgainiui pagerintos sistemos panaudojamumo savybės. Informacinės paslaugos yra pagrindinės, į paslaugas nukreiptos architektūros (angl. Service-Oriented Architecture) (toliau - SOA), dalis. Pati į paslaugas nukreipta architektūra jau yra apjungimo metodas, kuris pateikiamas paslaugoms, kurios tinkamos naudoti pakartotinai ir daugeliu atveju praplečia tradicinių, ne į paslaugas orientuotos architektūros, duomenų struktūrų įgyvendinimo galimybes [17].

Duomenų federacijos metodo pagrindinis tikslas yra išvengti duomenų pertekliaus pateikiant integruotą aplinką į įvairiarūšius šaltinius. Duomenų federacijos serveris, t.y. tas komponentas, per kurį įdiegiamas duomenų federacijos modelis, turi pateikti standartines užklausų sąsajas netradiciniam SOA. Tai užtikrina, kad daugelis tradicinės duomenų bazės taikomųjų programų galėtų naudoti apjungtus duomenis. Pagrindinis duomenų federacijos serveris taip pat turi gebėti organizuoti užklausų optimizavimą tam, kad efektyviau reaguotų į užklausas. Duomenų įvairiarūšiškumas reikalauja tinkamo išdėstymo, įgyvendinant tinkamą priėjimą prie integruotos aplinkos, suskaidant bei paskirstant darbo krūvį. Turint aiškiai išreikštą prieigą prie tokio integruoto aplinkos, duomenų federacijos serveris turi sinchronizuoti duomenų valdymą įvairiuose šaltiniuose į logišką darbo vienetą. Tai užtikrina, kad valentingumo, pastovumo, izoliavimo ir ilgaamžiškumo kriterijai, susiję su duomenų operacijomis, atitiks numatytuosius reikalavimus duomenims ir jų šaltinių integralumo vientisumą.

Duomenų federacinis serveris teikia sprendimą, kaip efektyviai sujungti ir apdoroti informaciją iš įvairiarūšių šaltinių. Šis metodas realizuojamas sinchroniškai, apdorojant integracijos metodą, skirtą duomenų paskirstymui. Duomenų federacinis serveris yra atsakingas už užklausų, nukreiptų pasinaudojant integruota įvairių šaltinių aplinką, gavimą. Jis pertvarko duomenis, naudodamas sudėtingus optimizavimo algoritmus, kurie suskaido užklausą į po-operacijų serijas, vadinamas užklausų padalinimu ir perrašymu, taikydamas po-operacijas atitinkamiems šaltiniams, rinkdamas rezultatus iš kiekvieno šaltinio, sutvarkydamas integruotus rezultatus ir, galų gale, sugražindamas integruotus rezultatus į užklaustos pradžių. Tokia apdorojimo seka yra atliekama sinchroniškai ir realiu laiku.

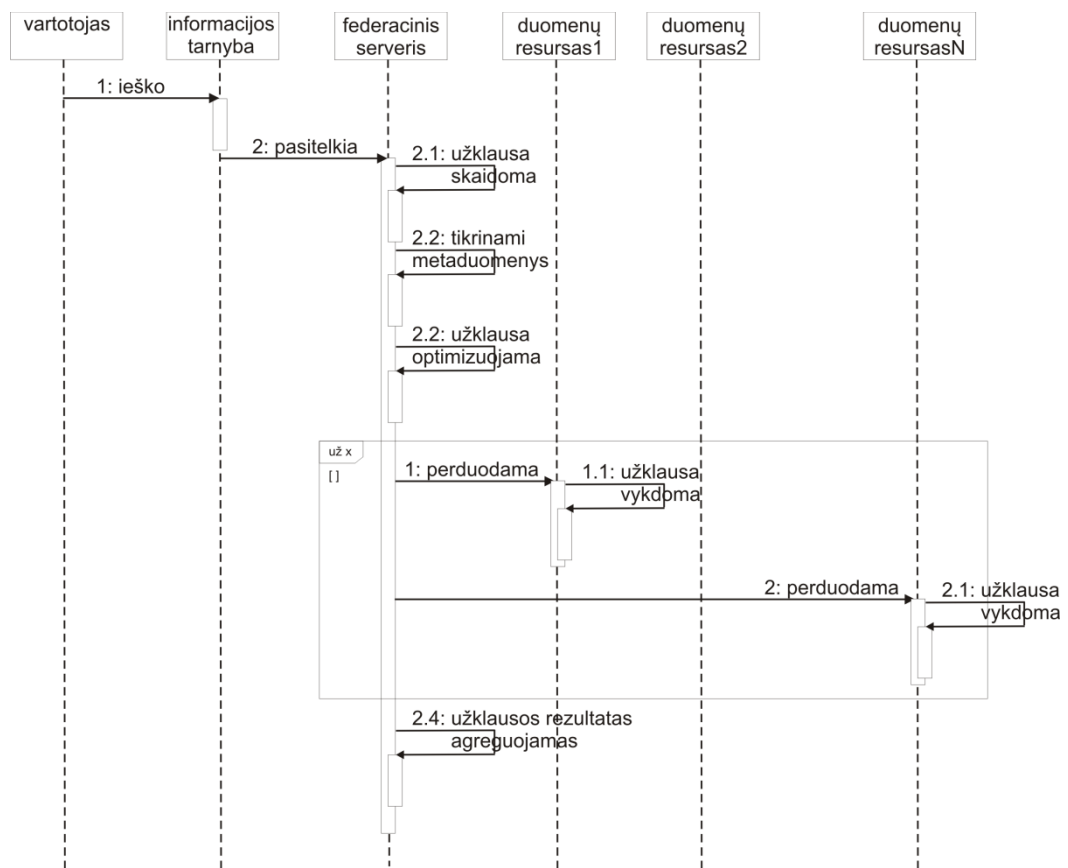
Projektuojant duomenų federacijos metodu, atsiranda poreikis atlikti duomenų elementų žymėjimą duomenims, kurie ateina iš įvairių duomenų šaltinių, ir kurie yra integruotos aplinkos ribose. Pavyzdžiui, informacija apie klientą, tokia kaip vardas ir adresas, gali būti saugoma vienoje lentelėje vienoje duomenų bazėje. Bet taip pat ji gali būti saugoma ir keliose lentelėse kitoje duomenų bazėje. Norint sudaryti integruotą vaizdą, reikia tuos skirtingus pateikimo būdus apjungti į bendrą vaizdą. Apjungimą rankiniu būdu gali atlikti ir žmonės arba jiems gali padėti šiuolaikinės

priemonės, kurios remiasi įvairiais apjungimo algoritmais, kurie taip pat atitinka visus reikiamus informacinius reikalavimus. Tai leidžia duomenų federacijos serveriui gauti užklausas iki integruoto vaizdo ir apskaičiuoti geriausią atliekamų po-operacijų skaičių ir tipus.

Naudojant duomenų apjungimo modelį SOA kontekste, reikia aktyvuoti jungtinių užklausų grupę ir užregistruoti ją kaip paslaugas, įeinančias į SOA. Duomenų apjungimo rezultatas projektavimo metu, paprastai, yra federacinės aplinkos, panašias į reliacinių duomenų bazių vaizdus, kurias po to galima išdėstyti arba sukurti duomenų federacijos serveryje.

Tad duomenų federacinis serveris gauna integruoto vaizdo užklausą. Remdamasis apjungimo apibūdinimu, duomenų federacijos serveris suskaido federacinę užklausą į daugelį po-operacijų. Šį žingsnį įtakoja daugelis veiksnių:

- žiūrima, kur yra duomenys, kurių reikia norint atlikti federacinę užklausą;
- kokių operacijų reikia norint sutvarkyti įvairiarūšius šaltinius, tokių kaip skirtingi duomenų tipai, normalizuoti ar nenormalizuoti duomenys;
- kokias operacijas turėtų kompensuoti duomenų federacijos serveris;
- kurias operacijas duomenų federacijos serveris turėtų nukreipti į šaltinius tam, kad būtų geriau išnaudojamos jų galimybės, kad būtų sumažintas duomenų perdavimas ir kad būtų pagerintas bendras darbas.



15 paveikslas. Duomenų federacija sekų diagrama

Norint tenkinti šiuos veiksnius, reikia turėti žinių apie duomenų šaltinių sistemą ir jų užklausų apdorojimo galimybes.

Kai duomenų federacijos serveris nustato geriausią visų po-operacijų atlikimo strategiją, jis prisijungia prie duomenų šaltinių – tiek struktūrinės, tiek nestruktūrinės informacijos – tam, kad išrinktų reikiamus duomenis, naudodamas tiems duomenų šaltiniams būdingas sąsajas. Gaunamas rezultatas yra apjungiamas į integruotą aplinką ir rezultatas yra pateikiamas vartotojui. Paveiksle 15 sekų diagramos pagalba pateikiami duomenų federacijos veiksmai.

SOA atveju vartotojas pateikia užklausą pagal iš anksto numatytą formą duomenų federacijos serveriui. Ši perduoda užklausą atitinkamoms SOA užklausoms. Čia jau atliekami tie patys užklausos išskaidymo, optimizavimo ir vykdymo žingsniai, kaip apibūdinta aukščiau. Vienintelis skirtumas SOA kontekste yra jo paskutinis žingsnis. Duomenų federacijos serveris paverčia tradicinio duomenų apjungimo metodo rezultatą paslaugos atsakymu ir sugrąžina jį paslaugos vartotojui per iš anksto numatytą paslaugos sąsają.

Duomenų federacijos metodo funkcionalumą galima pasiekti naudojant arba su duomenų baze susijusias technologijas, tokias kaip optimizavimo priemonė arba balansavimas, arba vietinę taikomąją programinę įrangą. Dėl užklausų optimizavimo sudėtingumo, susijusio su įvairiarūšiais šaltiniais, geriausia įrangos gamintojų siūloma praktika yra naudoti tokį duomenų apjungimo realizavimą, kuris įtakotų užklausų optimizavimo technologiją, kurią pateikia daugelis duomenų bazių valdymo sistemų.

Taikant duomenų federacijos metodą svarbu suprasti jo savybes ir kaip jį įtakoja nefunkciniai reikalavimai [9]. Atliekant duomenų apjungimą (arba vėlesniuose etapuose) sistemas galima praplėsti papildomomis struktūromis. Papildomos struktūros susijusios papildomais nefunkciniais reikalavimais bei funkcionalumu, kurio gali reikėti paslaugai. Tarpinę atmintį ir duomenų kopijavimo struktūras galima naudoti norint papildyti duomenų apjungimą arba kuriant mišrią struktūrą. Šios ir bet kurios kitos struktūros, kurios gali būti naudojamos bendrame įgyvendinimo procese, turi būti naudojamos atsargiai, kadangi jos gali trukdyti įgyvendinti kai kuriuos nefunkcinius reikalavimus, dėl kurių buvo pasirinktas duomenų apjungimas. Pavyzdžiui, jos gali padidinti duomenų užklausų vykdymo gaišties laiką ir sukurti duomenų perteklių. Reikia surasti suderinamumo taškus, paremtus nefunkciniais reikalavimais ir architektūriniais sprendimais. Nefunkcinių reikalavimų savybės apima tokius dalykus, kaip:

- Duomenų apsauga. Įrangos kūrėjai dažniausiai siekia įtakoti esančius saugumo mechanizmus, tokius kaip pradinių sistemų autentifikavimas ir autorizavimas. Dėl heterogeninio ir paskirstyto šios aplinkos pobūdžio, gali kilti sunkumų, susijusių su vieno prisijungimo bei visuotinio priėjimo kontrole, o tai yra už duomenų apjungimo struktūros masto ribų. Norėdami

susidoroti su tais sunkumais, architektai turi apjungti duomenų federacijos modelį su saugumo struktūromis.

- Duomenų gaištis laikas. Duomenų federacijos modelis leidžia laiku ir integruotai prieiti prie pradinių šaltinių duomenų.

- Šaltinio duomenų nepastovumas. Esant prieigai prie duomenų šaltinio duomenų ir gavus užklausą, duomenų federacijos serveris visada sugrąžins esamą duomenų šaltinio informaciją. Pabrėžtina, kad duomenų federacijos metodas nekuria šaltinių duomenų kopijų, šaltinio pokyčių nereikia dauginti ar apdoroti naudojant šį metodą.

- Duomenų logiškumas ir kokybė. Padažnėjus atvejams, kai reikia atlikti sudėtingas duomenų valymo, standartizavimo bei pakeitimo operacijas, neigiamo poveikio bendros reakcijos laikui tikimybė padidėja. Taip yra dėl sinchroniškos reakcijos į užklausas duomenų federacijos modelyje. Bet koks pakeitimas reiškia papildomą darbo krūvį reaguojant į integruotą užklausą.

- Duomenų prieinamumas. Integruotų duomenų prieinamumas priklauso nuo duomenų federacijos serverio bei integruotų šaltinių saugyklų prieinamumo užklausos vykdymo metu. Jeigu vienas iš pagrindinių serverių arba bet kokia jungtis tarp apjungimo ir šaltinio saugyklos neveikia, integruota aplinka tampa neprieinama.

- Integruoto modelio pokyčių poveikis. Galimybė atlikti pokyčius duomenų federacijos serveryje gali sumažinti paslaugos iniciatoriaus ar vartotojo galimybę susidurti su šiais pokyčiais naudojantis integruotais duomenimis. Integruotoje aplinkoje pokyčius galima atlikti nereikalaujant, kad jie atsispindėtų duomenų šaltinių moduluose.

- Operacijų dažnis. Duomenų federacijos serveriui užklausa siunčiama sinchroniškai. Gavus atsakymą vartotojas gali teikti kitą užklausą. Tad duomenų federacijos serveris turėtų priimti vykdymui lygiagrečias užklausas, kurias pateikia daug vartotojų. Visos užklausos turi būti realizuojamos vienodai ir vienodomis sąlygomis kaip bet kuri kita. Duomenų federacijos serverio gebėjimas atlikti operacijas dideliu dažniu yra nusakomas koeficientu, kuriuo jis gali prieiti prie duomenų šaltinių sistemų, bei tų duomenų šaltinių sistemų gebėjimu atsakyti į užklausas.

- Užklausų sutapimas. Duomenų federacijos serveris turi labai panašias savybes kaip ir duomenų bazių ar turinio valdymo serveriai. Gebėjimą efektyviai tvarkyti lygiagrečią prieigą lemia duomenų federacijos serverio eksploatacinės savybės bei integruoti duomenų šaltinių serveriai.

Aukščiau apibūdintą duomenų federacijos struktūrą, kaip duomenų operacijų metodą apžvelgėme, prieš integruotas ar trumpalaikės (virtualios) aplinkos sukūrimą, kur tikri duomenys yra sukaupti daugialypiuose skirtinguose šaltiniuose. OMG konsorciumo [35] tyrėjai apibrėžė privalumus ir trūkumus, taikant duomenų federacijos metodą.

Kada taikyti duomenų federacijos metodą:

- Informacijos pateikimo laikas yra svarbus prioritetas. Tada duomenų federacija suteikia greitą prieigą prie informacinių šaltinių be ilgų informacijos valdymo infrastruktūros pokyčių.

- Duomenų federacija sutvirtina keliamus reikalavimus, susijusius su duomenų kopijavimu, prieiga prie duomenų, kurie yra šaltinyje. Tokie reikalavimai atsiranda dėl nuostatų arba taisyklių, kurios riboja duomenų judėjimą arba kopijavimą, pvz., prenumeruojami duomenys ar asmeniniai duomenys iš skirtingų šaltinių.

- Realus laiko prieiga prie paskirstytos informacijos, tarytum iš vieno šaltinio. Informacija gali būti tiek struktūrinių, tiek nestruktūrinių duomenų pavidalu.

- Lankstus ir išplėstinis informacijos integravimo modelis dinamiškai besikeičiančiai aplinkai, ypač schemas evoliucija: dėl duomenų pertekliaus stokos, jungtinės schemas pokyčiai sumažina pokyčių poveikį integruotoms sistemoms.

- Duomenų apjungimo privalumai geriausiai išnaudojami, kai gaunamas nedidelis užklausų skaičius prieš apribotus rezultato komplektus, sudarytus iš daugelio nuoseklių, papildomų duomenų šaltinių.

Trūkumai, susiję su duomenų federacijos naudojimu:

- Integravimo scenarijai, kuriems reikia sudėtingų pakeitimų, norint sudaryti integruotą vaizdą, darys neigiamą įtaką reakcijos laikui naudojant šį metodą.

- Duomenų šaltinių serveriai gali būti neigiamai veikiami padidėjusio darbo krūvio, kai jie turi grąžinti duomenis, kurių prašoma jungtinėje užklausoje. Norėdama apdoroti užklausą integruotoje aplinkoje, federacijos serveris nusiųs po-operacijas į integruotus šaltinius. Kuo sudėtingesnės tos po-operacijos ir kuo dažniau jos yra siunčiamos į duomenų šaltinius, tuo daugiau papildomos darbinės apkrovos duomenų šaltinių serveriai turi apdoroti.

- Scenarijai, kurie yra didelių tarpinių rezultato komplektų, perkeltų iš tikslinių duomenų šaltinių į jungtinius serverius, priežastis, gali turėti ženklios įtakos darbui.

- Situacijos, kuriose taikomajai programinei įrangai reikia santykinai aukšto integruotų duomenų prieinamumo, nėra geros kandidatės šio metodo taikymui. Integruotų duomenų prieinamumas visiškai priklauso nuo visų federacinių ir šaltinių serverių tinkamumo dalyvauti procese, serverių ir tinklo prieigos.

- Daugelio duomenų federacijos metodo naudojimo sričių naudoja duomenis ribotai. Daugelis naudoja SQL kaip programavimo kalbą ir gali palaikyti tik SQL pakeitimus.

Apsisprendus dėl sistemų integracijos strategijos ir architektūros, reiktų apžvelgti taikomųjų programų, kurių pagrindu realizuojamos aplinkos, integravimo principus. Taikomųjų programų

integravimas parodo strateginį metodą, skirtą kelių sistemų apjungimui tiek paslaugų, tiek duomenų lygiu.

3.4. Programinių paketų integravimas

Šiuo metu daugelis programinės įrangos gamintojų susiduria su didelio skaičiaus programinės įrangos taikomųjų programų, parašytų skirtingomis programavimo kalbomis, integravimo problema. Šios taikomosios programos yra sudarytos tam, kad veiktų skirtingose programinės įrangos platformose, operacinėse sistemose, kad naudotų skirtingas duomenų bazines, kad būtų pritaikytos skirtingiems tinklo protokolams arba vidutinio darbo platformoms. Kuriama ir tobulinama komponentinių programų sintezės sistema skirta šiam procesui automatizuoti. Viena iš šios sistemos problemų – tikslios specifikacijos sukūrimas [16].

Informacijos sistemų integracijos kontekste sąvokos „taikomoji programa“, „komponentinė programa“ ir „programinis paketas“ bus suprantamos kaip programų rinkinys, skirtas tam tikros rūšies uždaviniams spręsti bei vartojamos kaip sinonimai. Programinių paketų integracijos poreikį turėtų lemti veiklos poreikiai ir galimybės. Paprastai, išskiriami šie programinių paketų integracijos lygmenys:

- Duomenų lygmuo. Tai yra duomenų integracija, kuri reiškia, kad programinio paketo naudojama duomenų saugykla, leidžiama naudotis kitiems programiniams paketams.

- Pranešimų lygmuo. Integracija vyksta keičiantis pranešimais tarp skirtingų programinių paketų. Pranešimų siuntimas ir gavimas turi įtakos programiniam paketui, kuriame tam suprogramuotos atitinkamos sąsajos.

- Procesų lygmuo. Integracija grindžiama veiklos proceso aprašymu ir esamų programinių paketų prijungimu prie aprašomo proceso. Tai tam tikras pranešimų integracijos lygmens išplėtimas, nes duomenimis keičiamasi siunčiant pranešimus, tačiau pranešimų apsikeitimo procesą organizuoja tarpinis programinis paketas.

Pasirenkant programinio paketo integracijos lygmenį reikia atsižvelgti į organizacijos poreikius ir įvertinti realizacijos kainą. Nors procesų lygmens integracija yra techninis sprendimas, kiti veiklos veiksniai taip pat turi būti įvertinti.

Jei organizacijos programinių paketų integracijos uždaviniai nėra sudėtingi ir disponuojama nedideliu biudžetu, tuomet geriausias integracijos sprendimo pasirinkimas yra programinio paketo integracija duomenų lygmenyje. Šis sprendimas dažniausiai pigiausias, nors tolimesnė jo plėtra ir palaikymas gali būti sudėtingas ir brangus. Šio sprendimo realizavimas gali būti sudėtingas su išorinėmis organizacijomis (dėl formų suderinamumo, standartų, sąvokų ir t.t.).

Pranešimų lygmens integracija taikoma nedidelėms organizacijoms, kuriose veiklos procesų kiekis mažas arba veiklos procesai nėra gerai apibrėžti. Pranešimų lygmens integracija gali būti naudojama kaip laikinas sprendimas pereinant į procesų lygmens integraciją. Iš pranešimų lygmens integracijos perėjimas į procesų lygmens integraciją yra pakankamai paprastas.

Procesų lygmens programinių paketų integracija įsivaizduojama kaip susijusių veiklos procesų aibė. Informacinių technologijų komponentai apdoroja realius duomenis ir skaičiuoja, o informacijos apsikeitimas atliekamas siunčiant pranešimus. Procesų lygmens programinių paketų integracija reguliuoja pranešimų eiliškumą, sąlyginių ir loginių taisyklių sukūrimą ir tokiu būdu apibrėžia bendrą procesų vykdymą.

Programinių paketų integracija yra įmanoma žinant technines paketų savybes bei realizaciją. Šiame darbe analizuojant tiek e. mokymosi, tiek akademinį informacijos sistemų aplinkas, įsitikinome, kad programiniai paketų realizacijos technologijų įvairovė plati ir bendro vardiklio išvesti nepavyktų. Tačiau vertinant programinių paketų integravimo lygmenis, tinkamiausias mūsų atveju būtų procesų lygmuo. Labiausiai paplitę du realizacijos principai: pranešimų tarpininko principas ir programų paketų serverio principas.

Pranešimų tarpininko principas. Programiniai paketai dažniausiai prijungiami prie pranešimų tarpininko panaudojant sluoksnį - siejiklį tarp pranešimų tarpininko ir programinių paketų. Sluoksnis – siejiklis nustato atitikmenis tarp integruoto programinio paketo ir standartinės pranešimų tarpininko sąsajos. Privalumai yra tai, kad kiekvienas programinis paketas sujungtas ne tarpusavyje, o tik su pranešimų brokeriu. Be to, pranešimų apsikeitimas vyksta asinchroniniu būdu. Pranešimų tarpininkas atlieka visas duomenų transformacijas, todėl ši operacija atliekama centralizuotai. Tarpininkas turi galimybę nustatyti veiklos taisykles. Tokią sistemą lengviau konstruoti. Trūkumas būtų tai, kad viso proceso transakcijos nepalaikomos, kadangi pranešimų apsikeitimas vykdomas asinchroniškai, todėl reikia sukurti kompensuojančius procesus (procesas, kai įvykdžius pirminį procesą atstatoma sistemos būseną, buvusi prieš pirminio proceso vykdymą, pvz. ištrynimą). Be to, sudėtinga duomenų saugos realizacija.

Programų paketų serverio principas. Visos sistemos ašimi yra programų paketų serveris. Programų paketai serveryje sujungiami panaudojant atitinkamus komponentų aplankus (angl. component wrapper). Procesas vyksta per serverio viduje esančius komponentus. Siejiklio ir aplanko pora sudaro tam tikrą abstrakcijos lygmenį, kurio dėka komunikacija tarp programinių komponentų atliekama tarsi komponentai būtų vykdomi programų paketų serverio aplinkoje. Privalumas yra tai, kad kiekvienas programinis komponentas sujungtas tik su programinių paketų serveriu, o ne tarpusavyje. Čia taip pat serveris atlieka visas duomenų transformacijas, todėl viskas centralizuota, reikiamos jungtys realizuojamos sukuriant siejikius ir aplankus, o proceso transakcijos gali būti realizuotos, jei komponentai jas palaiko. Pagrindinis trūkumas yra tai, kad

visas komunikavimas vyksta sinchroniškai, todėl ilgai trunkantis procesas atima nemažai serverio resursų, lieka mažiau priemonių veiklos procesų, taisyklių ir duomenų transformacijų palaikymui.

Norint atlikti programinių paketų integraciją procesų lygmenyje, reikia apibrėžti vidinius organizacijos veiklos procesus. Procesų lygmens integracijos principas sprendžia ne tik integravimo problemas, bet ir leidžia suvienodinti veiklos procesus visoje organizacijoje.

4. E. MOKYMOŠI APLINKOS INTEGRAVIMAS Į AKADEMINES INFORMACIJOS SISTEMAS

Atliekant e. mokymosi aplinkos integravimą į akademines informacijos sistemas aptarsime integravimo paradigmą, strategiją ir architektūrą. Pasiūlysimė kelias autentifikavimo proceso realizavimo priemones apjungiant skirtingas duomenų struktūras.

4.1. Integravimo paradigma

Kiekviena nauja informacijos sistemų integracija yra unikalus procesas. Nesant integracijos procesus grindžiančios ir įprasminančios bendros teorinės paradigmos, vyrauja fragmentiškas integracijos vaizdas, nepadedantis suvokti ilgalaikių šio proceso tendencijų bei tikėtinų rezultatų. Dažniausias kelių informacijos sistemų integracijos rezultatas yra naujos informacijos sistemos sukūrimas. Integruojant informacijos sistemas, galima būtų remtis informacijos sistemų kūrimo gyvavimo ciklo modeliais, tačiau „informacinių sistemų kūrimo metodikos nepateikia išsamių ir teoriškai pagrįstų jų integravimo su verslo sistema problemos sprendimų, apimančių architektūrų atitikimo užtikrinimą“ [1]. Integruojant informacijos sistemas reikia analizuoti esamas sistemas, specifikuoti vartotojų ir programinės įrangos reikalavimus, projektuoti, rašyti naują kodą naujai sistemai, testuoti ir diegti. Tačiau reikia nepamiršti, kad integravimo procesas vykdomas su veikiančiomis sistemomis ir jų veiklos procesų trikdymas gali sukelti nepageidaujamų padarinių, kurie sukompromituotų integraciją. Todėl būtini tinkami integravimo strateginiai ir architektūriniai sprendimai, tinkami ir saugūs duomenų integracijos metodai.

E. mokymasis yra viena iš aukštosios mokyklos paslaugų, kuri teikiama studentams ir dėstytojams, mokymuisi ir mokymui. Lietuvoje vienokia ar kitokia forma visos aukštosios mokyklos naudoja e. mokymosi paslaugą studijų procese. Lietuvoje dauguma aukštųjų mokyklų taip pat turi savo akademines informacijos sistemas, per kurias yra teikiama studijų administravimo paslauga ir vartotojų informavimas. Akademines informavimo sistemos yra daugiau informacinio ir pagalbino konteksto aplinkos, o e. mokymosi aplinkos – studijų programų tikslų įgyvendinimo erdvės. Pagrįstai kyla šių aplinkų paslaugų integracijos klausimas. Vertinant tai, kad abi aplinkos skirtos studijų kokybei gerinti, jų integracija sukurtų naują kokybinę studijų valdymo, administravimo ir jų tikslų įgyvendinimo aplinką. Viena sistema papildo kitą: prie informacijos apie studijuojamus dalykus atsiranda studijuojamo dalyko turinys su konspektu, studijuojamo modulio įvertinimo rezultatas, pasiektas naudojant e. mokymosi vertinimo įrankius, atsispindi studijų pasiekimų lentelėse ir t.t. Todėl sąveika tarp e. mokymosi aplinkos ir akademines informacijos sistemos reikalinga pirmiausia tam, kad akademines institucijos vartotojai nesidubliuotų ir tik esami

institucijos bendruomenės nariai galėtų naudotis jos resursais. Antra, tai pagerintų duomenų administravimą, nes visi duomenys būtų pasiekiami per vieną sąsają.

Norint jungti šias e. mokymosi ir akademinės informacijos sistemas, reikia įvertinti jose vykstančius veiklos procesus. Vienodi arba sutampantys veiklos procesai gali tapti sistemų jungimo sąlyčio taškai.

Teorinis ir praktinis e. mokymosi aplinkos į akademinę informacijos sistemą integravimas nėra sudėtingas procesas. Tačiau egzistuoja kliūtys, kurių įveikimas ir inspiruotų integracijos rezultatai. Pagrindinė problema išlieka duomenų saugumas. Aktualiausi šiame aspekte yra akademinėje informacijos sistemoje saugomi duomenys. Tačiau kyla ir nemažai kitokių išorinių kliūčių, kaip kad pavyzdžiui nuotolinių studijų įteisinimas aukštojoje mokykloje, dėstytojais bijantys technologinių naujovių ir kt.

4.2. Integravimo strateginiai ir architektūriniai sprendimai

Renkantis e. mokymosi aplinkos ir akademinės informacijos sistemos integravimo strateginį sprendimą akivaizdu, kad reikia integruoti tiek dalykiniu, tiek techniniu aspektais. Integruojant minimas informacijos sistemas siekiama, kad vartotojai turėtų paprastesnę ir aiškesnę sąsają, mažinant prisijungimo taškų skaičių ir norint gauti reikiamą informaciją, atlikti užduotis, pabendrauti su kitais vartotojais ir atlikti daug kitų veiklų. Todėl reiktų rinktis mišrią (hibridinę) integravimo strategiją. Būtų kuriamos naujos taikomosios programos, kuriose atsispindėtų esamų taikomųjų programų rezultatai ir galimybės bei atliekamas duomenų integravimas, naudojant tinkamus saugumo standartus ir metodus.

Informacijos sistemos architektūrą lemia įmonės ir jos aplinkos, taip pat pačios informacijos sistemos ribojimai: lėšos, skiriamos sistemai sukurti ir įdiegti, personalo kvalifikacija (pvz. tam tikrų metodikų, kalbų mokėjimas, sistemų projektavimo patirtis), technologinės galimybės, techninis įgyvendinamumas ir pan. [1].

Sprendžiant architektūrinį sistemų integravimo klausimą paželkime į visus galimus sprendimus. Tačiau kokį sprendimą bepasirinktume, mums pirmiausia reikia suvokti, ko siekiame. Institucijos ir informacinių sistemų architektūra yra institucijos tikslų (strateginių ir taktinių) rezultatas [1]. Todėl reiktų apibrėžti, kokių tikslų ir rezultatų siekiame. Mūsų atveju, integruojant e. mokymosi aplinkas į akademinės informacijos sistemas, pagrindinis tikslas yra sukurti vieną kokybinį informacijos telkinį, kuriame galėtume priimti, kaupti, saugoti, keisti, naudoti, skleisti akademinės organizacijos informaciją ir ją valdyti, taip pat suteikti galimybę dalį mokymosi proceso perkelti į virtualią erdvę.

Tiesioginio sujungimo atveju reiktų tinkamai išanalizuoti jungiamas sistemas, jas dekomponuojant. Viena iš sistemų turėtų būti pagrindinė ir jos pagrindu atlikta integracija. Mūsų atveju pagrindinė yra akademinė informacijos sistema, o e. mokymosi aplinką reiktų integruoti. E. mokymosi aplinkos veiklos įrankius sukelti į akademinę informacijos sistemą sudėtinga dėl skirtingų programinių produktų naudojamo. Todėl tektų perkoduoti veiklos procesus integruojant į naująją informacijos sistemą. Tai reiškia, kad tektų vieną aplinką kurti naujai arba atlikti kodo konvertavimą. Integracijos proceso kaina tokiu atveju smarkiai išauga, reikalingi aukštos kvalifikacijos specialistai, kurie gerai žinotų abi sistemas.

Tiesioginis sujungimo architektūros atvejis tinkamiausias tada, kai informacijos sistemos programiniai produktai yra koduoti toje pačioje duomenų bazių valdymo sistemoje, sistemų sąsajos yra panašios ir yra galimybė jas nesudėtingai papildyti.

Sistemų integracija naudojant tarpininkus leidžia sumažinti sąsajų tarp sistemos sudėtinių dalių skaičių. E. mokymosi aplinkos integracijos į akademinės informacijos sistemas atveju turime dvi skirtingas sąsajas, kurias reiktų pakeisti viena bendra. Sukūrus bendrą sąsają, reiktų nustatyti informacijos judėjimo taisykles, kurios užtikrintų tinkamą užklausų tenkinimą. Tačiau tarpininkas leistų tik tenkinti vartotojo užklausas, pagerintų informacijos sklaidą tarp dviejų sistemų, tačiau sudėtinga organizuoti veiklas, kurios apibrėžtos e. mokymosi aplinkoje.

Federacijos architektūrinis sprendimas taip pat sukuria tarpininką. Tačiau čia tarpininkas organizuoja prieigą prie visų bendros sistemos elementų ir centralizuotai tvarko visas teikiamas paslaugas. Naudojant federaciją būtų sukuriama viena sisteminė sąsaja, kurioje būtų pateikiamos skirtingų sistemų ir jų paslaugų rinkinys. Tiek e. mokymosi aplinkoje, tiek akademinėje informacijos sistemoje jokių esminių pakitimų nereiktų daryti, jos išliktų kaip savarankiškos sistemos. Tačiau vartotojas matytų vieną sąsają ir vieną sistemą, t.y. nuorodų sąrašą į atitinkamus komponentus. Kas slepiasi už šios sąsajos, jis nematytų. Ši sąsaja nesunkiai gali būti atnaujinama naujomis nuorodomis į naujas paslaugas, taip pat ir panaikinamos nuorodos, kurios nebetenkina sistemai iškeltų tikslų. Be to, integracijos realizavimo procesas visiškai netrikdytų integruojamų sistemų darbo.

Pasirinkus federacijos architektūrinį sprendimą, galima būtų sukurti virtualią integruotą sistemą. Sąsają galima kurti atskirai, į ją integruojant tik nuorodas į komponentus, kurie ir toliau gyvuoja savo sistemose (16 paveikslas). Be to, šis sprendimas yra pigiausias, nes nereikia perprogramuoti pačių informacijos sistemų ir aplinkos. Tad federacijos architektūrinis sprendimas priimtinausias, integruojant e. mokymosi aplinką į akademinės informacijos sistemas.

:: Studentas stud. kn. nr. 0000000, būsena: studijuoja
Mokymosi vieta: Komunikacijos fakultetas, informacijos sistemų vadyba, ...

:: Asmens duomenys
:: Studijos
:: Išrašai iš įsakymų
:: Studijų programų planas
:: Sesijos tvarkaraštis
:: Sesijų rezultatai
:: Studijų rezultatai
+ **Mano paskaitos**
:: **Tarpiniai įvertinimai**
:: Rotacijų rezultatai
:: Diplomo priedėlio eskizas
:: Darbų įvedimas į ETD
:: PD registracija
:: LD registracija (0/0 kred.)
:: Dėl skatinamųjų stipendijų
:: Išmokėtos išmokos
:: Mokėjimo kvitas
:: Deklaracija VMI
:: Pranešimai
:: **Forumai**
:: **Kalendorius**
:: Baigti darbai

Pasirinktos nuorodos
turinys

16 paveikslas. Galimos sąsajos variantas (modifikuota esama VU informacijos sistemos studento sąsaja).

16 paveiksle pavaizduotame studento sąsajos variante paliktos tos pačios parinktys, kurios yra galimos ir šiuo metu akademinėje sistemoje, pridėti raudonai pažymėti pasirinkimai. Kaip jau minėta, nuorodų gali būti ir daugiau. Be to, pasirinkimas „Mano paskaitos“ yra išsiskleidžiančio meniu tipo, išskleidus matytusi visų galimų paskaitų sąrašas.

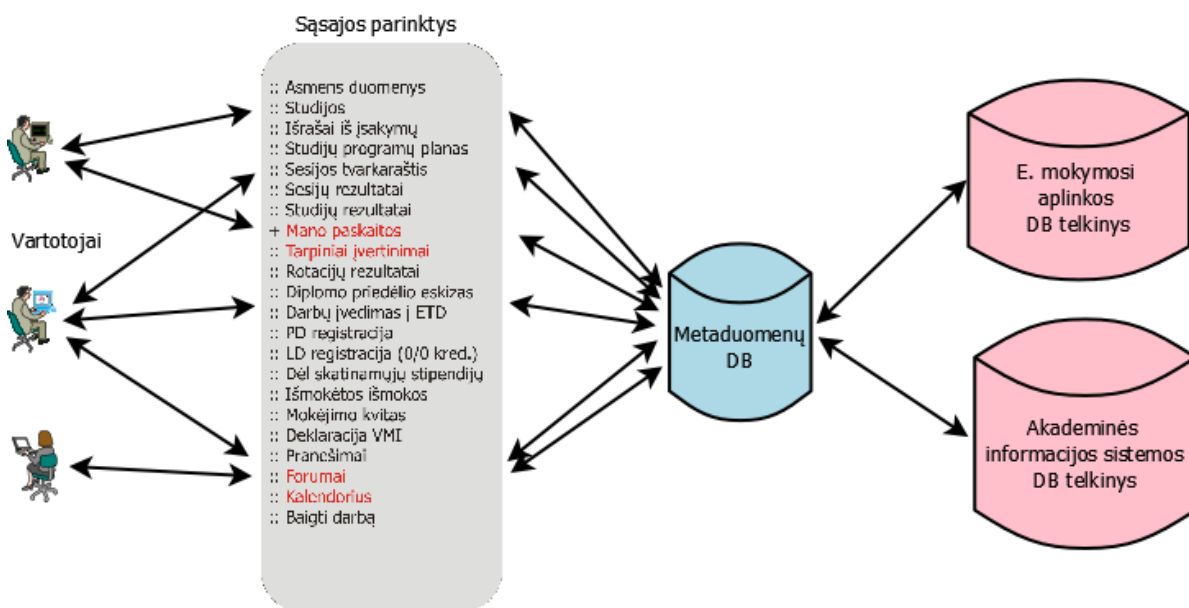
4.3. Integruojamų sistemų duomenys

Tiek e. mokymosi aplinkoje, tiek akademinėje informacijos sistemoje duomenys saugomi reliacinėse duomenų bazėse. Duomenų integracijos jungimo metodo atveju teoriškai neturėtų būti sudėtinga aprašyti integravimo procesą susiejant skirtingų sistemų duomenų laukus vieną su kitu per tarpinę lentelę arba pvz. metaduomenų aprašą. Vėliau problema nesunkiai išsprendžiama pasitelkiant duomenų eksporto/importo technologijas arba suprogramuojant konvertavimo programą ar įskiepi. Konvertavimas - vienos formos arba formato keitimas kita forma arba formatu, nekeičiant turinio (išsaugant informaciją) [13]. Tačiau šis metodas inicijuoja naujos, centralizuotos, duomenų bazės sukūrimą, į kurią surenkami visi duomenys iš sistemų posistemų. Duomenų bazės

kūrimas ir priežiūra reikalauja papildomų resursų. Kaip jau minėta anksčiau, yra labai sunku sukurti vieną duomenų modelį, kuris tiktų visiems vartotojams. Problemos kyla jei pirminiame duomenų šaltinyje įvedami nauji duomenų laukai.

Panaši situacija ir su išrinkimo metodo naudojimu, lankstumo trūkumas yra didžiausia problema, ypač jei sistemų struktūros kinta. Be to, reikalingos specializuotos programos, kurios transformuotų užklausas ir pateiktų vartotojams ar sistemų komponentams rezultatus.

E. mokymosi aplinkos integracijos į akademinės informacijos sistemas atveju geriausia tinka duomenų federacijos metodas. Jau aptartas aukščiau federacijos architektūrinis sprendimas panašiai nusako ir duomenų federacijos metodą. Vartotojui visiškai nesvarbu, kur yra duomenys, jam visai nesvarbu, kaip jie yra užkoduoti ir sudėlioti duomenų bazėje. Vartotojas gauna vieną sąsają ir visus reikiamus, užklausas tenkinančius, duomenis. Kaip jau minėta aukščiau, naudojantis duomenų federacijos metodu yra sukuriama viena duomenų saugykla, tačiau į ją sudedami ne duomenys, o jų metaduomenys, t.y. informacija apie tai, kur jie yra ir apie pačius duomenis. Visa ši saugykla realizuojama duomenų bazių valdymo sistemos pagrindu.



17 paveikslas. Duomenų federacijos metodo taikymo schema

Taikant duomenų federacijos metodą, esamą struktūrą nesunkiai galima praplėsti papildomomis paslaugomis, kurios yra naudojamos akademinėje institucijoje, padedant vykdyti kokybiškas studijas.

Tačiau reiktu nepamiršti, kad papildomos struktūros reikalingos ir nefunkciniams reikalavimams tenkinti, kurie gali būti reikalingi kokybiškoms paslaugoms. Dažniausiai ir labiausiai rūpi duomenų saugos klausimas. Pačios duomenų bazių valdymo sistemos sprendžia saugos klausimą savo sistemos ribose, tačiau integruojant kelias sistemas į vieną visumą atsiranda būtinybė

integruoti ir saugumo struktūras. Tačiau skirtingų sistemų saugos sprendimai gali būti labai skirtingi, o siekis, kad vartotojas galėtų jungtis prie visų duomenų vienu prisijungimu sunkiai įgyvendinamas. Todėl svarbus ir vartotojų registravimosi funkcijų integravimas ir prieigų kontrolės administravimas.

4.4. Vartotojų registravimosi funkcijų integravimas

Apsikeitimas duomenimis apie vartotojus gali būti atliekamas apjungus duomenų bazių vartotojų lenteles arba perleidžiant vartotojų tvarkymą kitoms sistemoms. Yra patogų kai visi institucijos vartotojai yra vienoje vietoje ir galimas jų registravimas, administravimas ir šalinimas.

4.4.1. LDAP katalogas

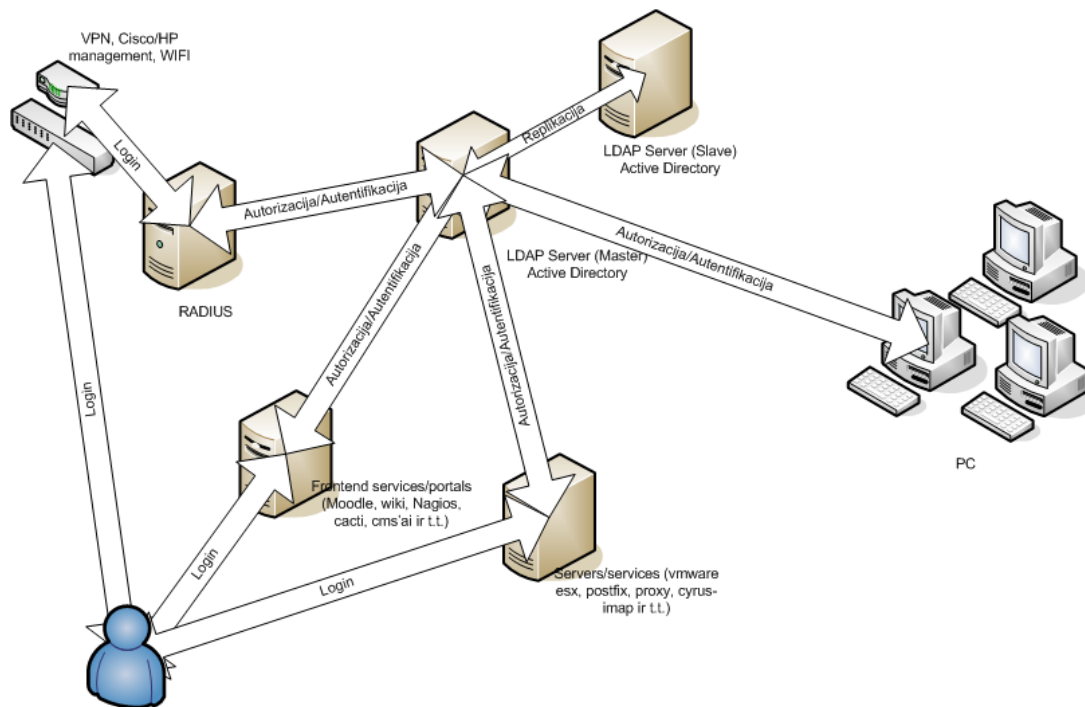
Vartotojui paranku turėti vieną identifikacinį duomenų rinkinį, kurio pagalba jis prisijungia prie visų teikiamų paslaugų ir visų institucijos bei jos partnerių siūlomų sistemų. Tokiu atveju patogų, kai institucija turi vieną aktyvų katalogą (angl. Active directory) ir visus vartotojų duomenis saugo vienoje vietoje, šiuo atveju LDAP (angl. „lightweight“ Directory Access Protocol) direktorijoje (alternatyva OpenLDAP). LDAP protokolas - supaprastintos kreipties į katalogus protokolas, skirtas prieiti prie katalogų paslaugų, pavyzdžiui, prie bendrovės adresų knygu, iš skirtingų operacinių sistemų [13].

LDAP katalogas - tai centralizuotas informacijos katalogas, saugantis įvairiai struktūrizuotą informaciją apie vartotojus, institucijas ir jų padalinius bei teikiamas paslaugas [11]. Naudojami standartizuoti prieigos protokolai, nesudėtinga integracija su kitomis sistemomis. LDAP kataloge yra galimybė atlikti vartotojų identifikavimą ir prieigos teisių kontrolę (authentication / autorization). Be to, tai yra saugomos informacijos prieigos kontrolė, galinti replikuoti su kitais katalogais (nepertraukiamo darbo užtikrinimas).

LDAP katalogas – tai vieninga, hierarchiškai organizuota, įvairiai struktūrizuotos informacijos, talpykla, kuri skirta informacijos apie įvairius objektus saugojimui (informacija apie asmenį, organizaciją, organizacijos padalinį, įrenginį ar paslaugą ir t.t.), vieningai vartotojų autentifikacijai atlikti (naudojami vienodi prisijungimo duomenys įvairiose sistemose), vieningai vartotojų autorizacijai.

LDAP direktorijoje saugoma informacija apie dėstytojus, darbuotojus, studentus ir laikinai į instituciją atvykusius (studentus, dėstytojus ir kt. darbuotojus) svečius. Naudojantis LDAP, sukuriamas vieningas identifikavimo mechanizmas visoms informacijos sistemomis bei aplinkoms. Pvz. Vilniaus kolegija vieningą autentifikavimo mechanizmą (18 paveikslas) naudoja Cisco VPN,

bevielio ryšio, prisijungimams prie kompiuterių, stebėjimo sistemose Nagios ir Cacti, e. mokymosi aplinkai Moodle, įrangos valdymui Cisco/HP management, vidiniam tinklui VIKO intranet, Wiki, elektroniniam paštui (darbuotojų ESX ir studentų postfix), Proxy paslaugai, elektroninis paštas per internetą OWA.

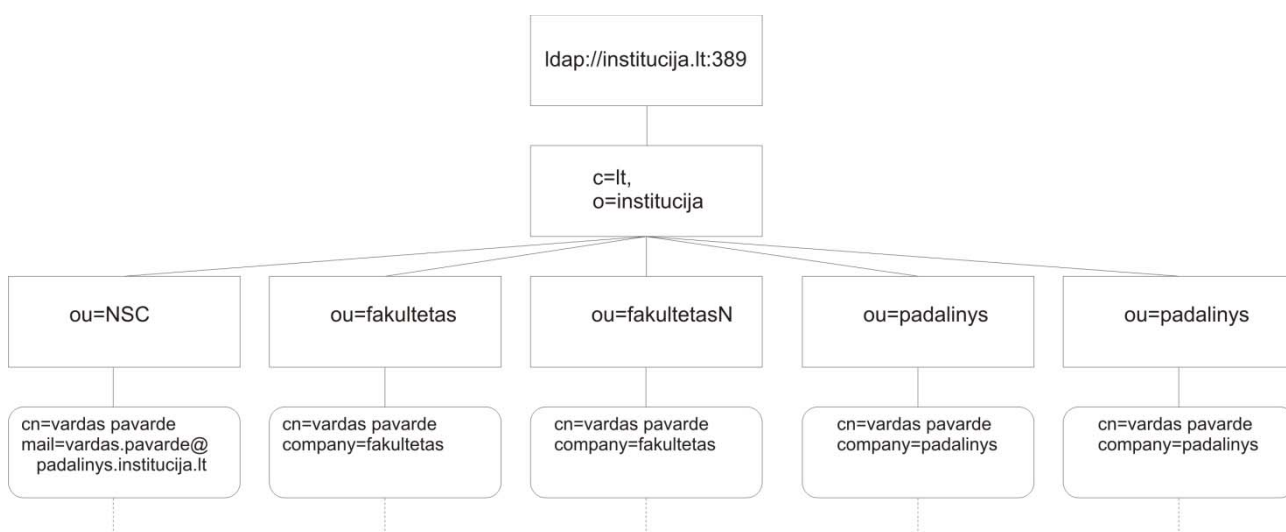


18 paveikslas. Vilniaus kolegijos autentifikacijos, naudojant LDAP, schema

Jeigu institucijoje buvo naudojama ne viena, o keli LDAP katalogai (pvz. institucijos padaliniai turėjo skirtingus katalogus), tai juos galima nesudėtingai apjungti panaudojant vieną schemą ir nustatant, viena – pagrindinį - LDAP katalogą. Tokiu būdu, duomenų apie vartotojus apskaitimo problema gali būti išspręsta negriaunant veikiančių sistemų, o panaudojant jose sukauptus duomenis. Aišku, tokiu atveju tolesni veiksmai būtų optimizavimas iki vieno katalogo. Tačiau šis atvejis turi ir tam tikrą pavojų. Sistemos darbo sutrikimo atveju prie centralizuotų duomenų prieigos neturėsime, būtina dubliuojanti sistema, saugos instrumentai, nes neteisėta prieiga prie tokios duomenų bazės atvers vartotojų duomenis. Bet to galima išvengti, jei pasirūpinsime dubliuojančiu katalogu ir tinkamai naudosisime saugos instrumentus. Kita problema yra ta, kad reikia detalizuoti poreikius, kokią informaciją būtina saugoti LDAP kataloge (įvertinant visų naudojamų informacijos sistemų poreikius). Trūkumas yra patogių įrankių, skirtų informacijos, esančios LDAP kataloge, valdymui nebuvimas.

LDAP katalogą apibrėžiantys modeliai [39]:

- Informacijos (angl. Information). Informacijos struktūros kurios saugomos LDAP kataloge.
- Įvardinimo (angl. Naming) - kaip informacija yra išdėstyta (angl. organized) ir identifikuojama LDAP kataloge.
- Funkcionalumo (angl. Functional) - kokios operacijos ar veiksmai gali būti atliekami su informacija, kuri saugoma LDAP kataloge.
- Saugumo (angl. Security) - kaip informacija, kuri saugoma LDAP kataloge, yra apsaugota nuo nesankcionuotos prieigos.



19 paveikslas. Galima LDAP schema

Galimi LDAP schemas atributai:

- displayName
- employeeNumber
- givenName
- homePhone
- homePostalAddress
- jpegPhoto
- mail
- mailLocalAddress
- mailRoutingAddress
- homeDirectory
- loginShell
- gecos
- mobile
- o
- uid
- userCertificate
- preferredLanguage
- userSMIMECertificate
- sn
- cn
- userPassword
- telephoneNumber
- ou
- street
- l
- eduPersonAffiliation
- eduPersonOrgDN
- eduPersonOrgUnitDN
- eduPersonPrimaryAffiliation
- eduPersonPrincipalName
- eduPersonEntitlement
- eduPersonPrimaryOrgUnitDN
- eduPersonScopedAffiliation

- uidNumber
- gidNumber
- postalCode
- postalAddress

Įrašo fragmentas:

```
CN=VardasPavarde,employeeNumber=50612,givenName=Vardas,homePhone=37052121212,mail=vardas.pavarde@institucija.lt,O=Institucija,CN=server.institucija.lt,L=Vilnius,OU=Fakultetas
```

Pateikiamų atributų sąrašą galima plėsti ir siaurinti pagal poreikius. Nurodytieji daugiau apibrėžia vartotoją bei informaciją apie jį.

LDAP katalogo naudojimas supaprastina vartotojų autentifikavimą, lengviau valdyti vartotojų duomenis, nes tai atliekama vienoje vietoje.

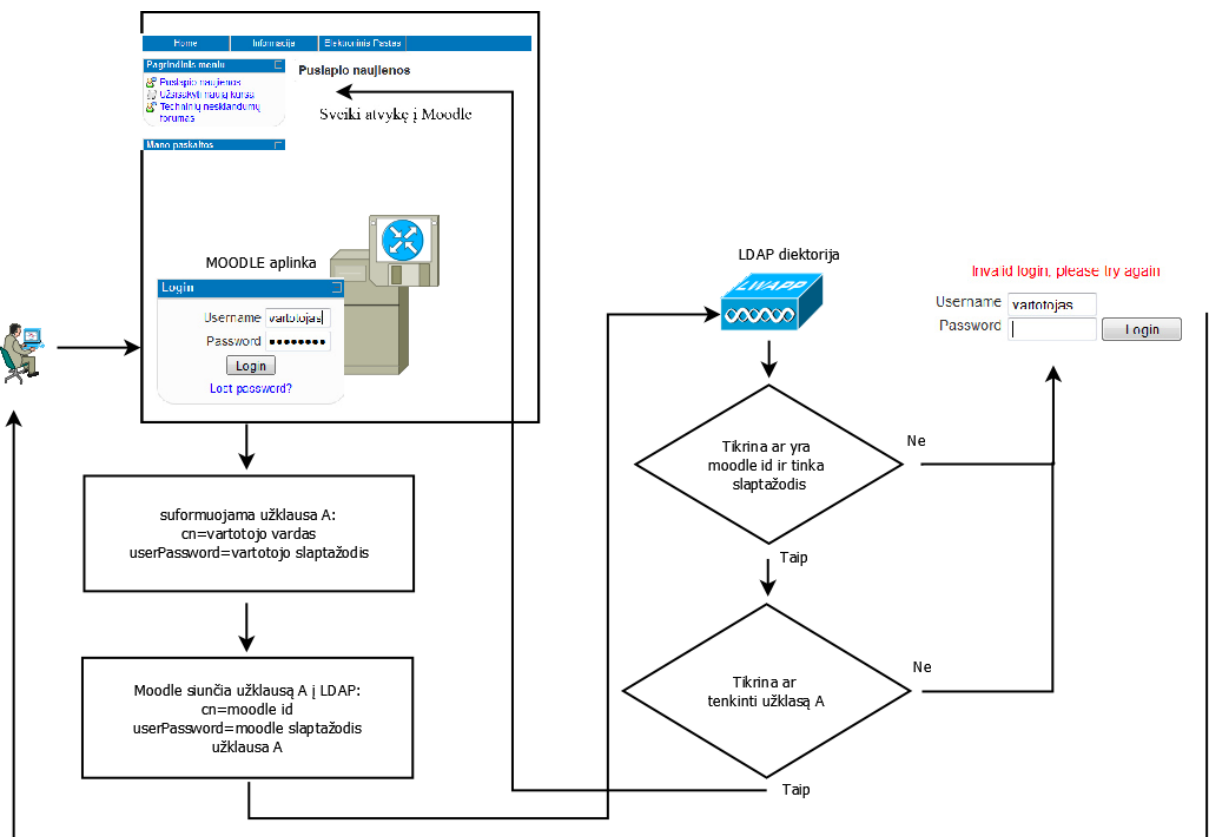
Pažiūrėkime, kaip vyksta autentifikacijos procesas e. mokymosi aplinkoje. Autorizacija, jungiantis į e. mokymosi aplinką (Moodle atvejis) naudojantis LDAP katalogą.

Pirmas variantas (20 paveikslas):

1. LDAP kataloge sukuriamas vartotojas Moodle aplinkos prisijungimui.
2. Moodle aplinkoje sukonfigūruojama nurodant, kad autorizacija vyksta per LDAP katalogą. Moodle aplinkoje galima pasinaudoti php-ldap moduliu.
3. Moodle aplinka naudoja savo identifikatorius. Šiuo atveju vartotojas jungiasi prie e. mokymosi aplinkos Moodle naudojant įprastą prisijungimo skydelį. Įvedamas vartotojo identifikatorius ir slaptažodis, tada Moodle aplinka jungiasi į LDAP katalogą naudojant savo vartotoją ir tikrina ar yra LDAP kataloge vartotojas, kuris siunčia prisijungimo užklausą. Jei toks vartotojas yra užregistruotas, gražinama autorizacija ir prijungiama prie aplinkos. Jei ne – užklausa netenkinama, atmetama.

Antras variantas:

1. LDAP kataloge sukuriamas vartotojas Moodle aplinkos prisijungimui.
2. Moodle aplinkoje sukonfigūruojama nurodant, kad autorizacija vyksta per LDAP katalogą.
3. Moodle aplinka naudoja vartotojo identifikatorius. Šiuo atveju vartotojas jungiasi prie e. mokymosi aplinkos Moodle naudojant įprastą prisijungimo skydelį. Įvedamas vartotojo identifikatorius ir slaptažodis, tada Moodle aplinka jungiasi į LDAP katalogą naudojant vartotojo identifikavimo duomenis: vardą ir slaptažodį. LDAP katalogas, tikrina ar yra toks vartotojas, kuris siunčia prisijungimo užklausą. Jei toks vartotojas yra tai prijungiama prie LDAP katalogo ir gražinama autorizacijos patvirtinimas ir prijungiama prie aplinkos. Jei ne – užklausa netenkinama, atmetama.



20 paveikslas. E. mokymosi (Moodle) aplinkos autorizacija per LDAP katalogą (1 variantas)

Antras variantas yra saugesnis, nes LDAP kataloge vartotojas gali prisijungti tik prie savo duomenų, su Moodle aplinkos identifikatoriais galima prieiga prie visos vartotojų duomenų bazės. Tačiau dažniausiai naudojamas pirmas variantas. Todėl, kad tai yra pradinis Moodle programinės įrangos nustatymas.

Vartotojų registravimosi funkcijų integracija gali būti atliekama bet kuriuo informacijos sistemų integracijos metu. Gali būti atliekamas ir nevykstant informacijos sistemų integracijai.

Naudojantis LDAP katalogu atsiranda galimybė įdiegti vieno prisijungimo metodą. Vieno prisijungimo metodas leistų vartotojui jungtis prie teikiamų paslaugų įvedus identifikacinių duomenų rinkinį vieną kartą.

4.4.2. SSO galimybė – vieno prisiregistravimo metodas

SSO (angl. Single Sign-On) – tai sprendimų visuma, užtikrinanti galimybę vartotojui vieną kartą save identifikavus, naudotis visomis teikiamomis paslaugomis, išvengiant pakartotinio

identifikavimo (jei vartotojui suteiktos teisės naudotis tomis paslaugomis) [23]. SSO – vieno prisiregistravimo metodas.

Vieno prisiregistravimo metodo atveju, iš sistemos reikalaujama surinkta visa vartotojo identifikavimo informacija, kurios reikia vartotojo tapatybės atpažinimui kiekvienoje antrinėje aplinkoje, su kuria vartotojas norės sąveikauti. Vartotojo įvestą tapatybės informaciją naudoja SSO paslaugos, esančios pirminėje aplinkoje tam, kad būtų palaikomas tiesioginio vartotojo tapatybės atpažinimas kiekvienoje antrinėje aplinkoje, su kuria vartotojas siekia iš tiesų sąveikauti.

Informacija, kurią tiesioginis vartotojas teikia kaip dalį pirminės aplinkos prisiregistravimo procedūros, gali būti naudojama antrinės aplinkos prisiregistravimui keliais būdais [39]:

- Tiesiogiai - vartotojo pateikta informacija yra perduodama antrinei aplinkai kaip antrinio registravimosi dalis.

- Netiesiogiai - vartotojo pateikta informacija yra naudojama tam, kad būtų surandama kita vartotojo tapatybės atpažinimo informacija, saugoma vieno prisiregistravimo informacijos valdymo bazėje. Surasta informacija po to naudojama kaip antrinės aplinkos prisiregistravimo operacijos pagrindas.

- Nedelsiant - nustatant laiką antrinėje aplinkoje kaip dalį pradinės sesijos nustatymo, t.y. vartotojai automatiškai prijungiami, nors komunikacija vyksta pirminio prisiregistravimo operacijos metu.

- Laikina saugoma arba talpinama ir naudojama tiesioginiam vartotojui pateikus užklausą antrinės aplinkos paslaugoms.

Žiūrint iš valdymo perspektyvos, vieno prisiregistravimo modelis sudaro vieno vartotojo sąskaitos valdymo sąsają, kurios dėka galima valdyti visas sudedamąsias sritis koordinuotu ir suderintu būdu.

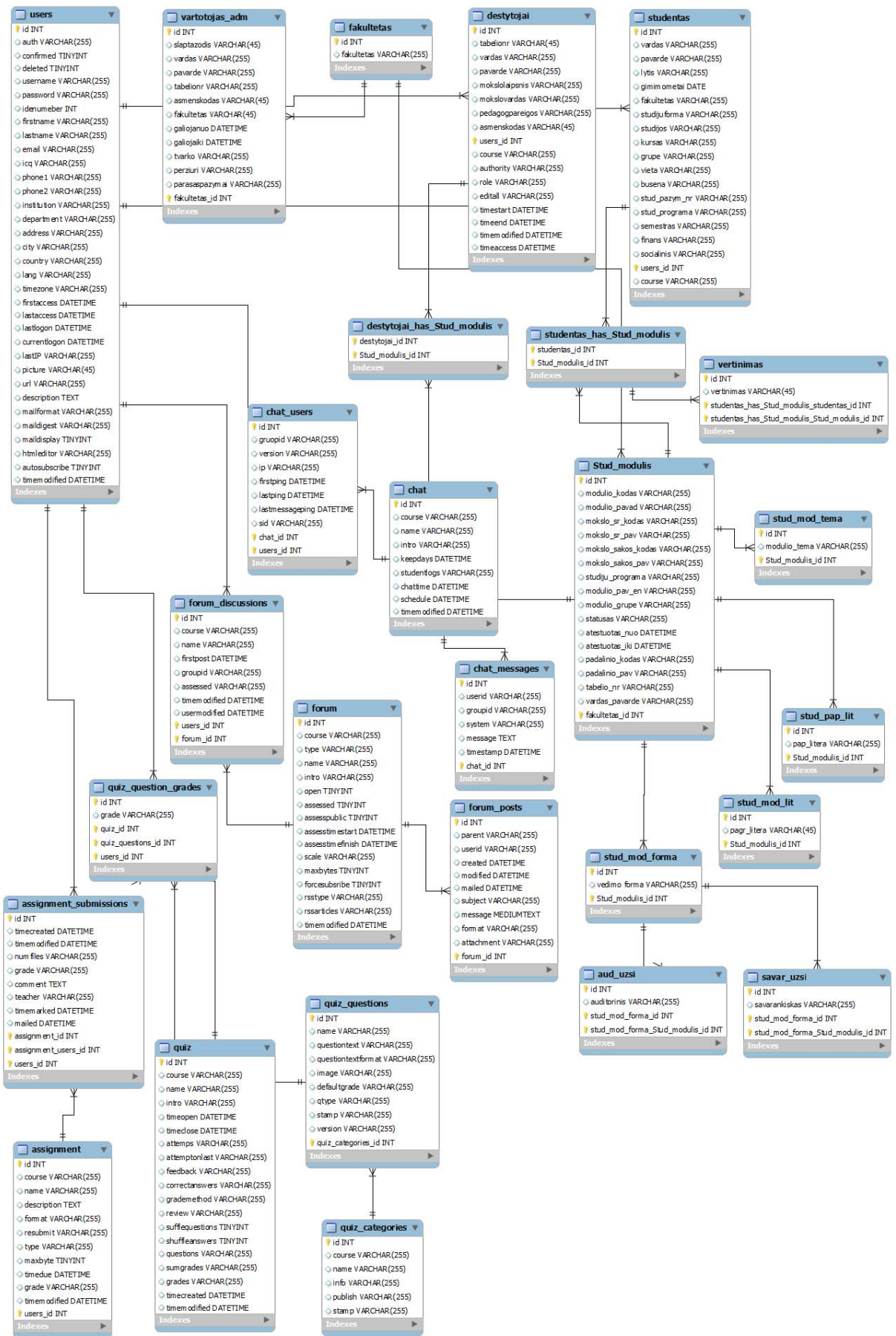
Reikšmingi SSO metodo saugumo aspektai yra tokie, kad antrinės aplinkos turi pasitikėti pirmine sistema, teisingai nurodant vartotojo tapatybę ir tapatybės nustatytas prieigos teises. Tapatybės nustatymo teisės turi būti apsaugotos nuo grėsmių, kylančių dėl perėmimo, galinčio baigtis netikrų vartotojų prisijungimu, kai yra perduodamos iš pirminės į antrinę aplinką.

Norint pasinaudoti SSO mechanizmu, reikalingi keli paruošiamieji darbai tiek aktyvaus katalogo serveryje, tiek informacijos sistemos serveryje. Aktyvaus katalogo serveryje reikia įdiegti Kerberos protokolo (Kerberos protokolas - pramoninio standarto tinklo autentifikavimo protokolas, kurio pagalba atliekamas abipusis autentifikavimas) palaikymą ir tinkamai aprašyti informacijos sistemos adresą. Tada, Kerberos protokolo pagalba, sukuriamas saugus raktas, kuris bus naudojamas autentifikuojant vartotojus. Informacijos sistemos serveryje nurodoma, kad vartotojų autentifikavimui naudotų Kerberos protokolą, nurodant nukopijuotą aktyvaus katalogo informacijas sistemos prieigos zonos saugų raktą. Paprastai dar nurodoma institucijos domeno sritis.

Kai vartotojas su savo identifikaciniais duomenimis jungiasi prie kompiuterio, kuris yra registruotas aktyviame kataloge, jam yra sukuriamas naujas saugus raktas, į kurį įeina ir aktyvaus katalogo serverio saugus raktas. Kai jis jungiasi į informacijos sistemą per naršyklę, ši mato, kad bus prašoma autorizacijos ir tikrina, ar vartotojas turi saugų raktą. Pamačiusi saugų raktą informacijos sistema sugeneruoja užklausą aktyviam katalogui, sugeneruodamas naują saugos raktą įtraukdamas savo ir vartotojo saugius raktus, kuriais klausiama, ar vartotoją galima autorizuoti. Aktyvus katalogas patikrina, ar tinkami saugos raktai, ar toks vartotojas užregistruotas ir turi prieigą prie informacijos sistemos. Atsakymą gražina informacijos sistemai ir vartotojas prisijungia prie informacijos sistemos, jei atsakymas teigiamas. Tokiu atveju viskas vyksta kodo lygmenyje, vartotojas nemato šių veiksmų. Tokiu būdu vartotojui nereikia įvedinėti identifikacinių duomenų rinkinio jungiantis į sistemas. Analogiškai jungiamasi ir prie kitų sistemų, jei tik yra naudojamas SSO metodas.

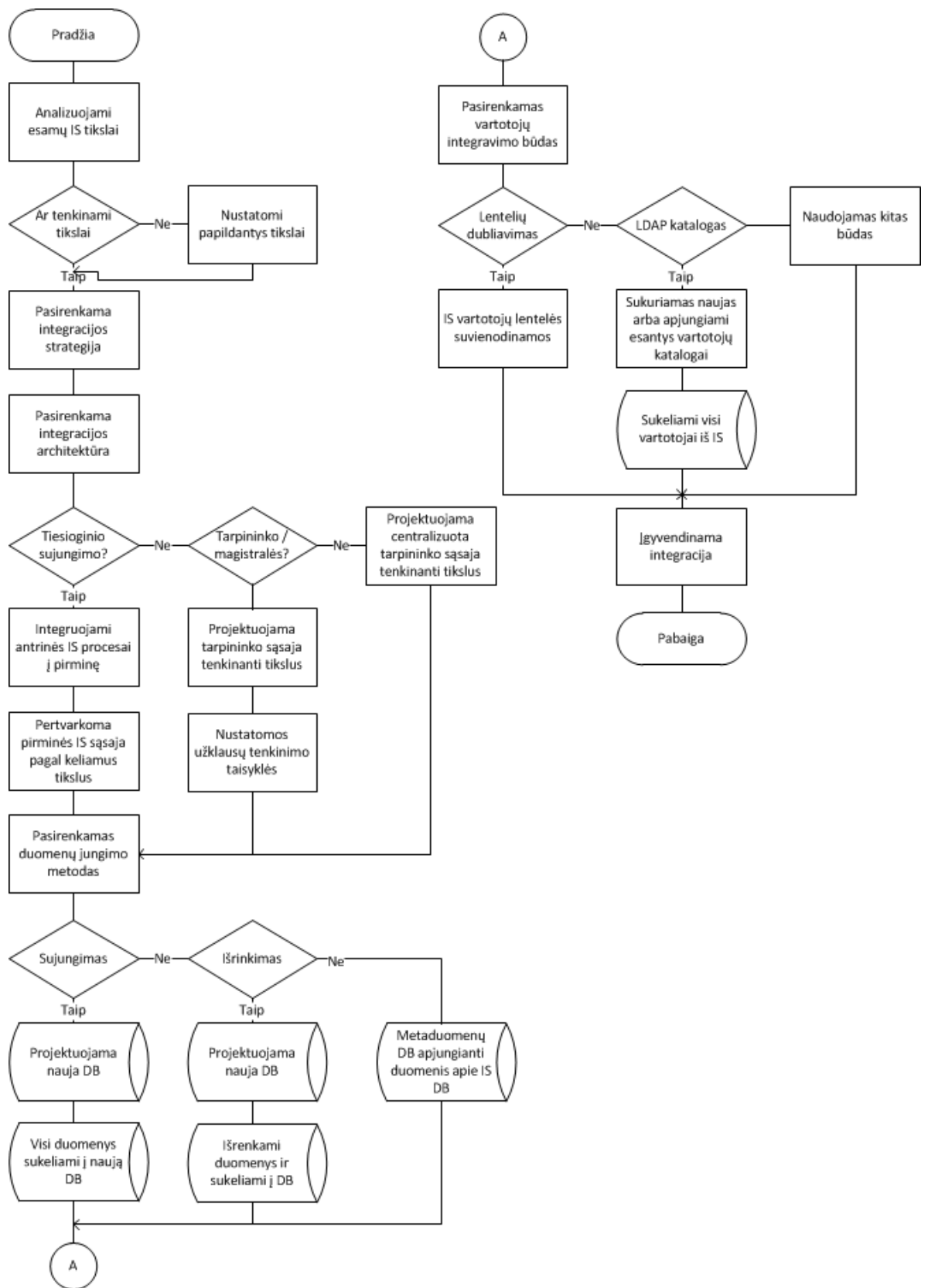
Tačiau šis metodas yra patogus, kai pradinis prisijungimas atliekamas prie įrenginių ar sistemų, kurie yra aktyvaus katalogo nariai (kompiuteriai, informacijos sistemos ir kt.). Ką daryti, jei vartotojas jungiasi iš namų kompiuterio ir nori naudotis integruota sistema. Tokiu atveju turi būti sukuriamas tarpininkas, pvz. centrinė atpažinimo sistema VU, kurio pagalba būtų sugeneruojamas saugus raktas arba išduodamas bilietas, einamajai sesijai. Jis galiojotų iki prisijungimo sesijos pabaigos.

Vartotojų valdymo funkcijų perdavimas naujai integruotai sistemai leidžia išvengti duomenų dubliavimo ir lengviau juos administruoti. Integruojant e. mokymosi aplinką ir akademinę informacijos sistemą, logiškai jas galima sujungti į vieną duomenų struktūrą, paliekant vieną vartotojų modulį. Integruotos sistemos duomenų struktūrą bei ryšius tarp duomenų galima pavaizduoti esybių ryšių diagrama, įtraukiant anksčiau naudotas pagrindines abiejų sistemų dalis. Remiantis dalykinės srities modeliais, vėliau gali būti braižomos detalios esybių ryšių diagramos, aprašančios fizinę sistemos duomenų bazės struktūrą. Tačiau atliekant integraciją naudojant federacinę architektūrą bus formuojama metaduomenų bazė, bet ne nauja duomenų bazė. Be to, tai nėra šio darbo uždavinys.



21 paveikslas. Esybių-ryšių diagrama sujungus e. mokymosi aplinką ir akademinę informacijos sistemą (esminiai komponentai)

4.5. Integravimo mechanizmo algoritmas



22 paveikslas. Integravimo mechanizmo algoritmas

Integravimo mechanizmo algoritmas apibendrina visus aukščiau aptartus integravimo procesus, suskirstant pagrindinius integravimo veiksmus į procesų blokus. E. mokymosi aplinkos ir akademinės informacijos sistemos integracijos mechanizmas yra toks:

1. Analizuojami informacijos sistemų tikslai. Svarbu identifikuoti tikslus, juos apibrėžti ir nusistatyti kokių rezultatų bus siekiama integruojant informacijos sistemas.
2. Pasirenkama informacijos sistemų integracijos strategija.
3. Pasirenkama informacijos sistemų integracijos architektūra.
4. Pasirenkamas duomenų jungimo metodas.
5. Pasirenkamas vartotojų registravimo funkcijų integravimo būdas.
6. Realizavus visus aukščiau paminėtus procesus, įgyvendinamas informacijos sistemų integravimas.

IŠVADOS

1. Išanalizavus e. mokymosi aplinkų naudojamą technologijų tipus, galima teigti, kad atviro kodo e. mokymosi aplinką daugiau galimybių integruoti į kitą informacijos sistemą dėl galimybės modifikuoti pradinį programinio paketo kodą.

2. Išanalizavus e. mokymosi aplinkose naudojamą technologijas ir architektūras galima teigti, kad asinchroninio tipo architektūros aplinka dėl nesudėtingo diegimo ir atviro kodo programinių paketų pasirinkimo yra paprasčiausia ir pigiausia e. mokymosi organizavimo forma, todėl plačiausiai naudojama.

3. Atlikus e. mokymosi aplinkų informacinių technologijų lyginamąją analizę, paaiškėjo, kad pasirinktos asinchroninės aplinkos funkciniu požiūriu yra beveik lygiavertės, turinčios panašias turinio valdymo sistemas ir įrankių rinkinius.

4. Akademinės informacijos sistemos Lietuvoje yra dar naujas reiškinys su besiformuojančia nauja patirtimi ir naujais sprendimais. Mažuma aukštųjų mokyklų turi visiškai funkcionuojančias akademinės informacinės sistemas, dauguma yra įdiegimo procese, todėl tik nedaugelis aukštųjų mokyklų yra suformulavusios informacijos sistemų tikslus ir uždavinius.

5. Išanalizavus pagrindinius akademinėse informacijos sistemose technologinius sprendimus, galima daryti išvadą, kad vyrauja karkaso tipo programiniai sprendimai, kurie dažniausiai realizuojami Oracle duomenų bazių valdymo sistemoje.

6. Esančios akademinės informacijos sistemos yra skirtingos informacijos pateikimo formomis ir jų gausa, vieningo informacijos pateikimo modelio nėra. Bandymai sukurti vieningą informacijos sistemą visoms akademinėms institucijomis patyrė nesėkmes (pvz. LiemSIS projektas).

7. Atlikta akademinėse informacijos sistemose vartotojų autentifikavimo mechanizmų analizė, atskleidė, kad jie yra pertekliniai, besidubliuojantys ir sunkiai valdomi ir administruojami.

8. Išanalizavus akademinės informacijos sistemas, darbe išskirti pagrindiniai informacijos sistemos veiklos funkcijos ir uždaviniai: apskaitos ir vartotojų sąskaitų valdymas, kontrolė, informacijos kaupimas, studijų planavimas ir vykdymas, komunikacija ir sklaida.

9. Integravimo strateginių sprendimų analizė patvirtino, kad mišrus integravimo būdas yra dažniausias, nes informacijos sistemos dažniausiai integruojamos ir dalykiniu, ir techniniais aspektais.

10. Galima išskirti tris architektūrinius integracijos sprendimus: tiesioginis integravimas, integravimas naudojant tarpininkus ir federacinį integravimą. Integracijos architektūrinį sprendimą lemia informacijos sistemų organizavimo sprendimų visuma.

11. Išanalizavus integracijai atlikti duomenų lygmenyje taikomus tris pagrindinius metodus, galima teigti, kad duomenų federacijos metodas yra saugiausias duomenų atžvilgiu, paprasčiausias ir reikalaujantis mažiau sąnaudų.

12. Galima išskirti tris programinių paketų integravimo lygmenis: duomenų integracija, pranešimų integracija bei procesų integracija. Integracijos lygmens pasirinkimą lemia organizacijos poreikiai ir finansinės galimybės.

13. E. mokymosi aplinkos ir akademinės informacijos sistemos integracija leistų sukurti kokybinę studijų valdymo, administravimo ir studijų tikslų įgyvendinimo aplinką.

14. Federacinis integracijos architektūrinis sprendimas ir duomenų federacijos metodo taikymas sukuria draugiškos aplinkos įvaizdį, kuri gali integruoti ir kitas informacijos sistemas bei kitas paslaugas. Tai galėtų būti prenumeruojamos duomenų bazės, bibliotekos paslaugos, skaitmeninių dokumentų duomenų bazės ir kt.

15. Vartotojų registravimosi funkcijų integravimas leidžia išvengti vartotojų duomenų dubliavimo ir pertekliškumo, palengvina valdymą, o vieno prisiregistravimo metodo naudojimas palengvina vartotojo darbą skirtingose informacijos sistemose ir programinių paketų aplinkose.

16. Vadovaujantis šiame darbe analizuotais integracijos procesais, apibrėžtas integravimo mechanizmas ir sudarytas jo algoritmas. Šis algoritmas suteiks galimybę nuosekliai, kryptingai ir racionaliai atlikti darbus prieš integracijos procesų realizavimą.

Integration of e-learning environments into academic information systems

Vaidas Liubinas

Summary

Examining information systems effective marketing issues in academic medium, one of the main problems becomes interaction of different information systems, data backup, redundancy and so on. Therefore integration of e-learning environment and academic information system is becoming more and more relevant enabling a qualitative study management, administration and study objectives implementation environment.

The subject of the paper is academic information systems. The purpose of the paper is to design and develop the algorithm (mechanism) of academic information systems and e-learning environment integration processes. The goals of the paper are to review and analyse technologies and architecture used in e-learning environment, review and analyse the key technological solutions of academic information systems, analyse possible integration ways and present the strategy and steps of e-learning environments and academic information systems integration process, the description of used components and technologies defining the integration mechanism.

In this paper, the conclusion that the asynchronous system environment due to ease of installation and the choice of open source software packages with more opportunities to integrate into other information systems is the most common form of organizing e-learning is made, after analyzing e-learning environments, used technologies and architectures.

The analysis of used or installed academic information systems in Lithuanian higher schools revealed the main information system operational functions and tasks to be monitoring and management of user accounts, control, information gathering, study planning and execution, communication and dissemination.

Strategic decision analysis of integration confirmed that the mixed integration method is the most common because most information systems are integrated by thematic and technical aspects. Architectural decision of integration depends on the entire decision of information systems implementation. The analysis of three basic methods at the level of data integration, assumed that the data federation method is the safest in respect of data, the simplest and requires less expenses. The choice of level of integration depends on the organization's needs and financial possibilities.

Federation integration architectural decision and federation data approach creates an image of friendly environment which can be integrated into other information systems and services. Users' sign on function integration allows avoiding data duplication and redundancy, easy management, and Single Sign-On method facilitates the user's work in different systems and environments.

According to the analyzed integration processes in this paper the integration mechanism is defined and the algorithm is designed. This algorithm will enable a coherent, focused and rational work before realization of the integration processes.

BIBLIOGRAFINIŲ NUORODŲ SARAŠAS

1. BAGUŠYTĖ, L. LUPEIKIENĖ, A. Verslo ir informacinių sistemų integravimas: architektūrinis aspektas. Informacijos mokslai, 2007, 42-43.
2. BELYK, Diane, et al. Classification of DE delivery systems [interaktyvus]. [Žiūrėta 2011-02-04]. Prieiga per internetą: http://www.immagic.com/eLibrary/ARCHIVES/GENERAL/ATHAB_CA/A020100B.pdf
3. BERNUS, P., MERTINS, K., SCHMIDT, G. Handbook on Architectures of Information Systems. Springer-Verlag, 1998.
4. BIDGOLI, H., et al. Encyclopedia of Information Systems. Elsevier Science, 2003.
5. Blackboard [interaktyvus]. Blackboard Inc, 1997-2011 [Žiūrėta 2010-10-14]. Prieiga per internetą: <http://www.blackboard.com/>
6. Blended Learning [interaktyvus]. Cognitive Design Solutions, 2003. [Žiūrėta 2010-10-09]. Prieiga per internetą: <http://www.cognitivedesignsolutions.com/DeliverySystem/DeliverySystem1.htm#Blended>
7. CHERIAN, M. S. A Semantic Data Federation Engine: Design, Implementation, and Applications in Educational Information Management [interaktyvus]. [Žiūrėta 2011-02-22]. Prieiga per internetą: <http://dig.csail.mit.edu/2010/cherian-thesis/main.pdf>
8. CUMMINS, F. A. Enterprise Integration. An Approach for Enterprise Application and Systems Integration. John Wiley & Sons, 2002.
9. DAVYDOV M. M. Corporate Portals and e-Business integration, USA : McGraw-Hill, 2001. 305 p. ISBN 0-07-138279-8.
10. Desire2Learn : Innovative Learning Technology [interaktyvus]. Desire2Learn Incorporated, 1999-2011. [Žiūrėta 2010-10-22]. Prieiga per internetą: <http://www.desire2learn.com/>
11. DONLEY, Clayton. LDAP Programming, Management and Integration. Manning Publications Co, 2003, p. 326. ISBN 1-930110-40-5
12. eLecta Live [interaktyvus]. Electa communication LTD, 2011. [Žiūrėta 2011-02-17]. Prieiga per internetą: <http://www.e-lecta.com/>
13. Enciklopedinis kompiuterijos žodynas [interaktyvus]. Informacinės visuomenės plėtros komitetas prie Lietuvos Respublikos vyriausybės, Matematikos ir informatikos institutas, 2001-2010. [Žiūrėta 2010-12-14]. Prieiga per internetą: <http://www.likit.lt/term/enciklo.html>
14. E-learning learning [interaktyvus]. Aggregage, 2011. [Žiūrėta 2011-01-18]. Prieiga per internetą: <http://www.elearninglearning.com/>

15. GEYSERMANS, Fransis, MILLER, Jeff. Build asynchronous applications with the Distributed Event-Based Architecture for Web Services [interaktyvus]. IBM, 2003. [Žiūrėta 2011-01-13]. Prieiga per internetą: <http://www.ibm.com/developerworks/webservices/library/ws-dbarch/>
16. GIEDRIMAS, V. Modelinės architektūros naudojimas kuriant komponentines programų sistemas. Informacijos mokslai, t. 50. Vilniaus universiteto leidykla, 2009. ISSN 1392-0561
17. HANSEN, Dain. Demystifying Data Federation for SOA [interaktyvus]. The SOA Magazine. [Žiūrėta 2011-03-03]. Prieiga per internetą: <http://www.soamag.com/I22/0908-1.php>
18. Inmedius Learning [interaktyvus]. Inmedius Inc, 1995-2011. [Žiūrėta 2010-11-15]. Prieiga per internetą: <http://www.gen21.com/>
19. Integrity eLearning [interaktyvus]. Integrity eLearning, 1997-2011. [Žiūrėta 2010-11-26]. Prieiga per internetą: <http://www.ielearning.com/>
20. YIN, Xuepeng, PEDERSEN, Torben Bach. The OLAP-XML Federation System [interaktyvus]. [Žiūrėta 2011-02-17]. Prieiga per internetą: <http://www.cse.iitb.ac.in/comad/2006/proceedings/222.pdf>
21. YONG, Jianming, YAN, Jun, HUANG, Xiaodi. WFMS-based Data Integration for e-Learning [interaktyvus]. USQ, 2009. [Žiūrėta 2010-11-19]. Prieiga per internetą: <http://eprints.usq.edu.au/5697/>
22. JASUTIENĖ, E., MARKAUSKAITĖ, L. Atviro kodo VMT bendrojo lavinimo galimybių tyrimas. Lietuvos matematikos rinkinys. T. 44(2004), p. 281–286. ISSN 0132-2818.
23. JEONG, Jongil, et al. Java-Based Single Sign-On Kibrary Supporting SAMPL (Security Assertion Markup Language) for Distributed Web Services // Advanced Web Technologies and Applications. 6th Asia-Pacific Conference, APWeb 2004. XU YU, Jeffrey, et al. p. 957. Springer-Verlag Berlin Heidelberg, 2004. ISBN 3-540-21371-6 Springer-Verlag Berlin Heidelberg New York
24. KRAUJUTAITYTĖ, L., PLEČKAITIS, J.S. Nuotolinių studijų organizavimas: strategijos ir technologijos. Monografija. Vilnius, 2003. 290 p. ISBN 9955-563-31-1.
25. KULVIETIENĖ, R., STANKEVIČ J. Blended e-Learning Delivery Using the LearningSpace – Virtual Classroom // Advanced Learning Technologies and Applications – 2003: Proceedings of the International Conference [Kaunas, 2003 m.]. Kaunas, 2003, p. 89–92. ISBN 9955-09-485-0.
26. LaunchForce [interaktyvus]. LaunchForce, 2010. [Žiūrėta 2010-11-01]. Prieiga per internetą: <http://launchforce.com/>
27. LearnLink : Digital Tools for Development [interaktyvus]. AED, 2003. [Žiūrėta 2010-11-30]. Prieiga per internetą: <http://learnlink.aed.org/>

28. LearningSpace [interaktyvus]. University College Falmouth [Žiūrėta 2010-11-30]. Prieiga per internetą: <http://learningspace.falmouth.ac.uk/>
29. LearningSpace – Virtual Classroom [interaktyvus]. IBM, 2003. [Žiūrėta 2010-11-23]. Prieiga per internetą: http://www.ibm.com/developerworks/lotus/library/ls-quickcourse_LVC/index.html
30. Lietuvos mokslo ir studijų informacijos sistemos akademinės dalies sukūrimas ir bandomasis įdiegimas [interaktyvus]. [Žiūrėta 2010-12-10]. Prieiga per internetą: <http://www.esparama.lt/lt/bpd/zemelapis/?id=4411>
31. LIU, Yan, LIANG, Xin, et al. Using Architecture Integration Patterns to Compose Enterprise Mashups [interaktyvus]. NICTA, 2009-2011. [Žiūrėta 2011-01-19]. Prieiga per internetą: http://www.nicta.com.au/research/research_publications?sq_content_src=%2BdXJsPWh0dHBzJTNBJTJGJTJGcHVibGljYXRpb25zLmluc2lkZS5uaWN0YS5jb20uYXUIMkZzZWZyY2gIMkZmdWxsdGV4dCUzRmlkJTNEMTk2NiZhbGw9MQ%3D%3D
32. Lotus Learning Management System [interaktyvus]. IBM. [Žiūrėta 2010-10-16]. Prieiga per internetą: <http://www-01.ibm.com/software/lotus/products/learning-management-system/>
33. MACKENZIE, Duncan. Architectural Options for Asynchronous Workflow [interaktyvus]. Microsoft, 2001. [Žiūrėta 2011-01-13]. Prieiga per internetą: <http://msdn.microsoft.com/en-us/library/ms978405.aspx>
34. MILNE, Andrew. Designing Blended Learning Space to the Student Experience [interaktyvus]. Educause, 1999 – 2011. [Žiūrėta 2010-11-04]. Prieiga per internetą: <http://www.educause.edu/learningspacesch11>
35. Object Management Group [interaktyvus]. OMG Inc., 1997-2011. [Žiūrėta 2010-09-19]. Prieiga per internetą: <http://www.omg.org/index.htm>
36. PICCOLI, G.; AHMAD, R.; IVES, B. (2001). Web-Based Virtual Learning Environments: A Research Framework and a Preliminary Assessment of Effectiveness in Basic It Skills Training. MIS Quarterly, 0276-7783. December 1, vol. 25, issue 4.
37. PEARLSON, K. E., SAUNDERS, C.. Managing and Using Information Systems: a Strategic Approach. John Wiley & Sons, Inc., 1999.
38. POGGI, Antonella, RUZZI, Marco. Filling the gap between data federation and data integration. [interaktyvus]. [Žiūrėta 2011-02-14]. Prieiga per internetą: <http://www.dis.uniroma1.it/~poggi/publi/SEBD2004.pdf>
39. REIMER, S., et al. With Microsoft Active Directory Team. Windows Server 2008: Active Directory. Resource Kit. Microsoft Press, 2008, p. 827. ISBN 978-0-7356-2515-0

40. RESTA, P., LAFERRIERE, T. Technology in support of collaborative learning [interaktyvus]. Educational Psychology Review, 19, 2007, p. 65-83. [žiūrėta 2011-01-31]. Prieiga per internetą: <http://jessicawicks.files.wordpress.com/2010/07/reastalaferriere.pdf>
41. RUTKAUSKIENĖ, D. TARGAMADŽĖ, A. et al. Nuotolinis mokymasis Kaunas: Technologija, 2003. 255 p. ISBN 9955-09-321-8.
42. Saba People Cloud [interaktyvus]. Saba software, 2011. [žiūrėta 2011-01-23]. Prieiga per internetą: <http://www.saba.com/>
43. SEKLIUCKIS V., GUDAS S., GARŠVA G. Informacijos sistemos ir duomenų bazės: Vadovėlis. Kaunas: Technologija, 2003. - 338 p
44. SEROTER, Richard, et al. Applied Architecture Patterns on the Microsoft Platform. Packt Publishing, 2010, p. 518. ISBN 978-1-849680-54-7
45. Server and Site Architecture Overview [interaktyvus]. Microsoft Corporation, 2011. [Žiūrėta 2011-02-04]. Prieiga per internetą: <http://msdn.microsoft.com/en-us/library/ms980916>
46. Student Academic Information Systems [interaktyvus]. University of California, Riverside [Žiūrėta 2010-11-04]. Prieiga per internetą: <http://cnc.ucr.edu/sais/>
47. TARGAMADŽĖ, A. Virtualūs universitetai. IKT vadmuo mokymiesi Studijų kokybė [interaktyvus]. 2010. [Žiūrėta 2010-12-04]. Prieiga per internetą: <http://www.smpf.lt/get.php?f.2521>
48. The Chartered Institute of Personnel and Development [interaktyvus]. CIPD, London, 2011. [Žiūrėta 2010-09-09]. Prieiga per internetą: <http://www.cipd.co.uk/>
49. WBT Systems : Solutions to Power the Intelligent Enterprise [interaktyvus]. WBT Systems, 2011. [Žiūrėta 2010-09-09]. Prieiga per internetą: <http://www.wbt systems.com/>
50. WCET Edu Tools: EduTools Course Management System Comparisons [interaktyvus]. [Žiūrėta 2010-11-04]. Prieiga per internetą: <http://www.edutools.info/course/compare/index.jsp>
51. WILSON, B. G. (1996). Constructivist Learning Environments: Case Studies in Instructional Design [interaktyvus]. In Educational Technology Publications. NJ: Englewood Cliffs [Žiūrėta 2010-11-11]. Prieiga per internetą: <http://www.elearning-reviews.org/topics/pedagogy/learning-design/1996-wilson-constructivist-learning-environments-case-studies-instructional-design/>
52. Win Solution for Life-Learning [interaktyvus]. Worldwide Interactive Network, 2008. [Žiūrėta 2010-11-22]. Prieiga per internetą: <http://www.w-win.com/>