



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
MULTIMEDIJOS INŽINERIJOS KATEDRA

Dalia Mankauskaitė

E. mokymosi programinio produkto kūrimas

Magistro darbas

Darbo vadovas

dr. Sigitas Drąsutis

Kaunas, 2009



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
MULTIMEDIJOS INŽINERIJOS KATEDRA

Dalia Mankauskaitė

E. mokymosi programinio produkto kūrimas

Magistro darbas

Recenzentas

Doc. Dr. Regina Kulvietienė

2009-05-20

Vadovas

dr. Sigitas Drašutis

2009-05-25

Atliko

IFN-7/2 gr. stud. Dalia Mankauskaitė

2009-05-25

Kaunas, 2009

TURINYS

1. SUMMARY	6
2. ĮVADAS.....	7
2.1. Problemos aktualumas	7
2.2. Tyrimų sritis ir įrankiai	8
2.3. Darbo tikslas	9
2.4. Darbo uždaviniai:.....	9
2.5. Praktinė svarba.....	10
3. ANALITINĖ DALIS	11
3.1. Interaktyvumas ir jo kokybė	11
3.2. E. mokymosi Paskalyje terpių palyginimas.....	12
3.3. Programinių įrankių analizė.....	17
3.3.1. <i>Paskalis, programuotojas ir kompiuteris</i>	17
3.3.2. <i>HTML</i>	20
3.3.3. <i>DHTML</i>	21
3.3.4. <i>AJAX</i>	21
3.3.5. <i>XML</i>	22
3.3.6. <i>Firebird Server Manager</i>	22
3.3.7. <i>Morfik Fx For Object Pascal</i>	22
3.3.8. <i>Serverio ir kliento programos</i>	23
3.4. Išvados.....	24
4. INTERAKTYVIOS E. MOKYMOSI PASKALYJE TERPĖS MODELIO PROJEKTAVIMAS.....	25
4.1. Technologijų pasirinkimas.....	25
4.2. Formalus aprašymas	27
4.3. Vartotojo sąsaja.....	28
4.3.1. <i>Schematinis produkto veikimo planas:</i>	28
4.4. Specifikacija.....	29
4.5. Architektūra	34
4.6. Išvados	35
5. VARTOTOJO DOKUMENTACIJA	36
5.1. Svetainės realizacija.....	36
5.2. Vartotojo instrukcija	36
5.3. Išvados	38

6. SVETAINĖS TESTAVIMAS.....	39
7. IŠVADOS	40
8. LITERATŪRA.....	41
9. TERMINŲ IR SANTRUMPŲ ŽODYNAS.....	42
1 PRIEDAS. Programos Morfik FX for Object Pascal pradinis langas	43
2 PRIEDAS. Mokomosios dalies tikslo pateikimo langas.	44
3 PRIEDAS. Pretesto pateikimo langas.....	45
4 PRIEDAS. Mokomosios medžiagos pateikimo langas.....	46
5 PRIEDAS. Pavyzdžio pateikimo langas.....	47
6 PRIEDAS. Patikros pateikimo langas.	48
7 PRIEDAS. Žodynėlio pateikimo langas.	50

Lentelių sąrašas

1 lentelė E.mokymosi Paskalyje terpių palyginimas	17
2 lentelė Duomenų saugojimo technologijų palyginimas pagal kriterijus.....	25
3 lentelė DOM technologijų palyginimas pagal kriterijus.....	25
4 lentelė Duomenų bazių sistemų palyginimas.....	26
5 lentelė Programavimo technologijų palyginimas pagal kriterijus	26

Paveikslėlių sąrašas

1 pav. G. Grigas, elektroninė knyga „Programavimas Paskaliu“	12
2 pav. „Programavimas Paskaliu“ (10 kl.).....	13
3 pav. V. Žemaitis, „Programavimo pradmenys“	14
4 pav. Informatika. Nuotolinio mokymo sistema	14
5 pav. Pascal Programming	15
6 pav. Pascal Programming	16
7 pav. Transliatorius išverčia programą iš Paskalio kalbos į kompiuterio kalbą.....	18
8 pav. Kompiuteris atlieka į jo kalbą išverstą programą.....	18
9 pav. Programos transliavimas ir vykdymas. Transliavimo rezultatas virsta programa	19
10 pav. HTML dokumento užklausa	23
11 pav. Vartotojo ir sistemos sąveikos schema	27
12 pav. Paskalio programavimo kalbos interaktyvios mokymosi svetainės panaudos atvejų diagrama.....	28

13 pav. E. mokymosi Paskalyje produkto pradinis langas	29
14 pav. Klasių diagrama	30
15 pav. UML sekos diagrama	31
16 pav. Būsenų diagrama.....	32
17 pav. Testo lentelių sąryšiai.....	32
18 pav. Testo veikimo veiklos diagrama	33
19 pav. Programos architektūra	34
20 pav. Pagrindinis svetainės langas.....	36
21 pav. Pretesto rezultatų apibendrinimo langas	37
22 pav. Teisingai suvesto kodo langas.....	48
23 pav. Neteisingai suvesto kodo langas	49

1. SUMMARY

In order to help those who want to gain knowledge of programming it was decided to create the electronic product for teaching of basics of programming in Pascal programming language.

This paper is intended for those who want to learn the various structures of Pascal language and the characteristic features of Pascal programming environment used for creation of programs: the first-course students of the higher schools, the teachers of informatics, the pupils of secondary, professional and further schools, as well as for independent studies of basics of programming.

In this paper the standard structures of Pascal language and the main libraries (Dos, Crt, Graph) are supplied.

The big volume and more complicated chapters are divided into more detailed topics. The main parts are:

1. Theoretic material of the topic.
2. Demonstrative samples.
3. Test for self-control.
4. Practical task.
5. Vocabulary.

The practical part of electronic training product was made in Borland Pascal programming language. The work was created by means of Morfik XP for Object Pascal program.

2. ĮVADAS

Nuotolinėse studijose besimokantysis ir dėstytojas dažniausiai tiesiogiai nebendrauja, reikalinga paprasta galimybė perduoti informaciją dideliais atstumais. Besimokantysis turi turėti galimybę pasinaudoti mokymosi medžiaga jam prieinamu laiku norimoje vietoje. Medžiaga turi neužimti daug vietos kompiuterio laikmenose. Norint pagerinti besimokančiųjų mokymosi motyvaciją mokomoji medžiaga turi būti *interaktyvi*. Taigi, norint patenkinti šiuos reikalavimus, nuotoliniame mokyme mokymosi medžiagai pateikti plačiai naudojamos pasaulinio voratinklio galimybės. Hipertekstas leidžia mokymo medžiagą pateikti labai vaizdžiai, įterpiant grafikus, vaizdus, garso, vaizdo įrašus bei redaguojamuosius laukus. Hierarchinis medžiagos išdėstymas, išsišakojimai ir detalizacijos principas leidžia imituoti intelektualią mokymosi aplinką ir prisitaikyti prie individualių besimokančiųjų savybių bei skirtingo jų pasirengimo lygio. Iškilus neaiškumams, besimokantysis gali paprašyti pateikti detalesnę informaciją, kuri gali būti aktuali ne visiems, nes įvairiems besimokantiejiems gali būti neaiškios skirtingos vietos.

Nors WWW priemonės ir turi nemažai trūkumų, bet šiuo metu tai yra viena efektyviausių priemonių rengiant individualias nuotolines studijas, todėl jų naudojimas ateityje plėsis.

Šiuo metu internete galima rasti įvairios mokomosios medžiagos, tačiau daugelis jų priskiriama neaktyviam interaktyviam mokymuisi. Yra manoma, kad aktyvus interaktyvus mokymasis mokiniams lengviau įsisavinamas.

Dauguma pasaulinių informatikos olimpiadų dalyvių programoms rašyti renkasi Paskalio kalbą.

Paskalio žymenys vartojami algoritams užrašyti. Algoritmus skaito ir nagrinėja žmogus. Taigi Paskalio kalba skiriama ne tik žmogaus bendravimui su kompiuteriu, bet ir žmogaus (programuotojo) bendravimui su kitu žmogumi (programuotoju). Paskalis yra patogi priemonė algoritmavimo idėjoms ir metodams išreikšti, kad su jais galėtų susipažinti kiti bendraminčiai – programuotojai. Todėl į Paskalį reikia žiūrėti visų pirma kaip į algoritminių žymenų sistemą, skirtą žmogui. Informatikoje (konkrečiau algoritmavime ir programavime) jis atlieka analogišką vaidmenį kaip matematiniai žymenys matematikoje, chemijos formulių kalba chemijoje, natų žymenys muzikoje ir pan.

2.1. *Problemos aktualumas*

Lietuvoje mokiniai gali rinktis laikyti mokyklinį ar valstybinį informacinių

technologijų egzaminą.

Valentina Dagienė „Švietimo naujienos“ 2007 m. Nr. 11 teigia, kad „...užsienio šalys neturi privalomo informatikos mokymo, nemoko programavimo. Taip, daugumai šalių pradžioje to nereikėjo, jie buvo ekonomiškai stipresni, greičiau apsirūpino kompiuteriais, naudojo juos mokydami. O dabar ėmė sukti galvą, kaip geriau mokytį, kas būtų naudingiau. Štai dabar prisimena mus, turėjusius ir turinčius informatikos mokymo pagrindus. Šveicarai nori pasimokyti iš mūsų ir kitais metais įves informatikos kursą, kurio didžiuma bus skiriama programavimui mokytį, bus naudojamas „Paskalis““. [1]

Tuo metu Lietuvoje paskalis – viena pagrindinių programavimo kalbų, naudojamų programavimo pagrindų mokymuisi bendro lavinimo mokyklose bei aukštųjų mokyklų pirmuose kursuose. Tai nenurungiamą programavimo kalbą pradedantiems.

Todėl mokantis informatikos svarbus kiekvienas žingsnelis, jis turėtų būti pamatuotas, pagrįstas. Moksleivis, norėdamas išlaikyti informatikos egzaminą, turi žinoti ir suprasti pagrindines informacijos ir informatikos sąvokas, būti įgijęs pagrindinius darbo su informacija įgūdžius, kritiškai mąstyti, gebėti įvertinti informacijos ir komunikacijos technologijos laimėjimus bei jų taikymą visuomenėje, suvokti esminius darbo kompiuteriu principus, suprasti pagrindines algoritmavimo konstrukcijas ir mokėti spręsti nesudėtingus algoritmavimo uždavinius, gebėti pasinaudoti kompiuteriu algoritmavimo uždaviniams atlikti.

Lietuvoje XI-XII klasėms yra numatytas programavimo pradmenų modulis, kuriam dėstyti skirtos 34 valandos. Tai ir turėtų sudaryti programavimo egzaminą branduolį. Tačiau ar to tikrai pakaks norint gerai pasiruošti programavimo egzaminui? Vargu. Moksleiviams svarbu turėti praktinių programavimo bei darbo kompiuteriu įgūdžių. Ko gero, tik nedaugelis moksleivių gebės per minimalų šiam moduliui skirtų valandų skaičių (34 valandas) įgyti šių įgūdžių. Vadinasi, norint pasiruošti programavimo egzaminui, reikia papildomai pasirinkti išplėstinį informatikos kurso lygį arba mokytis privačiai. [2]

Norint savarankiškai mokytis programavimo pagrindų, galima skaityti internetinėse svetainėse pateiktą teorinę medžiagą. Elektroninėje G.Grigo „Programavimas Paskaliu“ knygoje galima rasti ne tik teoriją, bet ir savikontrolei skirtų uždavinių, tačiau išsamių pavyzdžių, savikontrolės testų bei galimybės tuo pačiu metu išbandyti savo sukurtą algoritmą nėra.

2.2. Tyrimų sritis ir įrankiai

Siekiant padėti mokiniams, nuspręsta sukurti elektroninį produktą programavimo pradmenų e. mokymuisi. Juk negalima išmokyti kurti programas tik skaitinėjančias knygas. Taip galima tik susipažinti su programavimo elementais ir juos suprasti. Išmokyti kurti programas

galima tik pačiam jas rašant ir derinant kompiuteriu. Nuotolinis mokymas gali būti didelė paskata ne tik gerai besimokantiems, bet ir visiems mokymosi motyvacijos stokojantiems vaikams.

Atvira interaktyvi svetainė bus skirta tiems, kurie norės susipažinti su įvairiomis Paskalio kalbos struktūromis ir programų rengimui skirtomis Paskalio programavimo aplinkos savybėmis: aukštųjų mokyklų pirmųjų kursų studentams, informatikos mokytojams, vidurinių, profesinių ir aukštesniųjų mokyklų moksleiviams. Be abejo ir savarankiškomis programavimo pagrindų studijoms.

Darbe pateikiamos standartinės Paskalio kalbos struktūros ir pagrindinės bibliotekos: Dos, Crt, Graph.

Didesnės apimties ir sudėtingesni skyriai suskaidyti į smulkesnes temas. Pagrindinės jų dalys:

1. Teorinė temos medžiaga
2. Demonstraciniai pavyzdžiai.
3. Savikontrolei skirtas testas
4. Praktinė užduotis
5. Žodynėlis

Praktinė e. mokymosi produkto kūrimo dalis atliekama Borland Pascal programavimo kalba. Darbas kuriamas Morfik XP For Object Pascal programa. [3]

2.3. Darbo tikslas

Darbo tikslas. Elektroninio produkto, (tam tikros elektroninės terpės, leidžiančios rašyti algoritmus) skirto Paskalio programavimo pagrindų e. mokymuisi sukūrimas.

2.4. Darbo uždaviniai:

1. Išanalizuoti literatūros šaltinius.
2. Suprojektuoti svetainę, atviram nuotoliniam programavimo pradmenų Paskalyje mokymuisi:
 - 2.1. Individualizuoti programavimo Paskalyje pradmenų mokymąsi;
 - 2.2. Sudaryti galimybę studijuoti savarankiškai.
3. Atlikti sukurtos svetainės testavimą.

Metodai:

1. Informacijos paieška, sisteminimas ir apibendrinimas;

2. Programinės įrangos, skirtos internetinių svetainių ir duomenų bazių kūrimui analizė ir parinkimas;

3. Programų diegimas ir panaudojimas atviros internetinės svetainės, skirtos programavimo pradmenų Paskalyje mokymuisi, kūrimui.

2.5. Praktinė svarba

Programavimo pagrindų e. mokymosi produktas palengvina, norinčių išmokti programuoti Paskalio programavimo kalboje, žinių įgijimą. Jame galima studijuoti mokomąją medžiagą, analizuoti pateiktus pavyzdžius, spręsti savikontrolės užduotis bei patikrinti sugalvoto algoritmo funkcionalumą.

3. ANALITINĖ DALIS

3.1. *Interaktyvumas ir jo kokybė*

Internetas suteikia daugiau laisvės. Skaitytojas gali rinktis tuos informacijos šaltinius, kurie jam svarbūs ir įdomūs. Autorius taip pat laisvas. Elektroninis leidinys leidžia įtraukti įvairius požiūrius, įvairius informacijos paieškos metodus, derinti multimedijos informaciją ir pan.

Interaktyvumas – tai viena dažniausiai minimų naujosios žiniasklaidos savybių. Būtent interaktyvumas leidžia subjektyviai peržiūrėti kompiuterinę informaciją. Kita vertus, pats interaktyvumas nėra vienpusiška ypatybė. Interaktyvumas gali skirtis savo kokybe. Kitaip sakant, interaktyvumas gali būti laipsniuojamas [4].

Daugelyje elektroninių kūrinių interaktyvumas yra tik menamas, tai yra, sukuriama tik jo iliuzija. Dažnai skaitytojui leidžiama rinktis tik iš labai nedidelio skaičiaus alternatyvų, kitaip sakant, skaitytojo pasirinkimo erdvė yra griežtai apribojama.

Šis paprastasis interaktyvumas patiriamas kasdien, ne tik kompiuterinėje aplinkoje (pavyzdžiui, žiūrovas gali skambinti į televizijos stotį ir rinktis pageidaujamą vakaro filmą – tam tikra prasme žmogus turi galimybę rinktis, nors suvokiama, kad jo laisvė yra griežtai apibrėžta). Net ir planuodamas ateitį, žmogus žino, kad nors jam ir „visi keliai atviri“, tačiau savo rinkimąsi jis ribos atsižvelgdamas į daugelį veiksnių (patarimus, naudą, asmeninį požiūrį ir pan.).

Nors elektroniniai leidiniai vadinami interaktyviais, skiriasi jų interaktyvumo kokybė. Tokie kūriniai, kuriuose skaitytojas gali tik keisti informacijos peržiūrėjimo eigą, yra žemo interaktyvumo. Tikrasis interaktyvumas – tai galimybė skaitytojui keisti informaciją, tarsi „susiliesti“ su kūriniu, pasijusti aktyviu veiksmo dalyviu. Tokių kūrinių sutinkama retai. Galbūt virtualių tikrovių realizacijose.

Kita vertus, tikrasis interaktyvumas – tai galimybės vartotojui keisti informacijos turinį. Tai būseną, kai skaitytojas jau nebėra pasyvus stebėtojas, bet jis tampa aktyviu veiksmo dalyviu.

Į galimybę rinktis galima žiūrėti kaip į vieną iš pagrindinių naujosios žiniasklaidos savybių. Pasirinkime yra „užkoduotas“ tikėjimas, kad kažkoks rezultatas bus pasiektas. Kiekvienu pasirinkimu taip pat mažėja ir autoriaus vaidmuo – tačiau didėja skaitytojo atsakomybė.

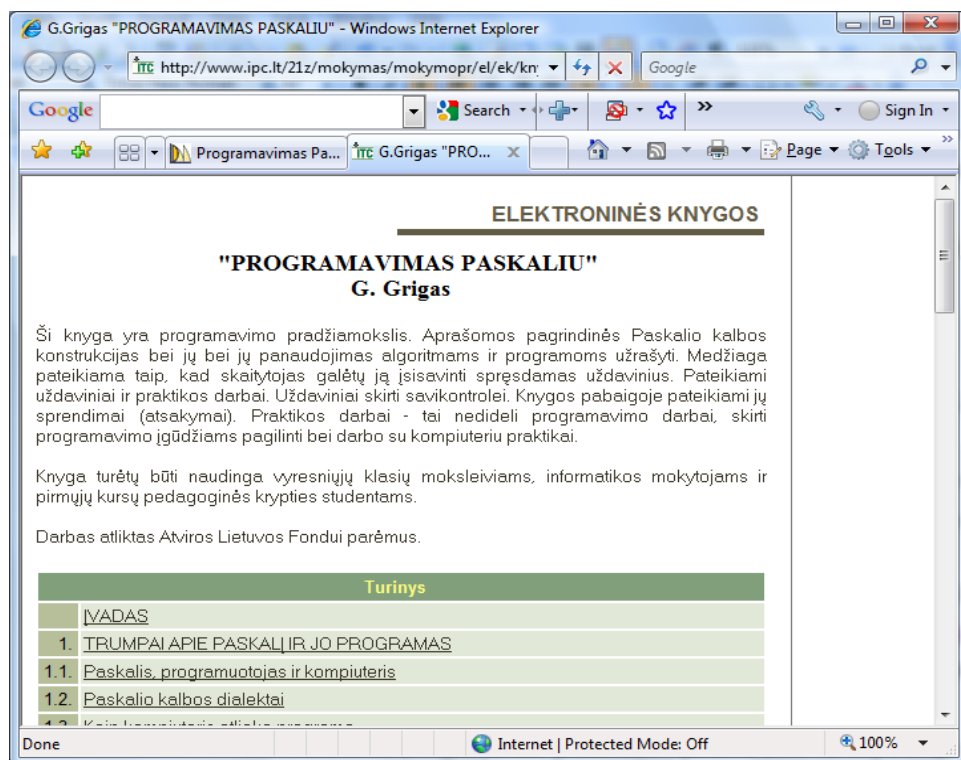
Šio darbo tikslas – elektroninio produkto, (tam tikros elektroninės terpės, leidžiančios rašyti algoritmus) skirto Paskalio programavimo pagrindų e. mokymuisi

sukūrimas. Tad norint įgyvendinti užsibrėžtą tikslą, tik žinių apie pačią mokomąją medžiagą, kuri bus talpinama svetainėje, nepakanka. Tam reikia išmanyti HTML, taip pat parinkti efektyviausią būdą turinio stiliaus projektavimui ir keitimui, išmanyti pasirinktą programavimo kalbą bei duomenų bases. Tad toliau ir pateikiamos pagrindinės priemonės interaktyvios svetainės kūrimui.

3.2. E. mokymosi Paskalyje terpių palyginimas

Norint įvertinti užsibrėžto tikslo autentiškumą buvo renkama informacija apie internetines svetaines ir elektronines knygas, kuriose pateikiama medžiaga, programavimo pradmenims Paskalio kalboje, mokytiis. Galima išskirti keletą produktų:

1. G. Grigas, elektroninė knyga „Programavimas Paskaliu“ [5]



1 pav. G. Grigas, elektroninė knyga „Programavimas Paskaliu“

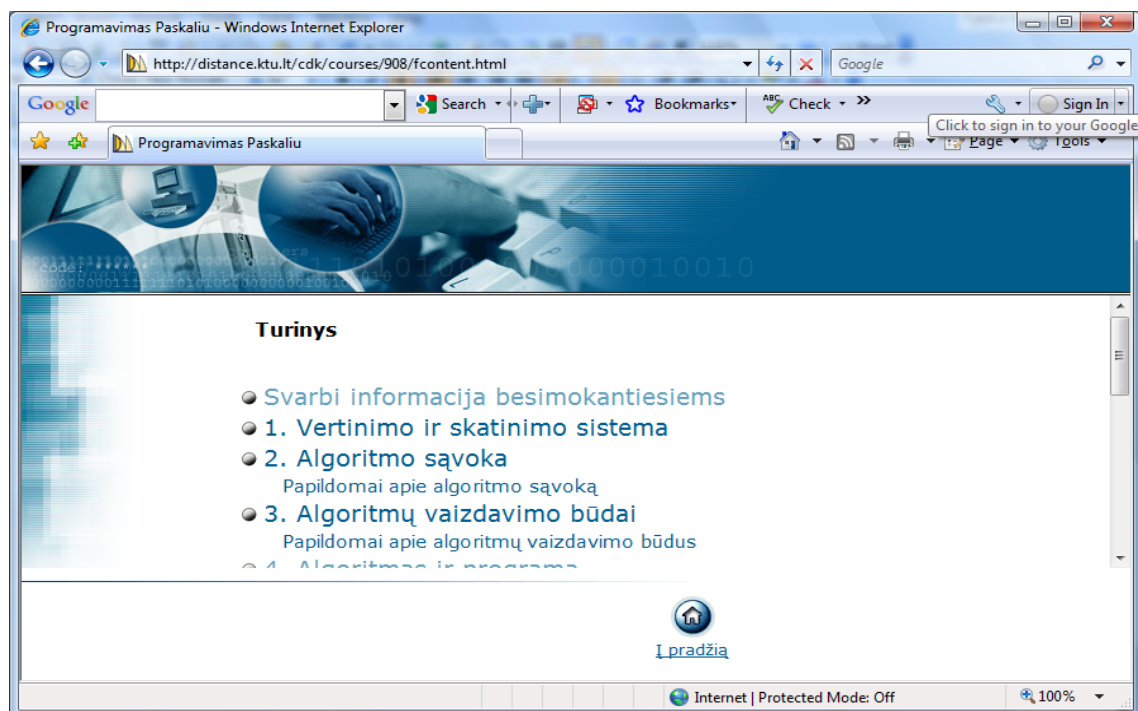
Ši elektroninė knyga (1 pav.) yra programavimo pradžiamokslis. Aprašomos pagrindinės Paskalio kalbos konstrukcijos bei jų panaudojimas algoritams ir programoms užrašyti. Medžiaga pateikiama taip, kad skaitytojas galėtų ją įsisavinti sprendamas uždavinius. Pateikiami uždaviniai ir praktikos darbai. Uždaviniai skirti savikontrolei (neinteraktyvi). Knygos pabaigoje pateikiami jų sprendimai (atsakymai). Praktikos darbai – tai nedideli programavimo darbai, skirti programavimo įgūdžiams pagilinti bei darbo su kompiuteriu praktikai.

Knyga turėtų būti naudinga vyresniųjų klasių moksleiviams, informatikos mokytojams ir pirmųjų kursų pedagoginės krypties studentams.

2. Doc. Dr. V. Undzēnas, „Paskalio pradmenys“ [6]

Tai Paskalio pradžiamokslis vadovėlis, kuriame pateikiama išsami informacija apie pagrindines Paskalio kalbos konstrukcijas bei jų panaudojimas algoritmams ir programoms užrašyti. Medžiaga pateikiama *.doc formatu, tad informacijos paieška užtrunka. Knygoje pateikti pavyzdžiai su kodo paaiškinimais. Tačiau pasigesta savikontrolės, praktinių užduočių.

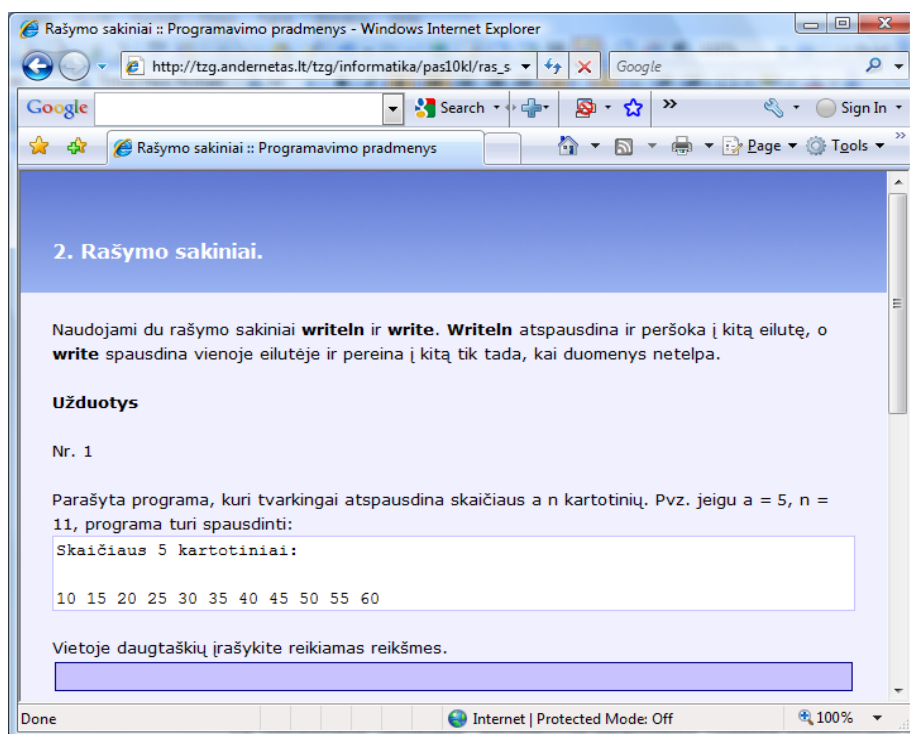
3. „Programavimas Paskaliu“ (10 kl.) svetainėje [7]



2 pav. „Programavimas Paskaliu“ (10 kl.).

Ši internetinė svetainė (2 pav.), skirta programavimo Paskalyje pagrindų mokymuisi, pirmiausia sužavi savo paprastu, bei žaviu dizainu, tačiau joje paliestos ne visos pagrindinės programavimo Paskalio kalboje temos. Šalimais teorijos, šioje svetainėje pateikiami išsamūs pavyzdžiai bei turinyje savikontrolės testas. Pasigesta praktinių užduočių bei demonstracinių pavyzdžių pateikimų.

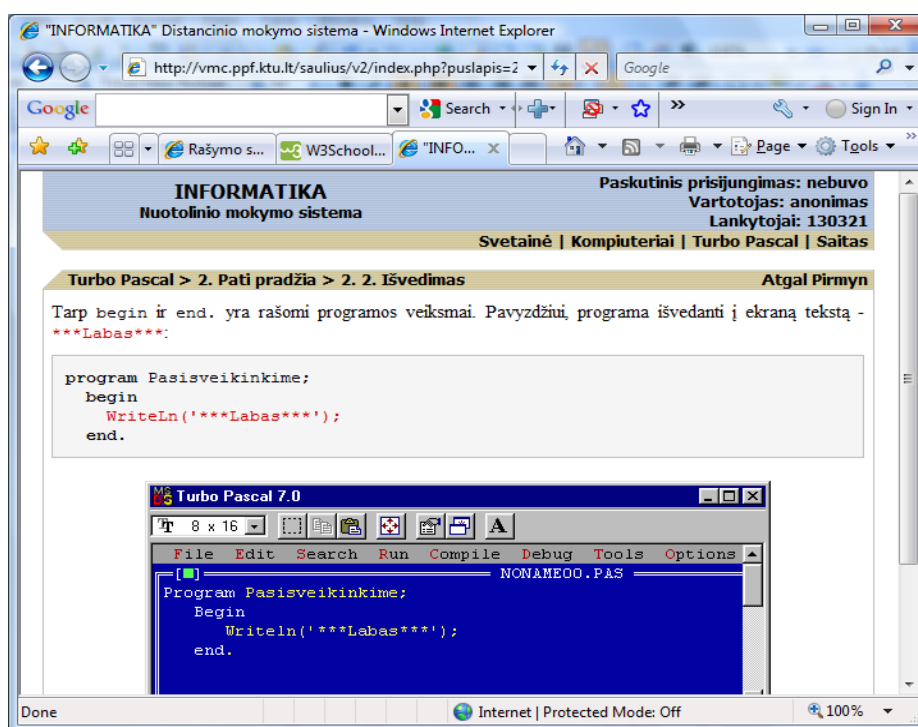
4. V. Žemaitis, „Programavimo pradmenys“ [8]



3 pav. V. Žemaitis, „Programavimo pradmenys“

Ši svetainė (3 pav.) nuo kitų išsiskiria tuo, kad joje pateikti demonstraciniai programavimo Paskalyje pavyzdžiai, trūkumas – vartotojas negali patikrinimui įvesti savo reikšmių. Svetainė nėra visiškai išbaigta, nes nenaudojamas savikontrolės testas bei nepateikiami kontroliniai darbai.

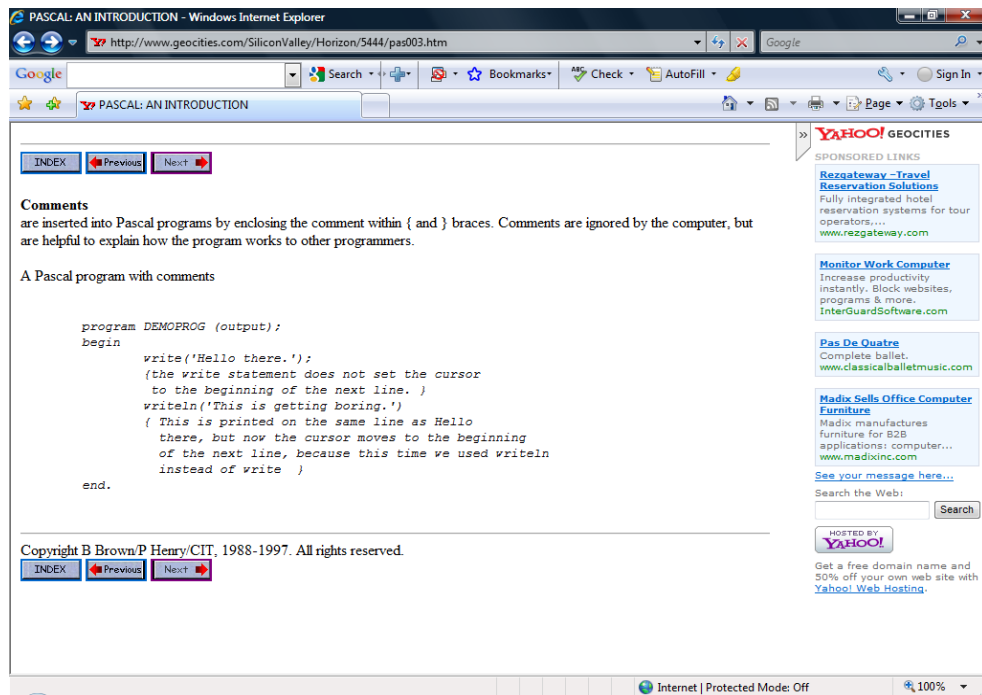
5. Informatika. Nuotolinio mokymo sistema [9]



4 pav. Informatika. Nuotolinio mokymo sistema

Šis produktas (4 pav.) sužavėjo savo pateiktais demonstraciniais pavyzdžiais, kuriuose galima pasirinkti: žiūrėti programos kodą ar gautus rezultatus, tačiau kaip ir ankstesnėse svetainėse, vartotojas išlieka pasyvus, nes pradinės reikšmės nėra leidžiamos pasirinkti. Taip pat šioje svetainėje nėra savikontrolės ir kontrolinių užduočių.

6. Pascal Programming [10]



5 pav. Pascal Programming

Pateikiama programavimo Paskalio kalboje medžiaga (5 pav.), tačiau anglų kalba, todėl ne kiekvienas žmogus gali sugebėti įsisavinti taip pateiktą medžiagą. Gan įdomiai pateikiami pavyzdžių paaiškinimai, aptariama kiekviena eilutė atskirai, tik tam tikrose dalyse pateikiamas ištisas programos kodas.

7. Pascal Programming [11]



6 pav. Pascal Programming

Šioje svetainėje (6 pav.) mokomoji medžiaga suskirstyta pagal pamokas numeriais (Lesson 1 ir t.t.), todėl neišku, apie ką bus kalbama pamokoje. Pavadinimą galima išvysti atvėrus atskirą pamoką. Medžiaga pateikiama tvarkingai ir išsamiai, su pavyzdžiais ir jų paaiškinimais. Tačiau nenaudojama savikontrolė bei nepateikiami kontroliniai uždaviniai. Ko nebuvo kitose svetainėse, bet yra šioje, tai programuotojų forumas, tačiau į jį galima patekti tik užsiregistravus.

1 lentelė E.mokymosi Paskalyje terpių palyginimas

Eil. Nr.	Sritis Produktas	Kontroliniai darbai (kodo rašymas)	Mokomosios medžiagos aiškumas	Lankytų nuorodų išskyrimas	Pavyzdžiai			Savikontrolė	Patogi navigacija	Sistemos „bendradarbiavimas“ su vartotoju
					Demonstraciniai	Pateikiamas programos kodas	Pateikiamas kodo paaiškinimas			
1.	G. Grigas, elektroninė knyga „Programavimas Paskaliu“	Taip	Taip	Ne	Ne	Taip	Taip	Taip	Ne	Ne
2.	Doc. Dr. V. Undzėnas, „Paskalio pradmenys“	Ne	Taip	Ne	Ne	Taip	Taip	Ne	Ne	Ne
3.	„Programavimas Paskaliu“ (10 kl.) svetainėje	Ne	Taip	Taip	Ne	Taip	Taip	Taip	Taip	Ne
4.	V. Žemaitis, „Programavimo pradmenys“	Ne	Taip	Ne	Taip	Taip	Taip	Ne	Ne	Ne
5.	Informatika. Nuotolinio mokymo sistema	Ne	Taip	Ne	Taip	Taip	Taip	Ne	Taip	Ne
6.	Pascal Programming	Ne	Taip	Ne	Ne	Taip	Taip	Ne	Taip	Ne
7.	Pascal Programming	Ne	Taip	Ne	Ne	Taip	Taip	Ne	Taip	Ne

Apžvelgus šiuos produktus, matyti, kad skaitytojas vis tik išlieka pasyvus. Didžiausiu, norimo sukurti produkto, „konkurentu“ galima vadinti KTU produktą „Programavimas Paskaliu (10 kl)“.

3.3. Programinių įrankių analizė

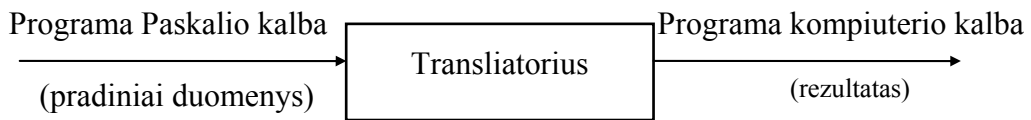
3.3.1. Paskalis, programuotojas ir kompiuteris

Kompiuteris gali atlikti tikrai tokias programas, kurios sudarytos iš jam suprantamų komandų, kurios žmogui atrodo kaip beprasmis dvejetainių (arba šešioliktainių) skaičių rinkinys.

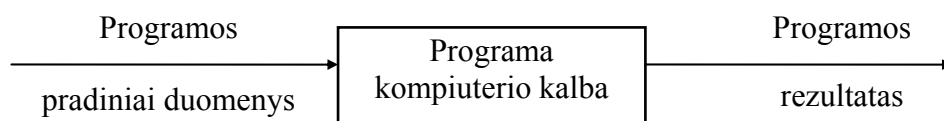
Kaip įveikti barjerą tarp žmogaus ir kompiuterio, t.y. kaip padaryti, kad kompiuteris suprastų ir galėtų įvykdyti Paskalio kalba parašytą programą?

Reikia programą iš Paskalio kalbos išversti į kompiuterio kalbą. Šį darbą atlieka programa, vadinama transliatoriumi (angl. *translator* – vertėjas) (7 pav.). Išverstą programą

kompiuteris jau gali vykdyti (8 pav.). Kompiuteriui pateikus programą, o po to pradinius duomenis, iš kompiuterio gaunamas rezultatas.



7 pav. Transliatorius išverčia programą iš Paskalio kalbos į kompiuterio kalbą

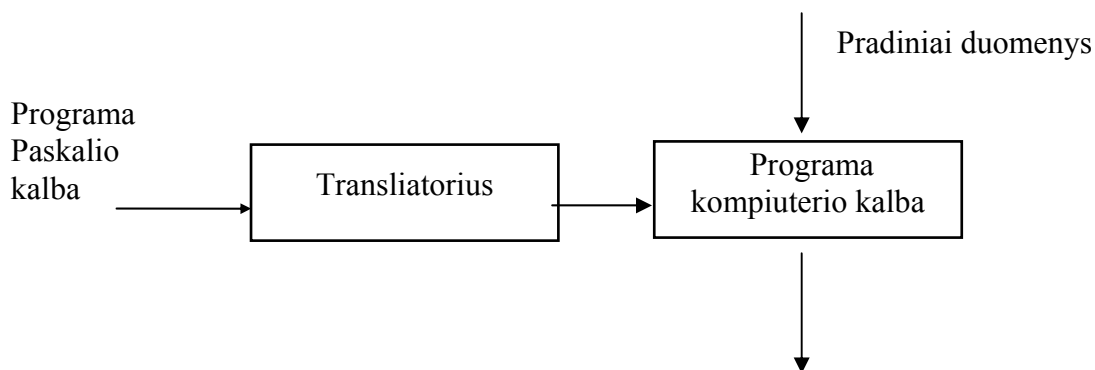


8 pav. Kompiuteris atlieka į jo kalbą išverstą programą

Kompiliatorius ne tik išverčia programos tekstą iš Paskalio kalbos į kompiuterio kalbą, bet ir į programą įjungia (įkompiluoja) tam tikras iš anksto parengtas programas arba programų fragmentus.

Paskalio kalbos kompiliatorius yra didelė ir sudėtinga programa, kuri šiame darbe nebus apžvelgiama ir į ją nesigilinama, kadangi tikslo siekimui pasirinktas Paskalio kalbą palaikantis produktas, kuriame yra reikiamas kompiliatorius.

7 ir 8 paveiksluose programos vaizduojamos stačiakampiais, o duomenys – rodyklėmis, jungiančiomis programas (į stačiakampį ateinančios rodyklės vaizduoja pradinius duomenis, o išeinančios – rezultatus). Tie patys duomenys vienai programai gali būti pradiniai duomenys (ateinanti rodyklė), kitai – rezultatas (išeinanti rodyklė). Tai, kad sąveikaujant programoms keičiasi duomenų rolė (rezultatai virsta pradiniais duomenimis) savaime suprantama. Tačiau atidžiau panagrinėję minėtus paveikslus galime pastebėti įdomesnį dalyką: duomenys gali būti laikomi programomis ir atvirkščiai – rezultatai programomis. Žmogaus parašytą Paskalio kalba programą transliatorius laiko pradiniais duomenimis ir ją perdirba į kitus duomenis – rezultata. Kai šis rezultatas pateikiamas kompiuteriui, kompiuteris ją laiko programa ir ima ją vykdyti. [12]



9 pav. Programos transliavimas ir vykdymas. Transliavimo rezultatas virsta programa

Programavimo terpė.

Darbo su programomis ir duomenimis schema pateikta 9 paveiksle, yra paprasta ir akivaizdi. Tačiau dirbti pagal šią schemą būtų nelabai patogus: reikia operuoti daugeliu duomenų ir programų bylų. Programos tekstą Paskalio kalba bei jos pradinius duomenis reikia parašyti su koku nors tekstų redaktoriumi. Po to programos tekstą reikia pateikti transliatoriui, iš jo gautą rezultatą (sutransliuotą programą) pateikti kompiuteriui, o kad jis galėtų ją vykdyti – pateikti programos pradinius duomenis, o iš jos gautą rezultatą bylą skaityti vėl su koku nors redaktoriumi. Taigi, reikia atlikti daug veiksmų su bylomis. Šie veiksmai ypač juntami, kai programa yra dar tik rašoma, nes tada ją dažnai tenka taisyti, o pataisius visus tuos veiksmus vėl reikia kartoti.

Tam, kad mažiau rūpesčių keltų darbas su bylomis, programuotojui pateikiamas ne vien transliatorius (kompiliatorius), bet visa programuotojui reikalinga terpė – *programavimo sistema*. Transliatorius (kompiliatorius) yra svarbiausias sistemos komponentas. Todėl kartais visa programavimo sistema sutapatinama su transliatoriumi (kompiliatoriumi).

Kitas svarbus programavimo sistemos komponentas yra *programos tekstų redaktorius*. (redaktorius). Tada programos tekstui rinkti bei taisyti nereikia atskiro kokio nors kito redaktoriaus. Pradėjus darbą su sistema, iškart įsijungia redaktorius ir kompiuterio ekrane galima rinkti programos tekstą. Programavimo kalbos redaktoriumi su tekstu galima atlikti tokias pat operacijas, kaip ir kitais redaktoriais: jį rinkti, taisyti, išbraukti, kopijuoti, įterpti į kitą tekstą, įkelti iš bylos, užrašyti į bylą ir pan.

Programavimo terpė paslepia ir patį transliavimo procesą. Ja naudojantis susidaro įspūdis, kad kompiuteris supranta Paskalio kalba parašytą programą ir čia pat ją atlieka.

Paskalio kalbos dialektai

Kompiliatorių autoriai stengiasi programavimo kalbą kiek galima geriau pritaikyti programuotojų poreikiams, t.y. patobulinti. Šitaip atsiranda kalbos variantai arba dialektai. Tai gerai, nes kalba vystosi, tobulėja. Tačiau pasidaro sunkiau susikalbėti skirtingų dialektų naudotojams. Dėl to programavimo kalbos norminamos, apibrėžiami jų standartai. Paskalio kalbos standartas yra kalbos branduolys, į kurį surinktos pačios svarbiausios jos konstrukcijos. Jis yra tarsi orientyras, vienijantis programuojančius įvairiais Paskalio dialektais.

Standartinis Paskalis turi universalų, bet nedidelį konstrukcijų rinkinį, kuriuo patogiu programuoti įvairių žmogaus veiklos sričių (matematikos, fizikos, chemijos, ekonomikos ir kt.) uždavinius. Tokie uždaviniai egzistavo visą laiką ir buvo sprendžiami įvairiais tuo metu naudotais kompiuteriais: didelėmis skaičiavimo mašinomis, stovėjusiomis skaičiavimo centruose, pirmaisiais kuklių galimybių mikrokompiuteriais ir šiuolaikiniais IBM PC genties kompiuteriais. Jų algoritmus galima pavadinti klasikiniiais. Jų mokomasi mokyklose, jie rašomi informatikos olimpiadose, jais užrašomi įvairių žmogaus veiklos sričių uždavinių matematiniai sprendimai.

Daugiausiai vartojamas Paskalio kalbos dialektas yra *Turbo Paskalis*. Jis paplito po visą pasaulį ir tapo netgi daugiau žinomas, negu standartinis Paskalis.

Turbo Paskalis turi beveik visas standartinio Paskalio konstrukcijas, o taip pat daugybę naujų konstrukcijų (papildymų). Dalis papildymų suteikia naujų, dažniausiai alternatyvių galimybių klasikiniams algoritmams užrašyti, tačiau daugiausia papildymų skirti kompiuterio įrenginiams valdyti. Jie įgalina programuoti grafinį žmogaus ir kompiuterio dialogą, tiesiogiai prieiti prie duomenų, saugomų kompiuterio atmintinėje ir jais operuoti. Dėl to Turbo Paskalis tinka taip vadinamoms sisteminėms programoms – įvairioms kompiuterio įrenginių tvarkyklėms, įvairiems operacinių sistemų komponentams rašyti.

3.3.2. *HTML*

HTML (HyperText Markup Language) – tai WWW puslapių aprašymo kalba, kuria „kalba“ pasaulinio tinklo WWW serveriai ir kurią „supranta“ tinklo naršyklės.

HTML – tai vienas iš SGLM kalbos variantų. Pastarasis dokumentų struktūros aprašymo būdas buvo patvirtintas ISO 8779 standartu. SGLM kalba vartojama pavyzdžiui, siekiant standartizuoti didelių tarptautinių organizacijų raštvedybą ir tarpusavio susirašinėjimą.

HTML buvo sumanyta kaip loginės struktūros aprašymo kalba, bet greitai paaiškėjo, jog WWW puslapių kūrėjams bei skaitytojams to nepakanka. Dėl to HTML be

loginių gali aprašyti ir fizines dokumento savybes kaip antai, vartojamo šrifto parametrus, lentelių, iliustracijų bei kitų elementų dydžius ir pan.

3.3.3. *DHTML*

Dinaminė HTML kalba, patobulintas HTML kalbos variantas, kai galima keisti tinklalapio turinį, vaizduojamą naršykle, atsižvelgiant į skaitytojo veiksmus, iš naujo neatsiunčiant tinklalapio iš serverio. Tinklalapį sudaro visuma objektų, kurių išvaizdą gali keisti su tinklalapiu susieti scenarijai. Pavyzdžiui, tinklalapyje gali būti trumpų užrašų, kuriuos paaiškinantys tekstai atsiranda tik tada, kai skaitytojas perkelia pelės žymeklį ant jų dominančio užrašo.

3.3.4. *AJAX*

AJAX arba Asinchroninis JavaScript ir XML programavimas – terminas, apibrėžiantis svetainių programavimo technologiją, naudojančią šias priemones maksimaliam interaktyvumui pasiekti:

- **HTML (ar XHTML)** bei stilių lenteles (Cascading Style Sheets) informacijos vaizdavimui;
- **Dokumento Objektinį Modelį (DOM)** bei JavaScript kalbą dinamiškam vaizdavimui bei interaktyvumui;
- **XML, XSLT ir XMLHttpRequest** objektą asinchroniniam duomenų apsikeitimui su serveriu (AJAX tam galėtų naudoti ir kitas technologijas).

Kaip ir DHTML ar LAMP, AJAX nėra technologija, bet tik abstraktus terminas, apimantis kelių technologijų kombinaciją. Puslapiai, sukurti naudojant AJAX technologiją, reikalauja naršyklių, palaikančių šias technologijas. Tokios naršyklės yra Mozilla Firefox, Internet Explorer, Opera, Konqueror ir Safari.

AJAX puslapių ypatumai

Tradicinių svetainių interaktyvumas kuriamas kaskart kreipiantis į serverį. Pavyzdžiui, užpildžius anketą (ar formą), kreipiamasi į serverį ir užkraunamas naujas puslapis. Tokiu būdu bereikalingai iššvaistomi resursai, nes didelė dalis informacijos nepakinta ir siunčiama kaskart iš naujo. Taip pat tokiu būdu neįmanoma pasiekti tokio interaktyvumo, kokį gali suteikti ne internetinės aplikacijos. AJAX svetainės gali siųsti užklausas serveriui, gauti atsakymą tam tikra apsibrėžta forma (SOAP ar kita XML paremta), bei naudojant JavaScript programavimą atnaujinti tik reikiamą puslapio dalį. Tokiu būdu sutaupomi tinklo resursai, nes žymiai sumažinami duomenų srautai. Taip pat taupomi ir serverio procesoriaus resursai, nes dalis logikos perkeliama klientui.

Privalumai bei trūkumai

Kaip ir DHTML svetainėms, AJAX svetainėms reikia žymiai daugiau testavimo, nes kiekviena naršyklė gali elgtis šiek tiek skirtingai. Tačiau naudojant šią technologiją paspartinamas interaktyvumas. Naudojant AJAX, padaugėja mikro-užklausų į serverį, tai neapgalvotai naudojant gali sulėtinti bendravimą tarp kliento ir serverio. Kartais sunku ar net neįmanoma AJAX svetainių pritaikyti nepalaikomoms naršyklėms, taip pat sudėtinga pritaikyti svetaines standartams, kurie atsižvelgia į žmones su negalia...

3.3.5. XML

XML skirta duomenims sisteminti. Sistematizuotiems duomenims priklauso tokie dalykai kaip elektroninių sprendinių lentelės, adresų knygelės, nustatymų parametrai, finansiniai pavedimai ir techniniai brėžiniai. XML yra taisyklių rinkinys (galima laikyti jas nurodymais ar susitarimais) tekstams formatams kurti, kurie leidžia sistematizuoti jūsų duomenis. XML nėra programinė kalba, ji palengvina užduotį kompiuteriui atvaizduoti, nuskaityti duomenis ir įsitikinti, kad duomenų struktūra nėra dviprasmiška. XML išvengia įprastų kalbos sandaros spąstų, yra išplečiama, nepriklauso nuo platformos ir palaiko internacionalizaciją ir lokalizaciją. XML visiškai atitinka Unikode standartą.

Kaip ir HTML, XML naudoja tegus (žodžius, atskirtus „<“ ir „>“) ir atributus (formoje pavadinimas="argumentas"). Kai HTML nurodo, ką reiškia kiekvienas tegas ir atributas (dažnai ir kaip tekstas atrodys naršyklėje), XML naudoja tegus tik atskirti duomenų dalis. Duomenų interpretavimą palieka taikomajai programai, kuri juos nuskaityto.

3.3.6. Firebird Server Manager

Firebird yra gana populiarus komercinio DB serverio „Borland Interbase 6.x“ atšaka. Tai reliacinė duomenų bazės valdymo sistema, pasižyminti greičiu ir siūlanti daug ANSI SQL standartinių ypatybių. Ji veikia Linux, Windows, ir Unix platformose. Firebird siūlo puikų sutapimą, aukštą atlikimą, ir galingą kalbos palaikymą aprūpintoms procedūroms ir trigeriams. Naudojant šį produktą, nėra jokių mokesčių už duomenų perkėlimą, registraciją, licencijavimą ar išdėstymą, tai laisvai platinama (pagal IPL ir IDPL licencijas) programa. [13]

3.3.7. Morfik Fx For Object Pascal

Programa Morfik FX for Object Pascal nutaikyta į išvystas Ajax paraiškų aukšto lygio programavimo kalbas tokias kaip Java, C #, pagrindinis objektas Paskalis. Morfik apima regimuosius projekto įrankius formas, užklausas, ataskaitas, lenteles, modulius, Web metodus, duomenų bazės struktūrai, ir klausimams. Ji palaiko klasikinį kliento serverio modelį, tačiau

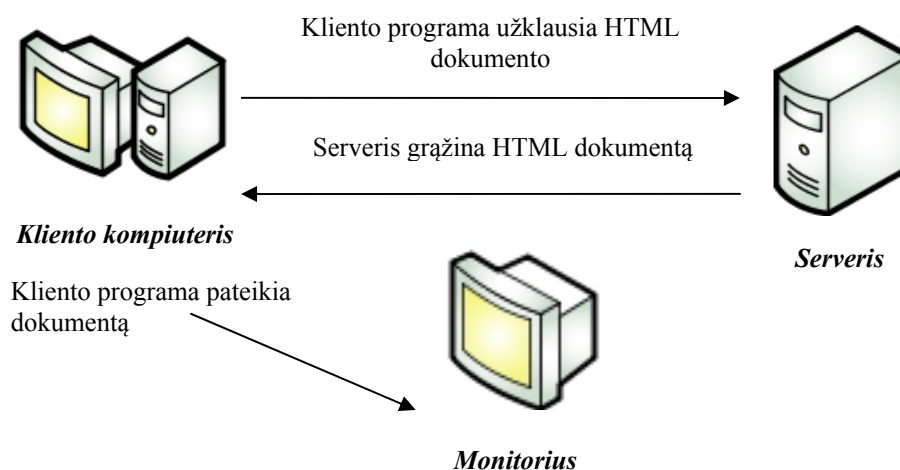
kaip visų Ajax paraiškų, kodiniai kliento pusės prabėgimai naršyklėje. Morfik išsivystymo įrankis konvertuoja formas, kurias vartotojas įtraukia į DHTML, surenka kliento pateiktą informaciją į JavaScript, ir sukuria aplikaciją bei duomenų bazės serverio variklius, kad įvykdytų serverio pusės kodą. [14]

3.3.8. *Serverio ir kliento programos*

HTML yra kliento programos technologija. Tai reiškia, kad HTML dokumentą apdoroja vien kliento programa. Serverio veiksmai nepriklauso nuo kodo, esančio HTML dokumente. Ji tiesiog pateikia užklaustus failus. Kaip juos pateikti, nusprendžia kliento žiniatinklio naršyklė. HTML nėra programavimo kalba, ji neturi jokių duomenų apdorojimo funkcijų.

Priešingai HTML, Morfik Fx For Object Paskal sukurtas projektas veikia tik serveryje. Kai vykdomas scenarijus, jo vykdymas nepriklauso nuo žiniatinklio naršyklės, ir tik galutinis scenarijaus produktas, dažniausiai HTML dokumentas, grąžinamas naršyklei. Jeigu naršyklė gautų neapdorotą Morfik Fx For Object Paskal scenarijų, ji bandytų jį atvaizduoti kaip įprastą HTML dokumentą. Naršyklės negali vykdyti Morfik Fx For Object Paskal scenarijų.

HTML yra būtinas žiniatinklio programų kūrimo komponentas. Morfik Fx For Object Paskal kodas gali būti įterptas tiesiai į HTML. Kai kliento programa serverio pareikalauja HTML dokumento, stotis jį siunčia klientui (10 pav.).



10 pav. HTML dokumento užklausa

Morfik Fx For Object Paskal scenarijaus užklauso veikia skirtingai. Prieš išsiunčiant dokumentą kliento programai, dokumentą peržiūri Morfik Fx For Object Paskal, kuri vykdo dokumente rastą visą Morfik Fx For Object Paskal kodą.

3.4. Išvados

1. Apžvelgtos išsamiausios internetinės svetainės ir knygos, kuriose pateikiama medžiaga programavimo pradmenims Paskalio kalboje mokytis. Nei vienoje iš jų nenaudojamas aktyvus interaktyvus mokymas t.y. pateikiami pavyzdžiai bei animacijos, tačiau besimokantysis išlieka pasyvus.

2. Atliekant programinių įrankių analizę, nuspręsta internetinės svetainės kūrimui pasirinkti tuos produktus, kurie mažiau sunaudoja resursų informacijos siuntimo metu. Taip tikintis pasiekti geresnį interaktyvumą.

3. HTML dokumento ir Morfik FX For Object Pascal scenarijaus užklauso parodo jų skirtumą: Borland Paskal vykdoma serveryje (FireBird) ir yra visavertė programavimo klaba, HTML skirta ne tik hiperteksto publikavimui, ji vykdoma kliento programoje. Morfik FX For Object Pascal palaiko HTML, todėl teksto apipavidalinimui galima naudti standartinius tegus.

4. INTERAKTYVIOS E. MOKYMO SI PASKALYJE TERPĖS MODELIO PROJEKTAVIMAS

4.1. Technologijų pasirinkimas

Duomenų struktūra

2 lentelė Duomenų saugojimo technologijų palyginimas pagal kriterijus

Technologija	Kaina	Privalumai	Trūkumai
XML	Nemokama	Struktūrizuota, išplečiama, nepriklauso nuo platformos ir palaiko internacionalizaciją.	Duomenys saugomi tekstiniu formatu.
Tekstinės rinkmenos	Nemokama	Lankstumas.	Sudėtingas nuskaitymas ir patikra.
Morfik FX for Object Pascal	499 \$ (Shareware)	Naudojama objektinio Paskalio sintaksė. Galimas XML panaudojimas. Formose sukelta informacija konvertuojama į JavaScript arba DHTML.	Reikalingi dideli techniniai resursai.

Projekto kūrimui pasirinkta Morfik FX for Object Pascal terpė. [20] Kiek man yra žinoma, tai viena naujausių (paleidimo data 2008.11.22) programų, kurių pagalba galima kurti svetainės Paskalio kalba, būtent ši savybė labiausiai ir lėmė pasirinkimą. Morfik FX Object Pascal terpės lango vaizdas pateiktas 1 priede.

Dokumento struktūra

3 lentelė DOM technologijų palyginimas pagal kriterijus

Technologija	Kaina	Privalumai	Trūkumai
Mozilla	Nemokama	Palaiko W3C sąsajas. Gerai rodo nestandartinius puslapius.	
Gecko	Nemokama	Nuo platformos nepriklausoma.	Prastai rodo standartų nesilaikančius puslapius.
Opera		Nuo platformos nepriklausoma. Greita.	Prastai suderinama su NET.
MSHTML DOM	Nemokama	IE branduolys visiškai palaiko W3C DOM sąsajas.	Lėtas, veikia tik Windows OS.

Labai didelio skirtumo kuri naršyklė bus pasirinkta nėra. Morfik FX for Object Pascal sukurtas projektas puikiai veikia su Internet Explorer, Mozilla, Opera, Chrome, Safari naršyklėmis. Pastarųjų nuorodos, pateikiamos kompiliavimo lango antraštėje.

Kūrimo metu buvo naudojama **Mozilla FireFox**.

Duomenų bazės

4 lentelė Duomenų bazių sistemų palyginimas

Technologija	Kaina	Privalumai	Trūkumai
MS SQLServer MSDE	Nemokama	Greitas, lankstus, visiškai suderinamas su XML ir NET.	Neturi administravimo įrankių, veikia tik Windows OS.
MySQL	Nemokama	Nuo platformos nepriklausoma.	Prastas suderinamumas su XML ir NET.
FireBird	Nemokama	Suderinama su Windows, Linux ir Unix. Suderinama su Borland	

Pasirinkta FireBird, nes ji suderinama su Borland. Ji automatiškai įdiegiama, naudojant Morfik FX for Object Pascal programą.

Programavimo kalba ir aplinka

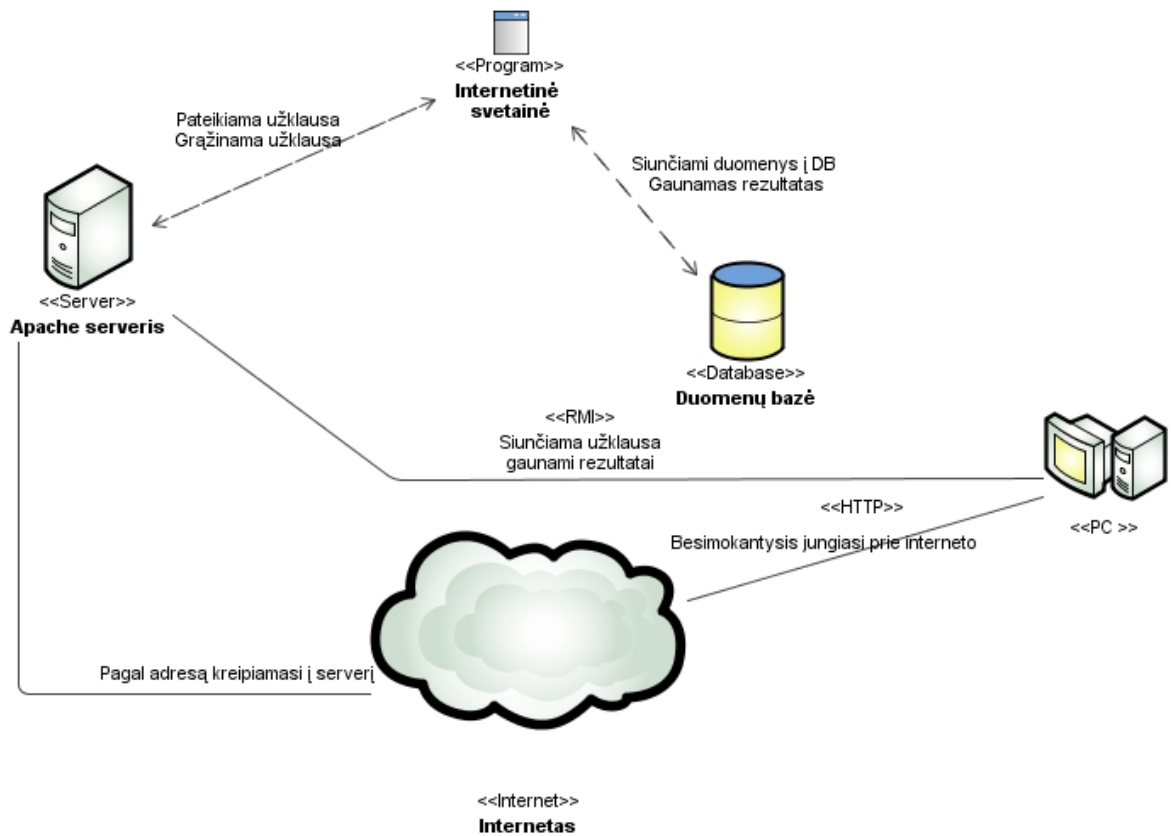
5 lentelė Programavimo technologijų palyginimas pagal kriterijus

Technologija	Kaina	Privalumai	Trūkumai
JAVA	Nemokama	Nuo platformos nepriklausoma.	Lėta
Borland Pascal	Nemokama	Saugumas, paprasta algoritmo struktūra	

Sukurtame e. mokymosi produkte mokoma programavimo Paskalyje pradmenų. Pačioje mokomojoje medžiagoje suteikiama galimybė įvesti besimokančiajam savo sugalvotą programos kodą. Norint išvesti klaidas ir jas paaiškinti, įvestas kodas turi būti patikrintas. Norint, kad nekiltų problemų su kodo kompiliavimu, programavimo kalba pasirinkta Borland Pascal.

Viso darbo įgyvendinimui naudojama Morfik FX for Object Pascal programa, kuri įvestą informaciją konvertuoja į Java skriptus, tai galima teigti, kad ir ši programavimo kalba bus panaudota, tik netiesiogiai.

11 paveikslėlyje pateikta vartotojo ir sistemos sąveikos schema.



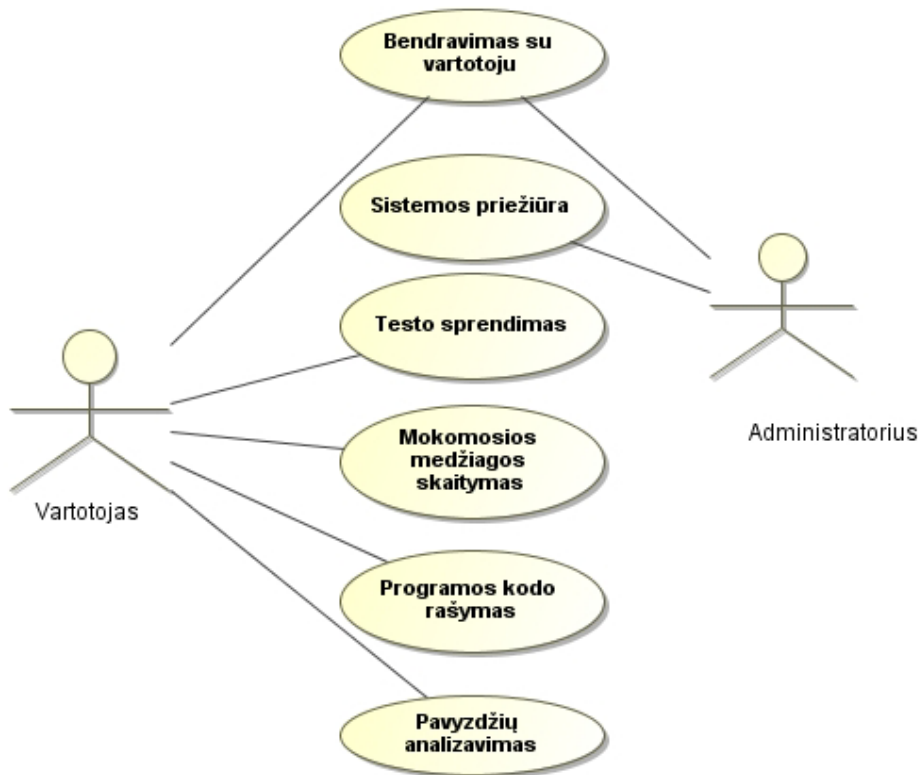
11 pav. Vartotojo ir sistemos sąveikos schema

4.2. Formalus aprašymas

Kaip jau minėta, sistemos kūrimo naudojama Morfik FX for Object Pascal programavimo – svetainių kūrimo programa:

- Stengiamasi projektuoti taip, kad sistemą būtų lengviau palaikyti, objektai būtų nepriklausomi. Objektų paskirties pakeitimas ar servisų papildymas nedarys jokios įtakos kitiems sistemos objektams;
- Objektams būdingas daugkartinis komponentų panaudojimas.

Projektuojant identifikuojami principiniai sistemos objektai ir jų tarpusavio ryšiai. Jiems atvaizduoti naudojama unifikauta modeliavimo kalba (UML) (12 pav.).



12 pav. Paskalio programavimo kalbos interaktyvios mokymosi svetainės panaudos atvejų diagrama

4.3. Vartotojo sąsaja

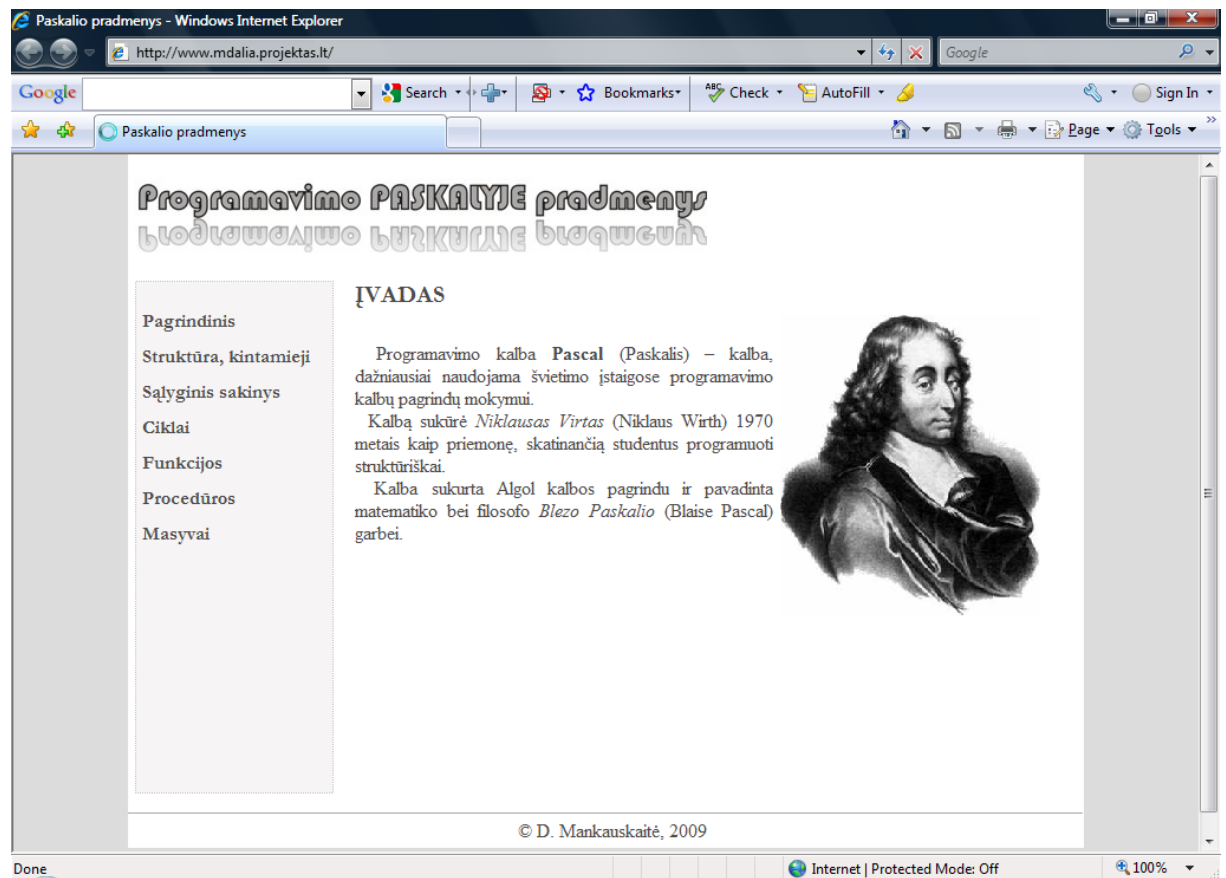
4.3.1. Schematinis produkto veikimo planas:

1. Kuriamos formos;
2. Formų komponentų parinkimas, procedūrų aprašymas;
3. Nustatomi vienodi vienos rūšies komponentų parametrai;
4. Mokomosios medžiagos pateikimas;
5. Pretesto ir savikontrolės klausimų kaupimas. Klausimų išvedimas ir atsakymo patikros algoritmo kūrimas;
- 6. Įvedamų laukų ir kodo patikros algoritmo rašymas;**
7. Kompilijuojama ir sukuriama paleidžiamasis failas.

Kuriamas produktas orientuotas į žmonės, kurie yra susipažinę su informacinėmis technologijomis. Kuriant vartotojo sąsają vadovaujama tuo, kad svetainė vartotojui būtų suprantama ir būtų patogiu ja naudotis. Todėl sukurta grafinė vartotojo sąsaja su nesudėtinga meniu sistema. Taip pat vartotojas pats įveda jam reikiamus duomenis. Vartotojo įvestų duomenų teisingumas yra tikrinamas ir vartotojas informuojamas apie padarytas klaidas. Kadangi programos naudotojas turi galimybę įvesti neteisingus duomenis

programoje atliekama duomenų kontrolė, vartotojo informavimas apie jo klaidingus veiksmus. Vartotojo pranešimams naudojami paprasti, aiškūs ir informatyvūs sakiniai.

Vartotojas turi galimybę mokytis jį dominančią temą, nereikia viso kurso eiti iš pat pradžių.



13 pav. E. mokymosi Paskalyje produkto pradinis langas

Paveikslėlyje Nr. 13 pateiktas darbo lango prototipas. Jis sudarytas iš 4 dalių:

- Darbo pavadinimo;
- Meniu, kuriame galima pasirinkti, temą;
- Pasirinkimų rodymo lango;
- Autorinio lauko.

4.4. Specifikacija

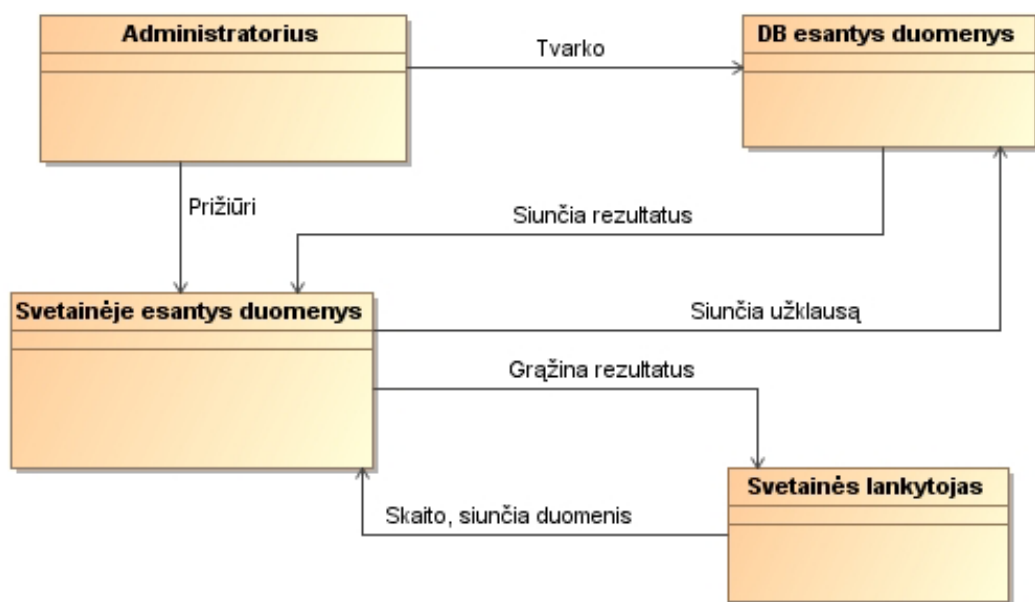
Kuriamas e.mokymosi kursas, skirtas Paskalio pradmenų mokymuisi. Šioje mokymosi terpėje mokomoji medžiaga pateikiama skyriais. Pasirinkus skyrių pirmiausia pateikiamas tikslas (kas bus dėstoma skyriuje). Šalia teksto pateikiami sekantys mokymosi žingsniai, kuriuo besimokantysis gali pasirinkti pagal savo žinias:

- Pretestas;
- Mokomoji medžiaga;

- Demonstraciniai pavyzdžiai;
- Praktinės užduotys;
- Žodynėlis;

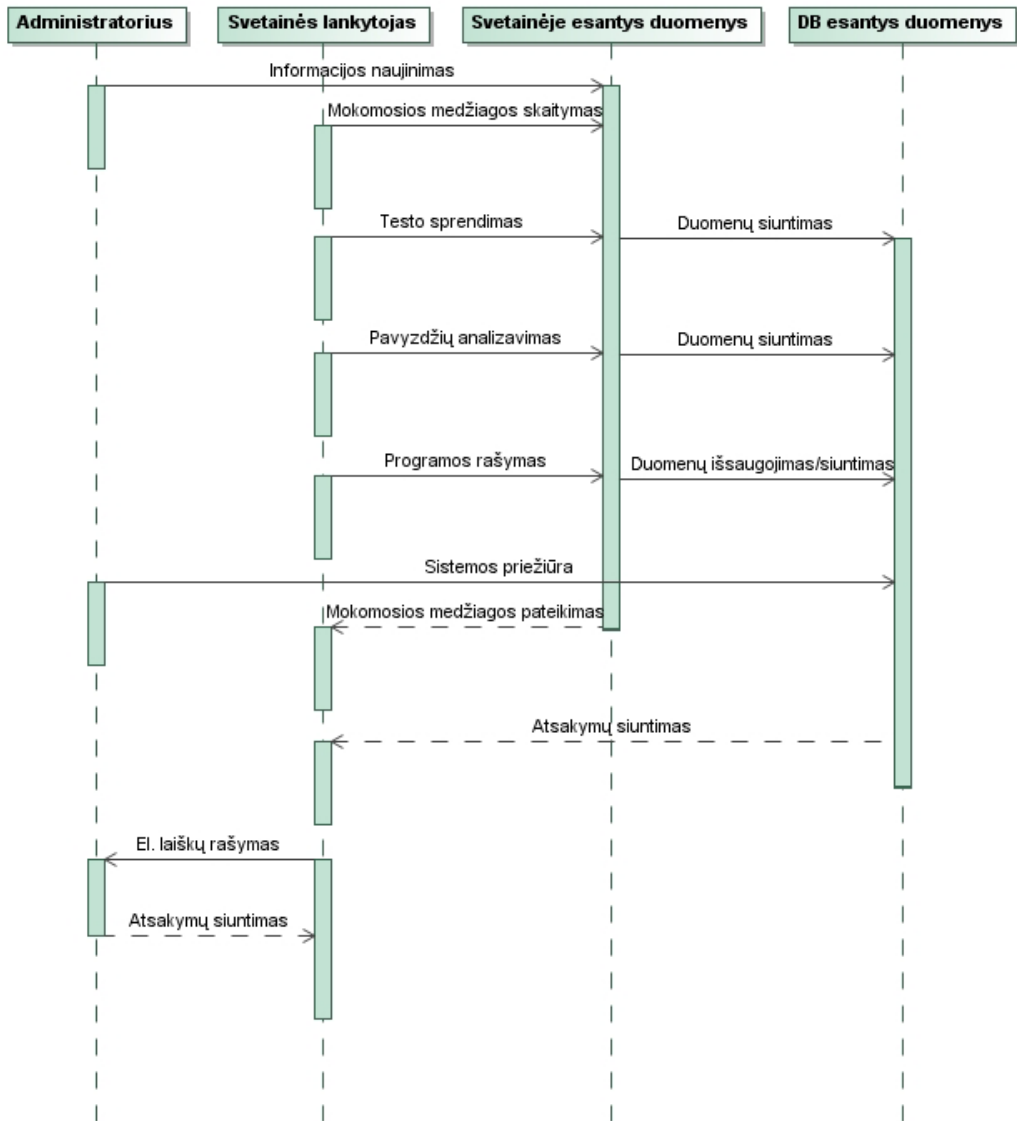
Įvykdęs pretesto užduotis besimokantysis sužino ar pasirinkto skyriaus medžiaga jam žinoma, ar dar reikia pagilinti žinias. *Praktinėje dalyje suteikiama galimybė įvykdyti savo sugalvotą kodą arba pritaikyti teorinėje dalyje įgytas žinias laukų užpildymui.* Atlikus duomenų įvedimą, gauta informacija siunčiama į serverį ir tikrinamos klaidos. Aptikus klaidas išvedamas aiškinamasis pranešimas.

Klasių diagrama (14 pav.):



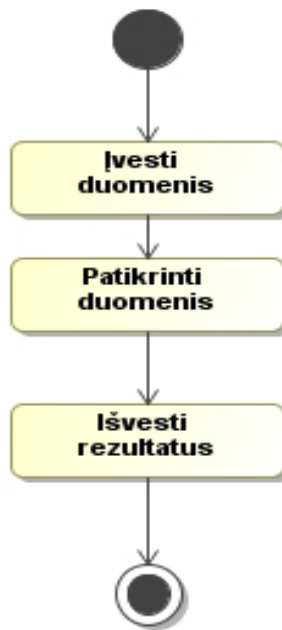
14 pav. Klasių diagrama

Informacijos gavimo ir atidavimo srantai pateikiami UML sekos diagramoje (15 pav.).



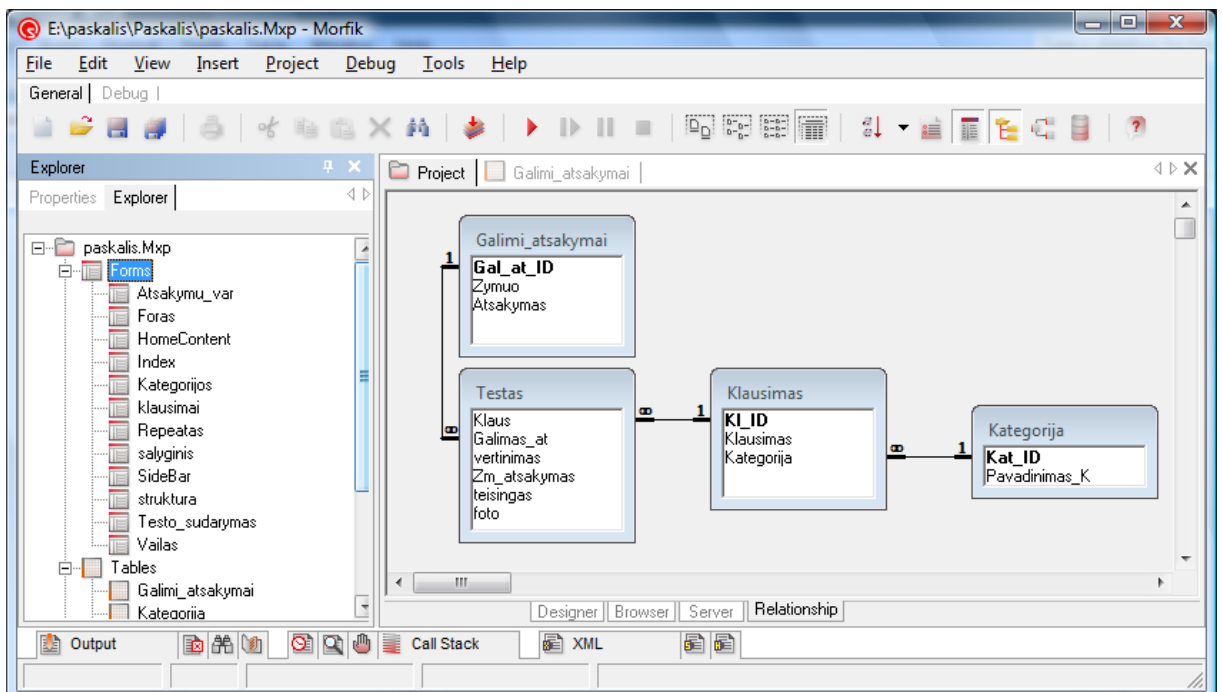
15 pav. UML sekos diagrama

Būsenų diagrama (parodo svarbiausias sistemos būsenas ir jų kitimą) (16 pav.):



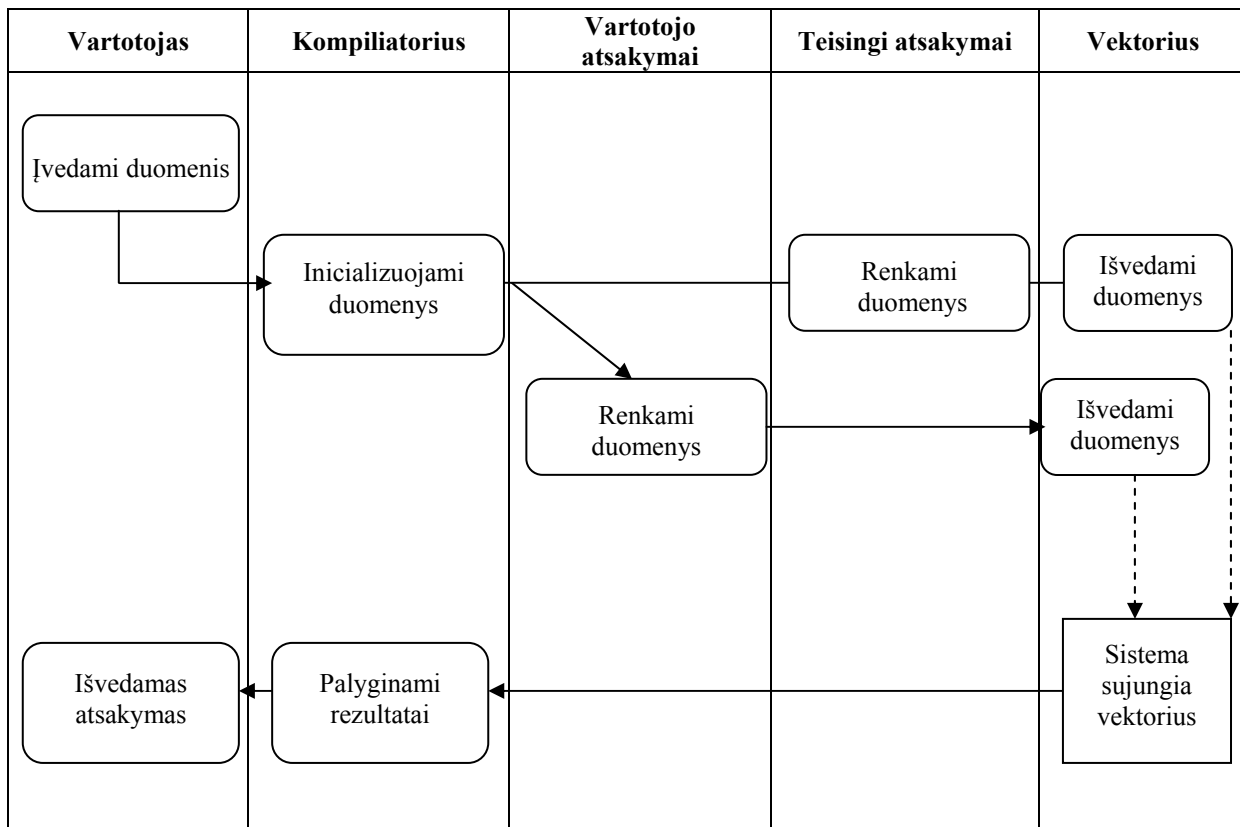
16 pav. Būsenų diagrama

Testo sudarymui, sukurta minimali duomenų bazė, kurios pagalba patikros testų klausimai bus kaupiami ir laikui bėgant keičiami. Testo lentelių sąryšiai pavaizduoti 17 paveikslėlyje.



17 pav. Testo lentelių sąryšiai

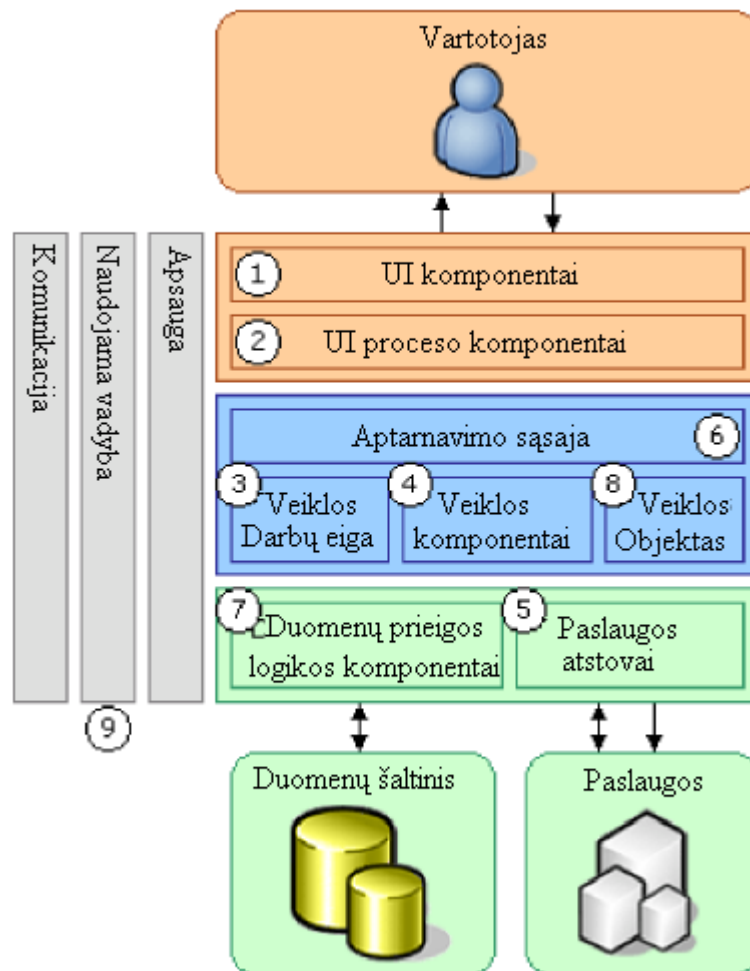
Testo veikimo veiklos diagrama (18 pav.):



18 pav. Testo veikimo veiklos diagrama

4.5. Architektūra

Programos architektūra projektuota pagal 3 lygių architektūrą:



19 pav. Programos architektūra

Trijų lygių architektūra leidžia kiekvieną komponentą kurti atskirai nuo kitų. Ryšiai tarp objektų aprašomi sąsajų pagalba. Kiekvienas komponentas pagal savo lygį gali kreiptis tik į žemesniame lygyje esantį komponentą. Trijų lygių architektūra leidžia geriausiai panaudoti visas objektinio programavimo priemones.

Prezentacijos lygis (vartotojo sąsaja (19 pav. (1)) ir vartotojo sąsajos valdiklis (19 pav. (2)));

Prezentacijos lygyje realizuojamas vartotojo sąsaja. Kuriamos formos skirtos duomenų vaizdavimui ir įvedimui. Kuriamo įrankio prezentacijos lygyje nėra realizuotos beveik jokios programos logikos, išskyrus tik duomenų įvedimo ir objektų vaizdavimo.

Veiklos komponentų lygis (pagrindinis lygis, valdantis pagrindines operacijas – patikros, pataisymų (3, 4, 6, 8));

Veiklos komponentų lygyje atliekamos visos skaičiavimo ir apdorojimo operacijos. Šis lygis – tai pagrindinė programos dalis. Iš vartotojo komponento į šį lygį perduodamos užduotys, ir pagal tai komponentas kreipiasi į duomenų lygį, kad išgautų teisingus atsakymus iš duomenų bazės.

Duomenų lygis (dirba tiesiogiai su duomenų šaltiniais (7))

Šis lygis skirtas klaidų aptikimui. Klaidos atveju išveda pranešimą, priešingai generuoja atsakymą.

4.6. Išvados

1. Tikslo įgyvendinimui pasirinkta Morfik FX For Object Pascal programa, kadangi ji visiškai palaiko Paskalio programavimo kalbą. Joje įdiegtos visos funkcijos, reikalingos Paskalio kodui įgyvendinti, pvz. kompiliatorius ir pan., todėl programos kodo įvykdymas tampa paprastesnis.

2. Morfik FX For Object Pascal programa naudoja priemones, skirtas maksimalias interaktyvumui pasiekti (AJAX, DHTML), tad įvestų duomenų tikrinimas neužtrunka;

3. **Programos kodo rašymo** metu vartotojas turi galimybę įvesti neteisingus duomenis, todėl programoje atliekama duomenų kontrolė. Vartotojo pranešimams naudojami paprasti, aiškūs ir informatyvūs sakiniai.

4. Kompiliavimo metu Morfik FX For Object Pascal siūlo pasirinkti projekto peržiūrą per: Internet Explorer, Mozilla FireFox, Samari, Opera, Chrome, tad kėblumų dėl atviros internetinės svetainės duomenų peržiūros neturėtų būti.

5. VARTOTOJO DOKUMENTACIJA

5.1. Svetainės realizacija

Sukurta svetainė patalpinta puslapiai.lt FTP serveryje. Adresas: <http://www.mdalia.projektas.lt>. Mokiniai ir kiti suinteresuoti asmenys gali apsilankyti šiuo adresu ir pasinaudoti šioje svetainėje esančia informacija, atlikti užduotis arba pasipraktikuoti rašant Paskalio kalbos kodą.

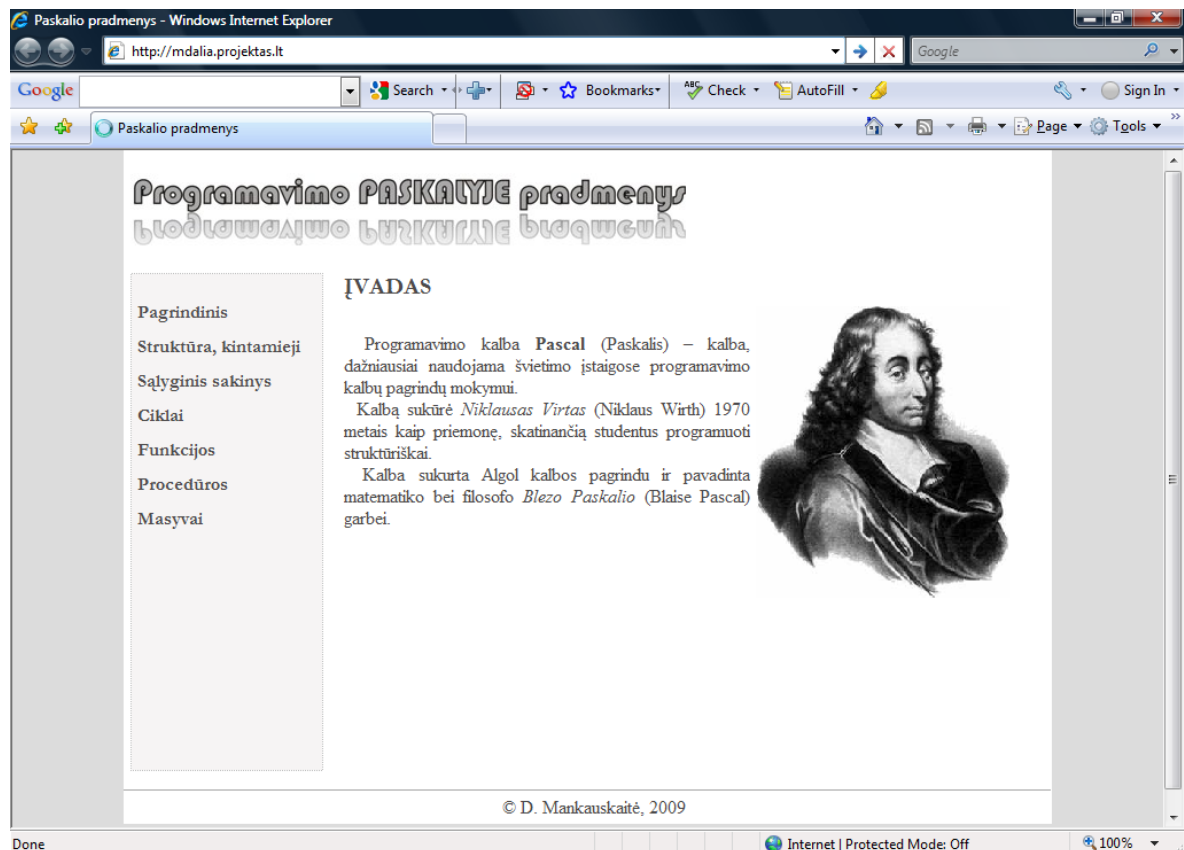
5.2. Vartotojo instrukcija

Svetainės paleidimui galima naudoti bet kurią iš išvardintų naršyklių:

1. Internet Explorer,
2. Mozilla,
3. Opera,
4. Chrome,
5. Safri.

Paleidus naršyklę adresu juostoje suvedamas adresas <http://www.mdalia.projektas.lt>.

Atsivėrusioje svetainėje matomas pradinis vaizdas (20 pav.)



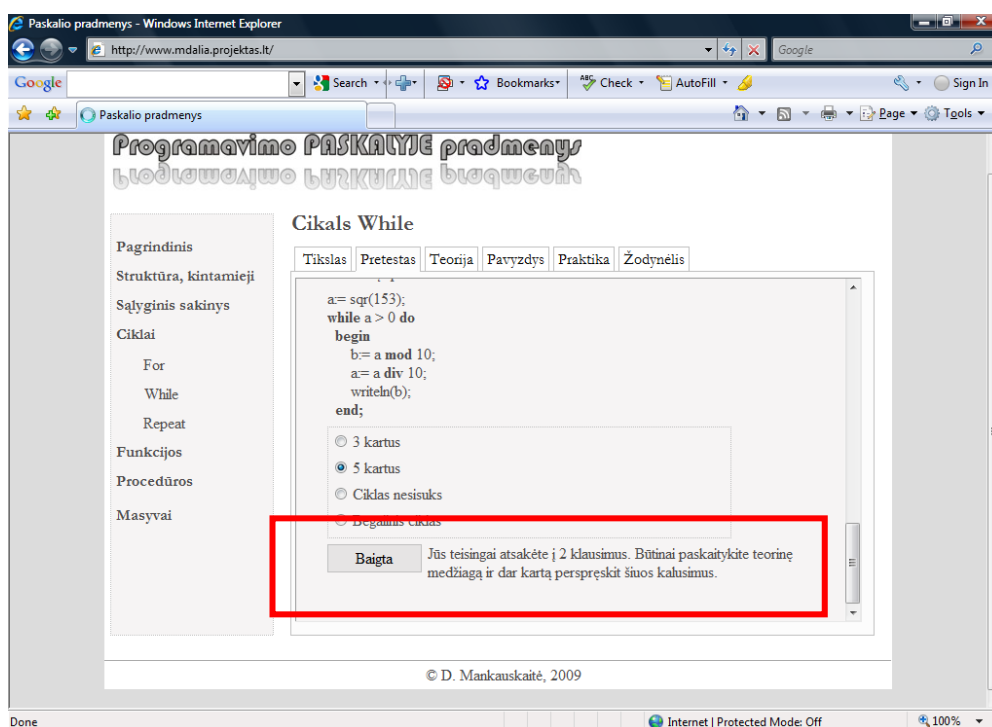
20 pav. Pagrindinis svetainės langas

Kairėje pusėje pateiktas pagrindinis meniu, kuriame galima pasirinkti norimą mokymosi sritį.

Pasirinkus norimą studijuoti sritį, dešinėje pusėje atveriamos kurso kortelės, kuriose pateikta jų pavadinimus atitinkanti medžiaga:

1. Tikslas – mokomosios dalies tikslas (žiūrėti 2 priede).
2. Pretestas – klausimai skirti žinių patikrinimui (žiūrėti 3 priede).

Sprendžiamas testas, kuriame galima pasirinkti vieną iš kelių atsakymų. Baigus spręsti spaudžiamas mygtukas „Baigta“. Paspaudus mygtuką išvedama informacija apie pasiektus rezultatus (21 pav.).



21 pav. Pretesto rezultatų apibendrinimo langas

3. Teorija – mokomoji medžiaga (žiūrėti 4 priede).
4. Pavyzdys – pateikiamas pavyzdys, atitinkantis mokomąją medžiagą (žiūrėti 5 priede).

Norint peržiūrėti kokią reikšmę grąžina pavyzdys, reikia spausti „Patikrinti“ mygtuką. Programa paprašys įvesti reikiamus duomenis. Įvedus duomenis spaudžiama bet kurioje lango vietoje arba ENTER klavišą, taip išvedamas rezultatas.

5. Praktika – sritis, kurioje galima patikrinti žinias suvedant programos kodą (žiūrėti 6 priede). Kodo įvedimo laukelyje suvedamas visas programos kodas. Jeigu norima rašyti naują programą, informacijos įvedimo laukelį galima išvalyti paspaudus mygtuką „Valyti“. Jeigu manoma, kad programos kodas suvestas teisingai ir ją galima vykdyti, tai spaudžiamas „Vykdyti“ mygtukas. Rezultatai išvedami „rezultatų išvedimas“ laukelyje.

6. Žodynėlis – pateikiami pagrindiniai programavimo kalbos žodžiai, naudojami mokomojoje medžiagoje (žiūrėti 7 priede).

5.3. Išvados

1. Sukurta nauja e. mokymosi svetainė, skirta Paskalio programavimo kalbos pradmenims mokytis, kurioje mokinys galės būti aktyvus veiksmo dalyvis.
2. Patikros testo duomenys kaupiami duomenų bazėje, todėl ateityje jį bus lengva atnaujinti.
3. Pavyzdžių rezultatų peržiūrai naudojami besimokančiojo pasirinkti duomenys, taip jam suteikiant galimybę geriau išanalizuoti pavyzdį ir įsisavinti mokomąją medžiagą.
4. Svetainėje yra programos tekstų redaktorių atitinkantis laukas, tad programos tekstui rinkti bei taisyti nereikia atskiro redaktoriaus.
5. Praktinėje dalyje, besimokančiojo parašytą Paskalio kalba programą Morfik FX transliatorius laiko pradiniais duomenimis ir ją perdirba į kitus duomenis – rezultatą. Vykdamas kreipimąsi į besimokančiojo suvestus duomenis, rezultatas pateikiamas kompiuteriui ir programa pradeda vykdyti.

6. SVETAINĖS TESTAVIMAS

Sėkmingam svetainės veikimo užtikrinimui daug laiko buvo skiriama testavimui. Svetainę testavo Alytaus profesinio rengimo centro mokiniai, atlikdami programavimo Paskalyje namų darbus.

Testuojant duomenis buvo tikrinama:

1. Ar suvestos mokomosios medžiagos pakanka užduočių sprendimui?

Sprendžiant patikros testus galima sugrįžti prie teorijos ir pagilinti žinias, jeigu sprendimo metu buvo padarytos klaidos.

2. Ar teisingai veikia pateikti pavyzdžiai?

Pavyzdžių peržiūros metu besimokinantys asmuo gali įvesti savo duomenis ir gauti rezultatą. Jeigu įvesti duomenys neatitinka pavyzdžio (pavyzdžiui, atliekami veiksmai su skaičiais, o įvedamas tekstas ir atvirksčiai), tai išvedamas pranešimas, kad įvykdyta duomenų įvedimo klaida. Rezultatai pateikiami 5 priede.

3. Ar sistemoje teisingai veikia kodo suvedimo patikra?

Suvedus programos kodą. Spaudžiamas mygtukas, kuris išsaugo suvestus duomenis ir sukompilavus grąžina rezultatą į langą. Jeigu kodas surašytas teisingai išvedamas atsakymas, priešingu atveju klaidos pranešimas. Rezultatai pateikiami 6 priede.

7. IŠVADOS

1. Išanalizavus internetines svetaines bei elektronines knygas, skirtas Paskalio programavimo kalbos mokymuisi, nuspręsta aktyviam interaktyviam programavimo pradmenų Paskalio kalboje mokymuisi pasirinkti tokias kūrimo aplinkas, kurių pagalba siekiamą galutinį maksimalų rezultatą gautumėm per kuo trumpesnę laiką.

2. Internetinės svetainės, skirtos programavimo pradmenų Paskalyje mokymuisi, projektavimo metu paaiškėjo, kad geriausiai darbui tinka Morfik FX Object Pascal programa, kuri yra nutaikyta į AJAX paraiškų aukšto lygio programavimo kalbas bei duomenų sisteminiui naudoja XML. Suvestas Paskalio programos kodas siunčiamas į Morfik FX kompiliatorių ir gaunamas greitas rezultatas. Morfik FX programoje programinis kodas kompiliuojamas Paskalio programavimo kalboje, todėl besimokančiojo suvesto programos kodo ir pavyzdžių duomenų patikrai sunkumų nebuvo.

3. Kiekvienas mokinys interaktyvioje svetainėje gali pasirinkti skyrių ir skyriuje esančią vieną iš 6 kortelių, kuriose pateikta informacija, skyriaus medžiagai įsisavinti bei galimybė parašyti savo duomenis. Taip kiekvienas besimokantysis gali susistatyti individualų mokymosi planą.

4. Suprojektuota ir sukurta duomenų bazė, testams saugoti. Ji leidžia pasiekti didesnę saugumo lygį, nei dirbant su bylomis.

5. Svetainės naudingumą išbandė Alytaus profesinio rengimo centro mokiniai, jie šioje svetainėje atliko programavimo Paskalyje namų darbus, bei teikė atsiliepimus ir patarimus apie svetainės veikimą.

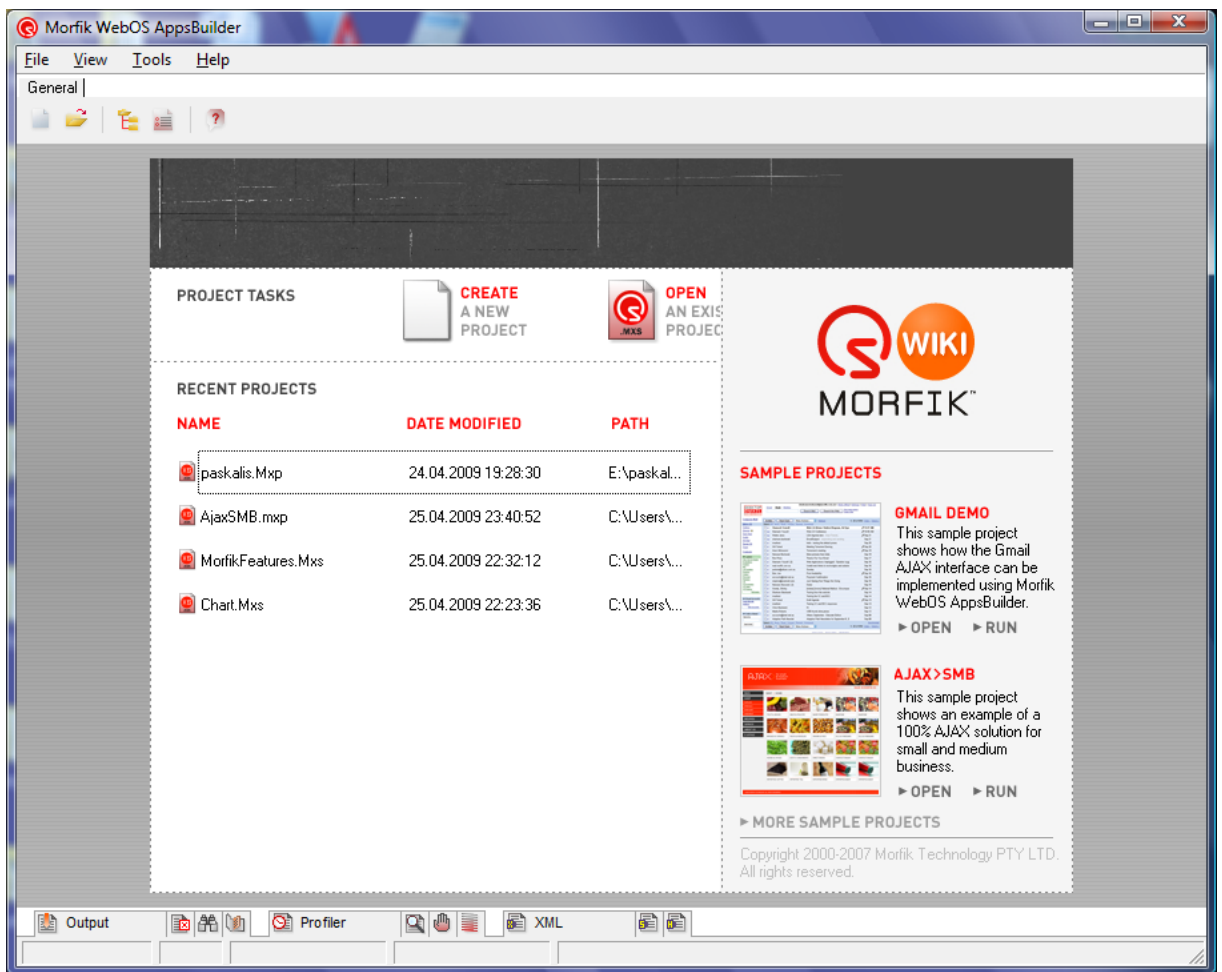
8. LITERATŪRA

1. Dagienė V., Švietimo naujienos, 2007 m. Nr. 11 (246), p. 8–9.
2. Balčytienė. A. Skaitmeninė kultūra humanitaroms. [Žiūrėta 2008-12-08]. Prieiga per internetą. <http://sk.mch.mii.lt/izanga1.html#kokybe>
3. Morfik. [Žiūrėta 2008-12-08]. Prieiga per internetą. <http://en.wikipedia.org/wiki/Morfik>
4. Būtaite D. Modulio „Programų sistemų inžinerijos įvadas“ laboratorinių darbų aiškinimas: UML diagramos. [Žiūrėta 2008-12-08]. Prieiga per internetą. http://e-stud.vgtu.lt/users/files/dest/3923/psi_uml%20diagramos_v01.ppt
5. Grigas G., elektroninė knyga „Programavimas Paskaliu“. [Žiūrėta 2008-12-08]. Prieiga per internetą. <http://www.ipc.lt/21z/mokymas/mokymopr/el/ek/knyga1.htm>
6. Doc. Dr. V. Undžėnas, „Paskalio pradmenys“. [Žiūrėta 2008-12-08]. Prieiga per internetą. http://62.80.232.136/informatika/valstybinis/paskalio_pradziamokslis.doc
7. „Programavimas Paskaliu“ (10 kl.). [Žiūrėta 2008-12-08]. Prieiga per internetą. <http://distance.ktu.lt/cdk/courses/908/fcontent.html>
8. V. Žemaitis, „Programavimo pradmenys“. [Žiūrėta 2008-12-09]. Prieiga per internetą. <http://tzc.andernetas.lt/tzg/informatika/pas10kl/>
9. Informatika. Nuotolinio mokymo sistema. [Žiūrėta 2008-12-09]. Prieiga per internetą. <http://vmc.ppf.ktu.lt/saulius/v2/index.php?puslapis=327>
10. Pascal Programming. [Žiūrėta 2008-12-08]. Prieiga per internetą. <http://www.geocities.com/SiliconValley/Horizon/5444/pstart.htm>
11. Pascal Programming. [Žiūrėta 2008-12-09]. Prieiga per internetą. <http://pascalprogramming.byethost15.com/index.php>
12. Vidžiūnas A., Blonskis J., Turbo Paskalis. Vartotojo vadovas. UAB Sekasoft, Vytauto Didžiojo universitetas. Kaunas, 1997.
13. Firebird. Server configuration and management. [Žiūrėta 2008-01-03]. Prieiga per internetą. <http://www.firebirdsql.org/manual/qsg2-config.html>
14. Morfik. [Žiūrėta 2009-01-03]. Prieiga per internetą. <http://morfik.com/#Home>

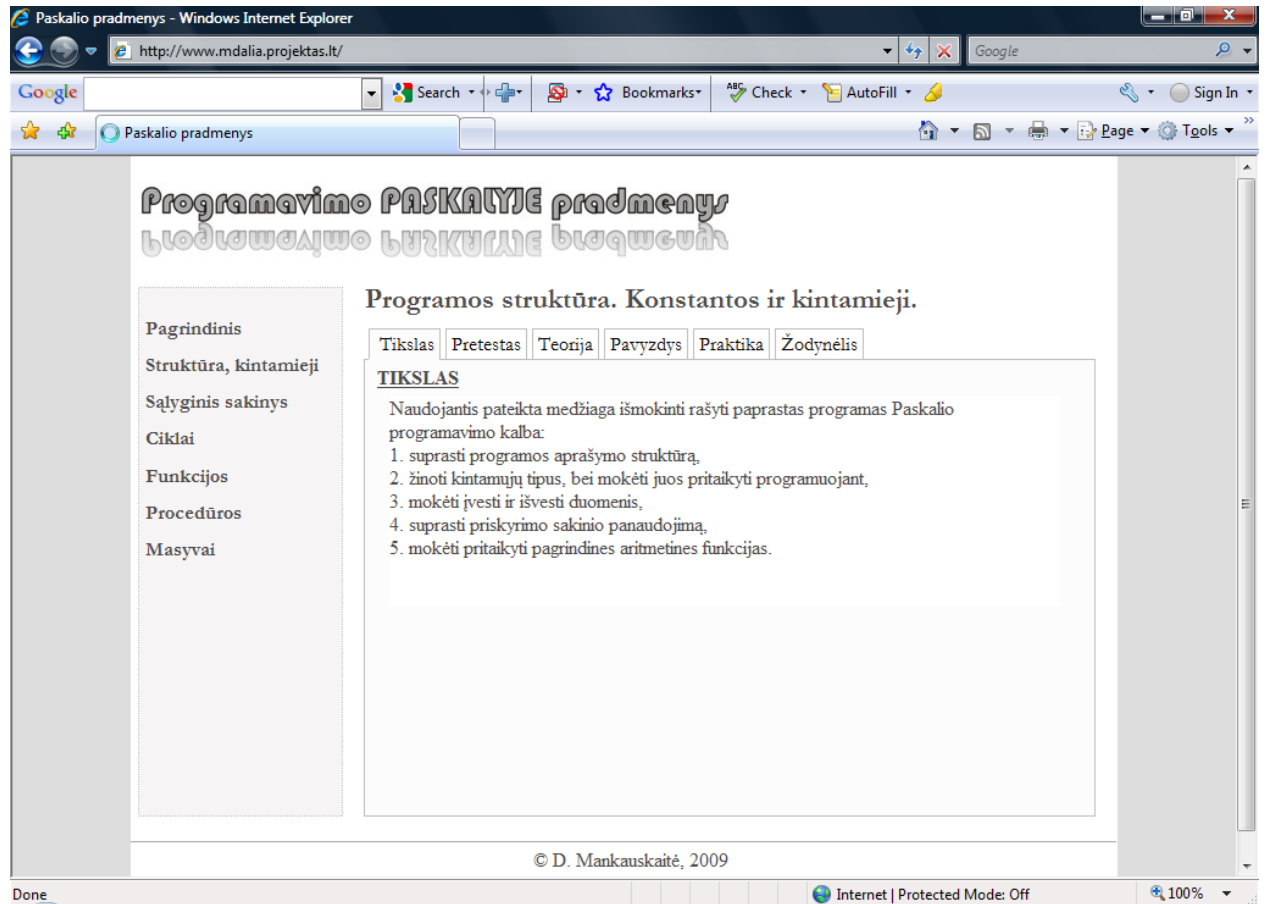
9. TERMINŲ IR SANTRUMPŲ ŽODYNAS

WWW	WWW (World Wide Web) arba pasaulinis voratinklis – tai naujas ir universalus informacijos pateikimo Internetė būdas.
Pascal	Paskalio programavimo kalba
Morfik FX	Tai programa, apimanti regimuosius projekto įrankius formas, užklausas, ataskaitas, lenteles, modulius, Web metodus, duomenų bazės struktūrai ir klausimams.
HTML	Hyper text Markup Language „Hiperteksto žymėjimo kalba“ – tai kompiuterinė žymėjimo kalba, naudojama pateikti turinį internete.
DHTML	Dinaminė HTML kalba, patobulintas HTML kalbos variantas, kai galima keisti tinklalapio turinį, vaizduojamą naršykle, atsižvelgiant į skaitytojo veiksmus, iš naujo neatsiunčiant tinklalapio iš serverio.
AJAX	AJAX – terminas, apibrėžiantis svetainių programavimo technologiją.
XML	XML (angl. eXtensible Markup Language) yra W3C rekomenduojama bendros paskirties duomenų struktūrų bei jų turinio aprašomoji kalba. Pagrindinė XML kalbos paskirtis yra užtikrinti lengvesnį duomenų keitimą tarp skirtingo tipo sistemų, dažniausiai sujungtų internetu.
LAMP	LAMP yra akronimas iš angl. žodžių: Linux, Apache, MySQL, PHP, naudojamas kalbant apie tipinę ir populiarią kompiuterio komplektaciją atviro kodo programine įranga, skirta internetinio serverio ir duomenų bazės valdymo funkcijoms atlikti.
MySQL	MySQL – viena iš reliacinių duomenų bazių valdymo sistemų (liet. santrumpa RDBVS, angl. – RDBMS), palaikanti daugelį naudotojų, dirbanti SQL kalbos pagrindu.
JAVA	Java – objektiškai orientuota programavimo kalba.

1 PRIEDAS. Programos Morfik FX for Object Pascal pradinis langas



2 PRIEDAS. Mokomosios dalies tikslo pateikimo langas.



Paskalio pradmenys - Windows Internet Explorer
http://www.mdalia.projektas.lt/

Google

Paskalio pradmenys

Programavimo PASKALYJE pradmenys

Προγραμματισμός ΒΑΣΙΚΕΣ ΒΙΒΛΙΟΓΡΑΦΙΕΣ

Programos struktūra. Konstantos ir kintamieji.

Tikslas | Pretestas | Teorija | Pavyzdys | Praktika | Žodynėlis

TIKSLAS

Naudojantis pateikta medžiaga išmokinti rašyti paprastas programas Paskalio programavimo kalba:

1. suprasti programos aprašymo struktūrą,
2. žinoti kintamųjų tipus, bei mokėti juos pritaikyti programuojant,
3. mokėti įvesti ir išvesti duomenis,
4. suprasti priskyrimo sakinio panaudojimą,
5. mokėti pritaikyti pagrindines aritmetines funkcijas.

© D. Mankauskaitė, 2009

Done Internet | Protected Mode: Off 100%

3 PRIEDAS. Pretesto pateikimo langas.

Paskalio pradmenys - Windows Internet Explorer
http://www.mdalia.projektas.lt/

Google Search Bookmarks Check AutoFill Sign In

Paskalio pradmenys

Programavimo PASKALYJE pradmenys

БІОДИВАЄЛІВІО БУЗКУСІЛІЄ БІОҚАСІН

Ciklas FOR

Tikslas Pretestas Teorija Pavyzdys Praktika Žodynėlis

Pagrindinis
Struktūra, kintamieji
Sąlyginis sakiny
Ciklai
For
While
Repeat
Funkcijos
Procedūros
Masyvai

1. Kas turi būti iš anksto žinoma, norint programose panaudoti ciklą for

- tokio ciklo nėra.
- ciklo „žingsnio“ dydis.
- ar pradinė reikšmė didesnė už galutinę.
- ciklo kartojimų skaičius.

2. Koks žodis naudojamas cikle for, jeigu jo reikšmės mažėja?

- to
- do
- downto
- begin

3. Kokia bus kintamojo a reikšmė įvykdžius šį programos fragmentą?

```
a:= 0;  
for i:= 1 to 2 do
```

© D. Mankauskaitė, 2009

Done Internet | Protected Mode: Off 100%

4 PRIEDAS. Mokomosios medžiagos pateikimo langas.

Paskalio pradmenys - Windows Internet Explorer

http://www.mdalia.projektas.lt/

Google

Paskalio pradmenys

Programavimo PASKALYJE pradmenys

БІОДИВАЇЛІВІ БУЗКУМІЕ БІОПАСУН

Programos struktūra. Konstantos ir kintamieji.

Tikslas | Pretestas | Teorija | Pavyzdys | Praktika | Žodynėlis


MOKOMOJI MEDŽIAGA

Paskalio kalba užrašytas baigtinis algoritmas, kurį atliekame kompiuteriu, vadinamas programa.

Algoritmas - tai aiškūs ir tikslūs nurodymai, kaip ir kokių veiksmų seka reikia atlikti norint pasiekti užsibrėžtą tikslą arba išspręsti suformuluotą uždavinį.

1 pavyzdys. Tai labai senas uždavinys, aprašytas dar VIII amžiuje.

Žmogui prireikė laivelio perkelti per upę vilką, ožką ir kopūstą. Tačiau mažame laivelyje galėjo tūpti tik žmogus ir su juo - arba vilkas, arba ožka, arba kopūstas. Tačiau palikęs vienus vilką su ožka, vilkas suėstų ožką, palikęs ožką su kopūstu, ožka sugrauztų kopūstą, o žmogaus akivaizdoje niekas nieko neliečia. Žmogus sėkmingai perkėlė savo krovinį per upę.



© D. Mankauskaitė, 2009

Done Internet | Protected Mode: Off 100%

5 PRIEDAS. Pavyzdžio pateikimo langas.

Paskalio pradmenys - Windows Internet Explorer
http://www.mdalia.projektas.lt/

Google

Paskalio pradmenys

Programavimo PASKALIJE pradmenys

ΛΙΟΔΙΩΑΑΑΛΛΛΛΛ ΛΥΡΚΥΥΛΛΛΛ ΛΙΣΩΩΩΩΩ

Pagrindinis
Struktūra, kintamieji
Sąlyginis sakinyss
Ciklai
 For
 While
 Repeat
Funkcijos
Procedūros
Masyvai

Ciklas Repeat

Tikslas | Pretestas | Teorija | Pavyzdys | Praktika | Žodynėlis

Programa tikrinanti ar teisingai įvestas slaptažodis. (Teisingas slaptažodis - juokas)

```
Program Slaptažodis;  
Var zodis, slaptas:string[9];  
Begin  
  slaptas:='juokas';  
  Repeat  
    Writeln('Koks slaptažodis?');  
    Readln(zodis);  
  Until zodis=slaptas;  
  Writeln('Teisingai');  
  Readln;  
end.
```

Patikrinti

Koks slaptažodis?

Neteisingai

© D. Mankauskaitė, 2009

Done Internet | Protected Mode: Off 100%

6 PRIEDAS. Patikros pateikimo langas.

#2 - Windows Internet Explorer
http://www.mdalia.projektas.lt/

Google

Programavimo PASKALYJE pradmenys
ԵՐԿՐԱՅԻՆԻ ԵՐԿՐԱՅԻՆԻ ԵՐԿՐԱՅԻՆԻ

Cikas FOR

Tikslas Pretestas Teorija Pavyzdys **Praktika** Žodynėlis

Kodo įvedimo laukas

```
Program pirmas;  
var x, y:integer;  
begin  
x:=5;  
for y:=1 to x do  
writeln(y);  
end.
```

Rezultatų išvedimas

```
1  
2  
3  
4  
5
```

Valyti Vykdyti

© D. Mankauskaitė, 2009

(1 item remaining) Internet | Protected Mode: Off 100%

22 pav. Teisingai suvesto kodo langas

#2 - Windows Internet Explorer
http://www.mdalia.projektas.lt/

Google

Programavimo PASKALYJE pradmenys
ԵՐԿՐԱՅԻՆԻ ԵՐԿՐԱՅԻՆԻ ԵՐԿՐԱՅԻՆԻ

Cikas FOR

Tikslas Pretestas Teorija Pavyzdys **Praktika** Žodynėlis

Kodo įvedimo laukas

```
Program pirmas;  
var x, y:real;  
begin  
x:=5;  
for y:=1 to x do  
writeln(y);  
end.
```

Rezultatų išvedimas

```
[Error]pr.pas[2]:For loop control  
variable must have ordinal type
```

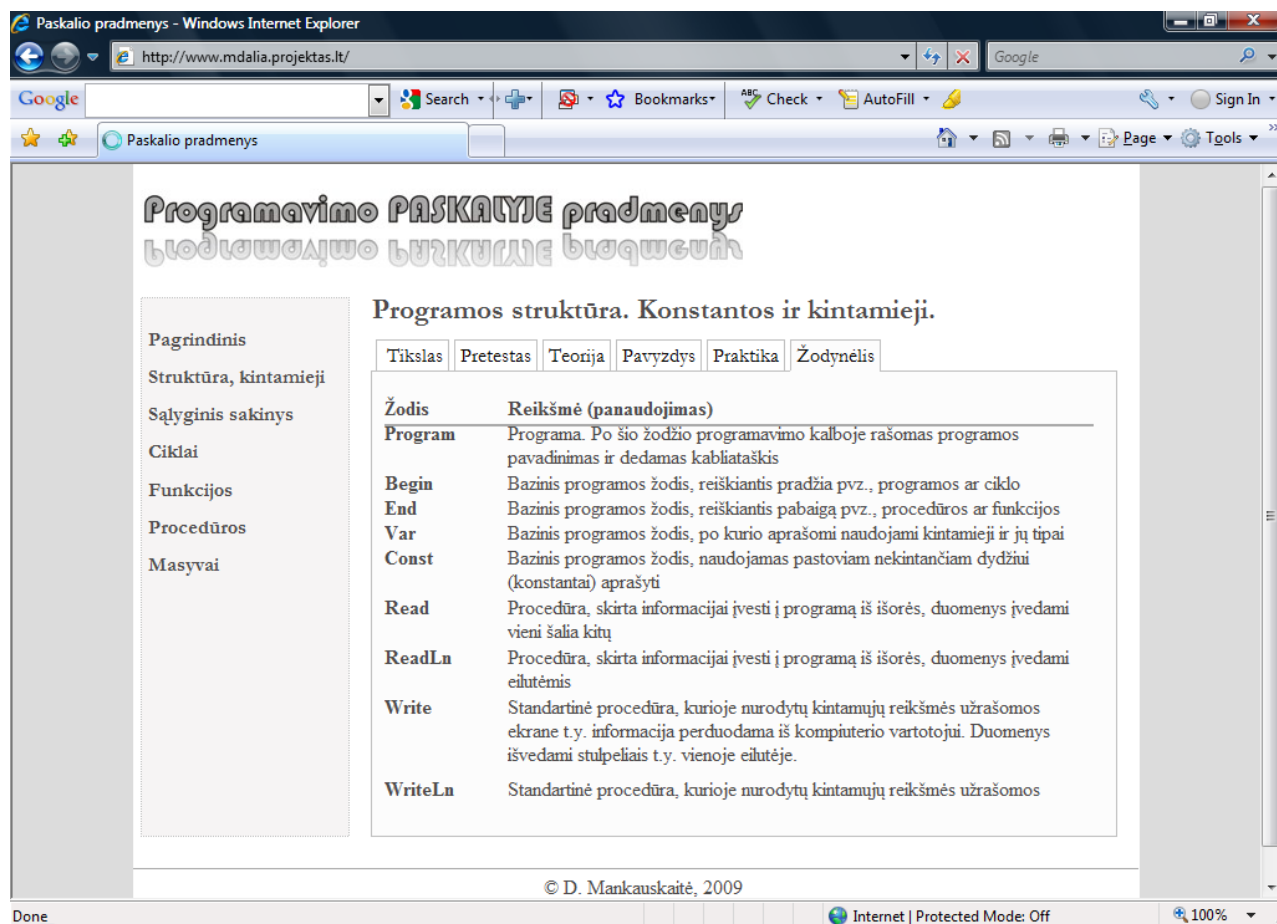
Valyti Vykdyti

© D. Mankauskaitė, 2009

(1 item remaining) Internet | Protected Mode: Off 100%

23 pav. Neteisingai suvesto kodo langas

7 PRIEDAS. Žodynėlio pateikimo langas.



Paskalio pradžmenys - Windows Internet Explorer
http://www.mdalia.projektas.lt/

Google

Paskalio pradžmenys

Programavimo PASKALYJE pradžmenys

Бодисааайао БУЗКУУУУЕ БУАУУУУ

Programos struktūra. Konstantos ir kintamieji.

Tikslas | Pretestas | Teorija | Pavyzdys | Praktika | Žodynėlis

Žodis	Reikšmė (panaudojimas)
Program	Programa. Po šio žodžio programavimo kalboje rašomas programos pavadinimas ir dedamas kabliataškis
Begin	Bazinis programos žodis, reiškiantis pradžia pvz., programos ar ciklo
End	Bazinis programos žodis, reiškiantis pabaigą pvz., procedūros ar funkcijos
Var	Bazinis programos žodis, po kurio aprašomi naudojami kintamieji ir jų tipai
Const	Bazinis programos žodis, naudojamas pastoviam nekintančiam dydžiui (konstantai) aprašyti
Read	Procedūra, skirta informacijai įvesti į programą iš išorės, duomenys įvedami vieni šalia kitų
ReadLn	Procedūra, skirta informacijai įvesti į programą iš išorės, duomenys įvedami eilutėmis
Write	Standartinė procedūra, kurioje nurodytų kintamųjų reikšmės užrašomos ekrane t.y. informacija perduodama iš kompiuterio vartotojui. Duomenys išvedami stulpeliais t.y. vienoje eilutėje.
WriteLn	Standartinė procedūra, kurioje nurodytų kintamųjų reikšmės užrašomos

© D. Mankauskaitė, 2009

Done Internet | Protected Mode: Off 100%