LATVIJAS UNIVERSITĀTE
DATORIKAS FAKULTĀTE

MĀJASLAPU IZVEIDES PAŠAPKALPOŠANĀS SISTĒMA

KVALIFIKĀCIJAS DARBS

Autors:
Dmitrijs Čudihins
Apl. Nr.: dc08029

Darba vadītājs:
Mg.dat. Krišs Rauhvargers

Rīga 2010

Anotācija

Kvalifikācijas darba mēris ir tīmekļa bāzētas mājaslapu konstruktora izstrāde. Sistēmas uzdevums ir dot iespēju izveidot, papildināt un aplūkot mājas lapas. Sistēma tika izstrādāta, ievērojot WG Agency izvirzītās prasības un vēlmes, lai atvieglotu un paātrinātu mājaslapas izstrādes procesu, tādēļ kvalifikācijas darba uzdevums ir, ievērojot Latvijas Valsts standartus programminženierijā un vadoties pēc industrijas labās prakses, radīt modulāru, mērogojamu un viegli modificējamu programmproduktu.

Sistēma izstrādāta kā tīmekļa vietne, izmantojot C#, ASP.NET MVC, SQL Server, Javascript un Ajax tehnoloģijas.

Atslēgvārdi: mājaslapu izstrāde, satura vadības sistēma, pašapkalpošanas sistēma

Abstract

The aim of this work is to develop a web-based website constructor. The purpose of this system is to create, manage and view websites. System was developed considering requirements and wishes expressed by WG Agency, to minimize time spent for website development thus, an essential additional goal is to follow the fine example of the industry as well as the program engineering standards of Latvia, in order to create a modular, scalable and easy to modify product.

The system is developed as a web site, using C#, ASP.NET MVC, SQL Server, JavaScript and Ajax technologies.

Keywords: website constructor, content management system.

SATURS

1. Ievads 1
2. Programmatūras prasību specifikācija 2
2.1. Ievads 2
2.1.1. Nolūks 2
2.1.2. Darbības sfēra 2
2.1.3. Definīcijas 2
2.1.4. Saistība ar citiem dokumentiem 3
2.1.5. Dokumenta pārskats 3
2.2. Vispārējs apraksts 3
2.2.1. Produkta funkcijas 3
2.2.2. Lietotāju raksturiezīmes 4
2.2.3. Vispārējie ierobežojumi 4
2.2.3.1. Tīkls 4
2.2.3.2. OS atbalsts un programmatūra 4
2.2.3.3. Aparatūra 4
2.3. Funkcionālās prasības 4
2.3.1. Lietotāju pārvaldības funkcijas 5
2.3.1.1. Jauna lietotāja reģistrācija 5
2.3.1.2. Pieteikšanas 5
2.3.1.3. Atteikšanas 6
2.3.1.4. Profila rediģēšana 7
2.3.1.5. Aizmirstas paroles maiņa 7
2.3.1.6. Administratora pieaicināšana 8
2.3.2. Mājaslapas pārvaldības funkcijas 9
2.3.2.1. Mājaslapas datu rediģēšana 9
2.3.3. Lapas pārvaldības funkcijas 10
2.3.3.1. Lapas pievienošana 10
2.3.3.2. Lapas saraksta attēlošana 10
2.3.3.3. Lapas saraksta secības maiņa 11
2.3.3.4. Lapas dzēšana 11
2.3.3.5. Informācijas bloka pievienošana 12
2.3.3.6. Informācijas bloka saturas maiņa 12
2.3.3.7. Lapas informācijas bloku secības maiņa 13

2.3.3.8. Standarta informācijas bloku pievienošana 13
2.3.3.9. Lapas kopēšana 14
2.3.3.10. Lapas informācijas apskatīšana 14
2.3.3.11. Lapas informācijas atjaunošana 15
2.3.3.12. Lapas informācijas bloka dzēšana 15
2.3.4. Noformējuma pārvaldības funkcijas 16
2.3.4.1. Mājaslapas logotipa augšupielāde 16
2.3.4.2. Lielas bildes augšupielāde 16
2.3.4.3. Noformējuma saraksta izvade 17
2.3.4.4. Noformējuma izvēle 17
2.3.4.5. Lielo bilžu saraksta izvade 18
2.3.4.6. Lielas bildes izvēle 18
2.3.4.7. Logotipa dzēšana 19
2.3.5. Drošības funkcijas 19
2.3.5.1. Lietotāja aizsargkoda izveidošana 19
2.3.5.2. Ievaddatu aizsargāšanas funkcija 20
2.3.5.3. Augšupielādēto failu pārbaude 20
2.4. Nefunkcionālas prasības 21
2.4.1. Datu drošība 21
2.4.2. Pieejamība 21
3. Programmatūras projektējuma apraksts 22
3.1. Nefunkcionālas prasības 22
3.1.1. Nolūks 22
3.1.2. Darbības sfēra 22
3.1.3. Definīcijas 22
3.2. Saistība ar citiem dokumentiem 22
3.3. Moduļu dekompozīcija 23
3.4. Starpprocesu atkarības 23
3.4.1. Datu plūsmas diagrammas 23
3.4.1.1. 0. līmeņa datu plūsmu diagramma - sistēmas kopskats 24
3.4.1.2. Pirmā līmeņa datu plūsmu diagramma 24
3.4.1.3. Otrā līmeņa datu plūsmu diagrammas 24
3.5. Datu detalizēts projektējums 25
3.5.1. Tabula Blocks 25
3.5.2. Tabula ContentTypes 26

3.5.3. HeaderImgs 26
3.5.4. HeaderSelectedImgs 26
3.5.5. ItemUploads 26
3.5.6. MemberProfiles 27
3.5.7. Members 27
3.5.8. MemberSites 27
3.5.9. SiteContents 28
3.5.10. StaticPageContents 28
3.5.11. StaticPages 28
3.5.12. TemplateBlocks 29
3.5.13. TemplateCategories 29
3.5.14. TemplateCategorySelected 29
3.5.15. Templates 29
3.5.16. TemplateTypes 30
3.5.17. WebsiteDomains 30
3.5.18. Websites 30
3.5.19. WebsiteStyles 31
3.6. Fiziskais ER modelis 32
3.7. Lietotāju saskarne 32
3.7.1. Lietotāja reģistrācija 32
3.7.2. Lietotāja pieteikšanas 33
3.7.3. Aizmirsta paroles maiņa 33
3.7.4. Profila rediģēšana 34
3.7.5. Administratora pieaicināšana 34
3.7.6. Mājaslapas datu rediģēšana 35
3.7.7. Jaunas lapas pievienošana 35
3.7.8. Lapas saraksta attēlošana 35
3.7.9. Lapas dzēšana 36
3.7.10. Informācijas bloka pievienošana 36
3.7.11. Informācijas bloka saturas maiņa 36
3.7.12. Lapas informācijas apskatīšana 37
3.7.13. Lapas informācijas bloka dzēšana 38
3.7.14. Mājaslapas logotipa augšupielāde 38
3.7.15. Lielas bildes augšupielāde 38
3.7.16. Noformējuma saraksta izvade 38

3.7.17. Lielo bilžu saraksta izvade 39
3.7.18. Logotipa dzēšana 39
4. Testēšanas dokumentācija 40
4.1. Testēšanas procesa apraksts 40
4.2. Vienībtestēšanas rezultāti 41
4.2.1. Jauna lietotāja reģistrācija 41
4.2.2. Pieteikšanas 42
4.2.3. Profila rediģēšana 43
4.2.4. Aizmirstas paroles maiņa 43
4.2.5. Administratora pieaicināšana 44
4.2.6. Mājaslapas datu rediģēšana 45
4.2.7. Lapas pievienošana 46
4.2.8. Lapas dzēšana 46
4.2.9. Informācijas bloka pievienošana 46
4.2.10. Informācijas bloka saturas maiņa 47
4.2.11. Lapas kopēšanas 47
4.2.12. Lapas informācijas atjaunošana 47
4.2.13. Lapas informācijas bloka dzēšana 47
4.2.14. Mājaslapas logotipa augšupielāde 48
4.2.15. Lielas bildes augšupielāde 48
4.2.16. Spriedzes testēšanu 48
5. Programmatūras projekta organizācija 50
6. Kvalitātes nodrošināšana 51
7. Konfigurāciju pārvaldība 52
8. Darbietilpības novērtējums 53
9. Programmatūras pirmkods 54

1. IEVADS

Uzsākot jauno biznesu, ir ļoti svarīgi paziņot par savu darbību internētā, tāda veida piesaistot jaunus klientus. Apskatot tirgus piedāvājumus, kļuva zināms, ka mājaslapas izstrādes process ir apgrūtinošs, laikietilpīgs un saistīts ar lielām izmaksām. Mājaslapu izveides pašapkalpošanās sistēma atvieglos šo procesu. Izmantojot MIPS fiziskām personām un uzņēmumiem radīsies iespēja neizejot no mājas vai biroja, pašiem izveidot sev mājaslapu, papildināt to ar tekstu, bildēm un citām, mājaslapai nepieciešamām lietām.

2. PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA

2.1. Ievads

2.1.1. Nolūks

Programmatūras prasību specifikācija (PPS) nolūks ir precīzi aprakstīt pasūtītāja un piegādātāja idejas par programmatūras prasībām. PPS mērķis ir aprakstīt, kas programmatūrai ir jādara, neaprakstot, kā tas ir jādara.

Šis dokuments ir paredzēts programmatūras pasūtītājiem. Dokuments nodrošina iespēju kontrolēt izstrādi un pārliecināties, ka sistēma būs atbilstīga prasībām. Izstrādājamajā sistēmā tika realizētas tikai un vienīgi tās prasības, kas iekļautas šajā dokumentā.

2.1.2. Darbības sfēra

Mājaslapu izveides pašapkalpošanās sistēma ir paredzēta informatīvas mājaslapu izveidei internetā, kur veicot reģistrāciju, izvēloties noformējumu un pievienojot informāciju var apskatīt jaunizveidoto mājaslapu. Sistēma būs paredzēta gan brīva piekļuve, gan tikai reģistrētiem lietotājiem, lai veiktu izmaiņas.

2.1.3. Definīcijas

Sistēma, turpmāk tekstā MIPS – objektu, procedūru, datubāzu vai paņēmienu kopums un to savstarpējās attiecības, kas funkcionāli veido vienotu veselumu.

Klients – persona, kas izmanto sistēmu ar minimālam piekļuves tiesībām.

Lietotājs – klients, kurš pieteicas sistēmā un veic izmaiņas mājas lapā.

Administrators – lietotājs, kas pārvalda sistēmu.

Atribūti – konkrēti lauki, kuri lietotājam ir jāaizpilda ar datiem, darbojoties „Mājaslapu izveides pašapkalpošanās sistēma”.

SQL Server 2005 – izmantotais datubāzes serveris.

C# – programmēšanas valoda, kādā veidota mājaslapu izveides pašapkalpošanās sistēma.

ASP.NET MVC Framework – Microsoft izstrādātais tīmekļa aplikāciju bibliotēku kopums, kas darbojas uz „Modelis-Skats-Kontrolleris” projektēšanas šablona.

IIS – izmantotais tīmekļa serveris.

Domēna vārds – pēc noteiktiem principiem veidotas un ar punktu atdalītas simbolu virknes, kas reprezentē ciparu formā izteiktu tīkla adresi.

2.1.4. Saistība ar citiem dokumentiem

Dokumenta sastādīšanā izmantots standarts LVS 68:1996 „Programmatūras prasību specifikācijas ceļvedis”, kas lietotajam garantē dokumenta standartu atbilstību.

2.1.5. Dokumenta pārskats

Dokuments sastāv no 4 nodaļām:

1. Ievads. Ievadinformācija, kas satur dokumenta nolūku, identificē programmatūras projektu, nepieciešamos skaidrojumus, saistību ar citiem dokumentiem;
2. Vispārējs apraksts. Izklāstīta informācijas par produkta funkcijām, lietotāja raksturiezīmēm, vispārējiem ierobežojumiem, pieņēmumiem un atkarībām;
3. Funkcionālās prasības. Detalizēti aprakstīti funkciju mērķi, ievaddati, apstrāde, izvaddati un kļūdu paziņojumi;
4. Nefunkcionālās prasības. Informācija ar datu drošību un sistēmas pieejamību.

2.2. Vispārējs apraksts

2.2.1. Produkta funkcijas

Sistēma paredzēta gan brīvai, gan iekšējai lietošanai. Iekšējai sistēmas izmantošanai nepieciešams veikt pieteikšanas, pēc kuras būs atļauts veikt vairākas izmaiņas un darbības:

- Jauna lietotāja reģistrācija;
- pieteikšanas;
- atteikšanas;
- aizmirstas paroles maiņa;
- jaunas mājaslapas izveides funkcija;
- domēna vārda piesaistes funkcija;
- mājaslapas ieslēgšanas un izslēgšanas funkcija;
- mājaslapas identificēšanas funkcijas;
- lapas pievienošanas funkcijas;
- lapas saraksta attēlošanas funkcija;
- lapas saraksta secības maiņas funkcija;
- lapas informācijas bloku secības maiņas funkcija;
- standarta informācijas bloku pievienošanas funkcija;
- lapas kopēšanas funkcija;
- lapas apskatīšanas funkcija;

- lapas rediģēšanas funkcija;
- lapas informācijas bloka dzēšana;
- logotipa augšupielādes funkcija;
- lielas bildes augšupielādes funkcija;
- noformējuma saraksta izvades funkcija;
- noformējuma izvēles funkcija;
- lielo bilžu izvades funkcija;
- lielas bildes izvēles funkcija;
- logotipa dzēšanas funkcija.

2.2.2. Lietotāju raksturiezīmes

Lietotājiem nepieciešamas pamatzināšanas darbā ar datoru un internētu.

2.2.3. Vispārējie ierobežojumi

2.2.3.1. Tīkls

Sistēmas darbībai ir nepieciešams interneta pieslēgums ar minimāli 256 kb/s ātrumu, lai lietotāji varētu pieslēgties sistēmai, ērti iegūt ielādējamās lapas saturu un aizpildāmās formas.

2.2.3.2. OS atbalsts un programmatūra

Sistēma tiek atbalstīta tikai Windows 2008 Server un Windows 2008 R2 Server operētājsistēmas vidē, uzstādīšanai nepieciešams datubāzes serveris SQL Server 2005 Express vai jaunāka versija, tīmekļa serveris IIS 7 vai jaunāka versija, .NET Framework 4 vai jaunāka versija, ASP.NET modulis IIS serverim. Papildus var izmantot datubāzu pārvaldības programmnodrošinājumu, piemēram, SQL Server Management Studio 2005, kurš darbojas Windows XP vai jaunāka versija.

2.2.3.3. Aparatūra

Sistēmas veiksmīgai darbībai rekomendēts vismaz 1.4GHz (x86 procesors) vai 2GHz (x64 procesors) ātrs centrālais procesors un 2GB liela operatīvā atmiņa. Cietajā diskā tiks aizņemti 20 MB vietas, kur glabāsies nepieciešamie faili, kas nodrošinās sistēmas darbību; nenoteikts brīvas vietas apjoms lietotāju augšupielādētiem failiem; nenoteikts brīvas vietas apjoms datubāzei, kur datubāzes izmērs būs atkarīgs no izveidoto lappušu skaitā.

2.3. Funkcionālās prasības

2.3.1. Lietotāju pārvaldības funkcijas

2.3.1.1. Jauna lietotāja reģistrācija

2.3.1.1.1. Mērķis:
Funkcija nodrošina jaunu lietotāju reģistrāciju.
2.3.1.1.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: Vārds, no 1 - 30 simboliem; E-pasts, no 1 - 40 simboliem, drīkst saturēt latīņu alfabēta lielos un mazos burtus, ciparus; Parole, no 1 - 30 simboliem, drīkst jebkādus simbolus; Parole atkārtoti, no 1 - 30 simboliem, drīkst jebkādus simbolus; Mājas lapas adrese, no 3 - 30 simboliem, drīkst saturēt latīņu alfabēta lielos un mazos burtus, kā arī ciparus no 0 – 9 (nav reģistrjutība); Mājas lapas nosaukums, līdz 30 simboliem.
2.3.1.1.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi, pārliecinās par e-pasta adreses pareizību un mājaslapas adreses unikalitāti un saglāba informāciju datubāzē.
2.3.1.1.4. Izvaddati:
Ja lietotājs ir korekti aizpildījis visus laukus, tiek paziņots, ka reģistrācija ir noritējusi veiksmīgi un nosūta automātiski ģenerētu reģistrācijas apstiprinājuma vēstuli uz lietotāja norādīto e-pasta adresi.
2.3.1.1.5. Kļūdu paziņojumi:
„Sekojošie lauciņi jāaizpilda obligāti:” „Izlasiet un piekrītiet pakalpojuma lietošanas noteikumiem.” „Ievadiet pareizu e-pasta adresi (piemērs: email@mail.com).” „Diemžēl tīmekļa lapa ar tādu adresi jau reģistrēta. Mēģiniet reģistrēt tīmekļa lapu ar citu nosaukumu.”.

2.3.1.2. Pieteikšanas

2.3.1.2.1. Mērķis:
Funkcija nodrošina klienta piekļuvi sistēmai lietotāja līmenī.
2.3.1.2.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: E-pasts, no 1 - 40 simboliem, drīkst saturēt latīņu alfabēta burtus, ciparus; Parole, no 5 - 30 simboliem, drīkst saturēt jebkādus simbolus; Mājaslapas adrese, līdz 3 - 30 simboliem, drīkst saturēt latīņu alfabēta lielos un mazos burtus, kā arī ciparus no 0 - 9 (nav reģistrjutība, nav obligāta);
2.3.1.2.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un pārbauda vai eksistē norādītais lietotājs.
2.3.1.2.4. Izvaddati:
Ja klients korekti aizpildījis visus laukus, tas tiek piesaistīts lietotāja sesijai un pāradresēts administrācijas daļā.
2.3.1.2.5. Kļūdu paziņojumi:
„Lai ieietu, jāaizpilda laukums "E-pasts" un "Parole".” „Lietotājs ar šādu e-pasta adresi un paroli neeksistē. Lūdzu, pārbaudiet ievadītos datus.”

2.3.1.3. Atteikšanas

2.3.1.3.1. Mērķis:
Funkcija nodrošina lietotāja sesijas pārtraukšanu un sesijas datu dzēšanu.
2.3.1.3.2. Ievaddati:
-
2.3.1.3.3. Apstrāde:
Funkcija veic sesijas pārtraukšanu un sesijas datu dzēšanu no lietotāja pārlūkprogrammas.
2.3.1.3.4. Izvaddati:
„Jūs veiksmīgi izgājāt no satura vadības sistēmas.”
2.3.1.3.5. Kļūdu paziņojumi:

-

2.3.1.4. Profila rediģēšana

2.3.1.4.1. Mērķis:
Funkcija nodrošina lietotājam piekļuvi datu izmaiņu formai, kuras laukos ir norādīta pašreizējā informācija.
2.3.1.4.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: Vārds, no 1 - 30 simboliem, drīkst saturēt latīņu alfabēta lielos un mazos burtus, ciparus; Parole, no 5 - 30 simboliem, drīkst saturēt latīņu alfabēta lielos un mazos burtus, kā arī ciparus no 0 - 9 (ir reģistrjutība, nav obligāts); Parole atkārtoti, no 5 - 30 simboliem, drīkst saturēt latīņu alfabēta lielos un mazos burtus, kā arī ciparus no 0 – 9 (ir reģistrjutība, nav obligāts); Uzņēmuma nosaukums, līdz 30 simboliem (nav obligāts); Amats, līdz 30 simboliem (nav obligāts); Adrese, līdz 100 simboliem (nav obligāts); Tālrunis, līdz 15 simboliem (nav obligāts);
2.3.1.4.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un lietotāja pilnībā vai daļēji aizstāt ar jauniem datiem.
2.3.1.4.4. Izvaddati:
Pēc sekmīgas profila rediģēšanas tiek paziņots, ka dati ir izmainīti. „Jūsu informācija veiksmīgi atjaunota.”
2.3.1.4.5. Kļūdu paziņojumi:
„Sekojošie lauciņi jāaizpilda obligāti:” „Jūs neesat atkārtoti ievadījis paroli.” „Ievadītās paroles nesakrīt.”

2.3.1.5. Aizmirstas paroles maiņa

2.3.1.5.1. Mērķis:
Funkcija nodrošina jaunas lietotāja paroles izveidošanu.
2.3.1.5.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: E-pasts, no 6 - 40 simboliem, drīkst saturēt latīņu alfabēta lielos un mazos burtus, ciparus.
2.3.1.5.3. Apstrāde:
Funkcija pārbauda vai eksistē lietotājs ar noradīto e-pasta adresi un nosūta automātiski ģenerētu vēstuli uz norādīto e-pasta adresi, kas satur jaunus lietotāja piekļuves datus.
2.3.1.5.4. Izvaddati:
„Izveidota jauna parole ir nosūtīta Jums uz norādīto e-pasta adresi.”
2.3.1.5.5. Kļūdu paziņojumi:
„Ievadiet pareizu e-pasta adresi (piemērs: email@mail.com).” „Jūs ievadījāt nepareizu e-pasta adresi. Lietotājs ar šādu e-pasta adresi nepastāv.”

2.3.1.6. Administratora pieaicināšana

2.3.1.6.1. Mērķis:
Funkcija nodrošina iespēju pieaicināt jaunu administratoru mājaslapai.
2.3.1.6.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: Vārds, no 1 – 30 simboliem (nav obligāts); E-pasts, no 1 - 40 simboliem, drīkst saturēt latīņu alfabēta lielos un mazos burtus, ciparus.
2.3.1.6.3. Apstrāde:
Funkcija pārliecinās par datu pareizību un nosūta ielūgumu uz noradīto e-pasta adresi pievienoties mājaslapas izveides procesam.
2.3.1.6.4. Izvaddati:
„Uzaicinājums veiksmīgi nosūtīts. Tiklīdz lietotājs pievienosies mājas lapai, Jūs ieraudzīsiet viņa ikoniņu šajā sadaļā.”

2.3.1.6.5. Kļūdu paziņojumi:
„Ievadiet pareizu e-pasta adresi (piemērs: email@mail.com).”

2.3.2. Mājaslapas pārvaldības funkcijas

2.3.2.1. Mājaslapas datu rediģēšana

2.3.2.1.1. Mērķis:
Funkcija nodrošina piekļuvi mājaslapas datu izmaiņu formai, kuras laukos ir norādīta pašreizējā informācija.
2.3.2.1.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: Domēna vārds, no 5 – 25 simboliem (nav obligāts); Nosaukums, līdz 40 simboliem, drīkst saturēt latīņu alfabēta lielos un mazos burtus, ciparus (nav obligāts); Apraksts, līdz 300 simboliem (nav obligāts); E-pasts, no 1 - 40 simboliem, drīkst saturēt latīņu alfabēta lielos un mazos burtus, ciparus (nav obligāts); Statuss („izslēgts” vai „ieslēgts”).
2.3.2.1.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un pārliecinās par mājaslapas adreses unikalitāti. Ja dati ir korekti, sistēma saglāba jaunu mājaslapu datubāzē.
2.3.2.1.4. Izvaddati:
„Jūsu informācija veiksmīgi atjaunota.”
2.3.2.1.5. Kļūdu paziņojumi:
„Sekojošie lauciņi jāaizpilda obligāti:” „Ievadiet pareizu e-pasta adresi (piemērs: email@mail.com).” „Šis domēna vārds tiek izmantots citā projektā.” „Nepareizs domēna vārds.”

2.3.3. Lapas pārvaldības funkcijas

2.3.3.1. Lapas pievienošana

2.3.3.1.1. Mērķis:
Funkcija nodrošina jaunas lapas izveidi.
2.3.3.1.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: Nosaukums, līdz 40 simboliem, drīkst saturēt latīņu alfabēta lielos un mazos burtus, ciparus; Apraksts, līdz 300 simboliem (nav obligāts); Statuss („izslēgts” vai „ieslēgts”).
2.3.3.1.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un saglāba jaunu lapu datubāze.
2.3.3.1.4. Izvaddati:
Funkcija veic ievaddatu korektuma pārbaudi un saglāba jaunu lapu datubāzē.
2.3.3.1.5. Kļūdu paziņojumi:
„Lauciņš "Nosaukums" ir obligāti jāaizpilda.”

2.3.3.2. Lapas saraksta attēlošana

2.3.3.2.1. Mērķis:
Funkcija attēlo visas mājaslapas lapas dilstoša saraksta.
2.3.3.2.2. Ievaddati:
Mājaslapas ID
2.3.3.2.3. Apstrāde:
Veic datubāzes pieprasījumu iekļaujot norādīto ID.
2.3.3.2.4. Izvaddati:
Visas mājaslapas lapas dilstoša saraksta.
2.3.3.2.5. Kļūdu paziņojumi:

-

2.3.3.3. Lapas saraksta secības maiņa

2.3.3.3.1. Mērķis:
Funkcija nodrošina lapas saraksta secības maiņu.
2.3.3.3.2. Ievaddati:
Masīvs (Vecākas lapas ID, Lapas ID)
2.3.3.3.3. Apstrāde:
Atjauno visas mājaslapas lapas norādot vecākas lapa ID un kārtas numuru.
2.3.3.3.4. Izvaddati:
Atjaunotais lapas saraksts.
2.3.3.3.5. Kļūdu paziņojumi:
-

2.3.3.4. Lapas dzēšana

2.3.3.4.1. Mērķis:
Funkcija dzēš noteikto lapu.
2.3.3.4.2. Ievaddati:
Lapas ID
2.3.3.4.3. Apstrāde:
Lapa ir dzēsta no datubāzes.
2.3.3.4.4. Izvaddati:
Atjaunotais lappušu saraksts.
2.3.3.4.5. Kļūdu paziņojumi:
„Lapa nav atrasta.”

2.3.3.5. Informācijas bloka pievienošana

2.3.3.5.1. Mērķis:
Funkcija nodrošina informācijas bloka pievienošanu lapā.
2.3.3.5.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: Nosaukums, līdz 100 simboliem, drīkst saturēt latīņu alfabēta lielos un mazos burtus, ciparus; Pozīcijas ID , vesels skaitlis (atkarīgs no lapas noformējuma tipa); Tips; Saturs, teksts (nav obligāts).
2.3.3.5.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un saglāba jaunu informācijas bloku datubāzē.
2.3.3.5.4. Izvaddati:
OK
2.3.3.5.5. Kļūdu paziņojumi:
„Lapa nav atrasta.” „Lauciņš „Elementa nosaukums” ir obligāti jāaizpilda.”

2.3.3.6. Informācijas bloka saturas maiņa

2.3.3.6.1. Mērķis:
Funkcija nodrošina iespēju mainīt informācijas bloka saturu.
2.3.3.6.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: Informācijas bloka ID, vesels skaitlis; Saturs, teksts (nav obligāts).
2.3.3.6.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un saglāba izmaiņas datubāzē.
2.3.3.6.4. Izvaddati:

OK
2.3.3.6.5. Kļūdu paziņojumi:
„Informācijas bloks nav atrasts.” „Lauciņš „Elementa nosaukums” obligāti jāaizpilda.”

2.3.3.7. Lapas informācijas bloku secības maiņa

2.3.3.7.1. Mērķis:
Funkcija nodrošina lapas informācijas bloku secības maiņu.
2.3.3.7.2. Ievaddati:
Masīvs (Lapas ID, Pozīcijas ID, Bloka ID)
2.3.3.7.3. Apstrāde:
Atjauno visas lapas informācijas blokus norādot to secības numuru, kā kārtas numuru masīva un saglāba izmaiņas datubāzē.
2.3.3.7.4. Izvaddati:
Atjaunotais lapas informācijas bloku saraksts.
2.3.3.7.5. Kļūdu paziņojumi:
„Lapa nav atrasta.” „Informācijas bloks nav atrasts.”

2.3.3.8. Standarta informācijas bloku pievienošana

2.3.3.8.1. Mērķis:
Funkcija pievieno standarta informācijas bloku lapai.
2.3.3.8.2. Ievaddati:
Lapas ID
2.3.3.8.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un pievieno jaunus informācijas blokus lapā.
2.3.3.8.4. Izvaddati:

OK
2.3.3.8.5. Kļūdu paziņojumi:
„Lapa nav atrasta.”

2.3.3.9. Lapas kopēšana

2.3.3.9.1. Mērķis:
Funkcija nodrošina lapas kopēšanu kopā ar visiem informācijas blokiem.
2.3.3.9.2. Ievaddati:
Lapas ID
2.3.3.9.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un izveido lapas kopiju.
2.3.3.9.4. Izvaddati:
OK
2.3.3.9.5. Kļūdu paziņojumi:
„Lapa nav atrasta.”

2.3.3.10. Lapas informācijas apskatīšana

2.3.3.10.1. Mērķis:
Funkcija nodrošina lapas informācijas apskatīšanu.
2.3.3.10.2. Ievaddati:
Lapas ID
2.3.3.10.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un pievieno jaunus informācijas blokus lapā
2.3.3.10.4. Izvaddati:
Lapas informācija (lapas id, lapas nosaukums, lapas apraksts, lapas statuss); informācijas bloku masīvs (nosaukums, pozīcijas id, kārtas numurs, saturs).

2.3.3.10.5. Kļūdu paziņojumi:
„Lapa nav atrasta.”

2.3.3.11. Lapas informācijas atjaunošana

2.3.3.11.1. Mērķis:
Funkcija nodrošina izveidotas lapas datu atjaunošanu.
2.3.3.11.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: Nosaukums, līdz 40 simboliem, drīkst saturēt latīņu alfabēta lielos un mazos burtus, ciparus; Apraksts, līdz 300 simboliem (nav obligāts); Statuss („izslēgts” vai „ieslēgts”).
2.3.3.11.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un saglāba izmaiņas datubāzē.
2.3.3.11.4. Izvaddati:
OK
2.3.3.11.5. Kļūdu paziņojumi:
„Lauciņš "Nosaukums" ir obligāti jāaizpilda.”

2.3.3.12. Lapas informācijas bloka dzēšana

2.3.3.12.1. Mērķis:
Funkcija nodrošina lapas informācijas bloku dzēšanu.
2.3.3.12.2. Ievaddati:
Informācijas bloka ID
2.3.3.12.3. Apstrāde:
Dzēš norādīto informācijas bloku no datubāzes.
2.3.3.12.4. Izvaddati:

OK
2.3.3.12.5. Kļūdu paziņojumi:
„Lapa nav atrasta.”

2.3.4. Noformējuma pārvaldības funkcijas

2.3.4.1. Mājaslapas logotipa augšupielāde

2.3.4.1.1. Mērķis:
Funkcija nodrošina mājaslapas logotipa augšupielādi.
2.3.4.1.2. Ievaddati:
Nepieciešams nodrošināt iespēju augšupielādēt failu.
2.3.4.1.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un saglāba izmaiņas datubāzē.
2.3.4.1.4. Izvaddati:
OK
2.3.4.1.5. Kļūdu paziņojumi:
„Neizdevās ielādēt failu. Pārbaudiet faila izmēru (līdz 1 Mb) un veidu bilde).”

2.3.4.2. Lielas bildes augšupielāde

2.3.4.2.1. Mērķis:
Funkcija nodrošina lielas bildes augšupielādi.
2.3.4.2.2. Ievaddati:
Nepieciešams nodrošināt iespēju augšupielādēt failu.
2.3.4.2.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un saglāba izmaiņas datubāze..
2.3.4.2.4. Izvaddati:

OK
2.3.4.2.5. Kļūdu paziņojumi:
„Neizdevās ielādēt failu. Pārbaudiet faila izmēru (līdz 1 Mb) un veidu (bilde).”

2.3.4.3. Noformējuma saraksta izvade

2.3.4.3.1. Mērķis:
Funkcija nodrošina mājaslapas noformējuma saraksta izvadi.
2.3.4.3.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: Kategorijas ID, vesels skaitlis (nav obligāts).
2.3.4.3.3. Apstrāde:
Veic datubāzes pieprasījumu iekļaujot norādītas kategorijas ID.
2.3.4.3.4. Izvaddati:
Noformējuma saraksts.
2.3.4.3.5. Kļūdu paziņojumi:
-

2.3.4.4. Noformējuma izvēle

2.3.4.4.1. Mērķis:
Funkcija atjauno mājaslapas noformējuma stilu.
2.3.4.4.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: Noformējuma ID, vesels skaitlis.
2.3.4.4.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un atjauno mājaslapas noformēja stilu.
2.3.4.4.4. Izvaddati:

OK
2.3.4.4.5. Kļūdu paziņojumi:
„Noformējums nav atrasts.”

2.3.4.5. Lielo bilžu saraksta izvade

2.3.4.5.1. Mērķis:
Funkcija nodrošina mājaslapas lielo bilžu saraksta izvadi.
2.3.4.5.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: Kategorijas ID, vesels skaitlis (nav obligāts).
2.3.4.5.3. Apstrāde:
Veic datubāzes pieprasījumu iekļaujot norādītas kategorijas ID.
2.3.4.5.4. Izvaddati:
Lielo bilžu saraksts.
2.3.4.5.5. Kļūdu paziņojumi:
-

2.3.4.6. Lielas bildes izvēle

2.3.4.6.1. Mērķis:
Funkcija atjauno mājaslapas lielo bildi.
2.3.4.6.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: Lielas bildes ID, vesels skaitlis.
2.3.4.6.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un saglāba izmaiņas.
2.3.4.6.4. Izvaddati:

OK
2.3.4.6.5. Kļūdu paziņojumi:
„Bilde nav atrasta.”

2.3.4.7. Logotipa dzēšana

2.3.4.7.1. Mērķis:
Funkcija dzēš mājaslapas logotipu.
2.3.4.7.2. Ievaddati:
Nepieciešams nodrošināt iespēju ievadīt šādu informāciju: Lielas bildes ID, vesels skaitlis.
2.3.4.7.3. Apstrāde:
Funkcija veic ievaddatu korektuma pārbaudi un dzēš mājaslapas logotipu.
2.3.4.7.4. Izvaddati:
OK
2.3.4.7.5. Kļūdu paziņojumi:
-

2.3.5. Drošības funkcijas

2.3.5.1. Lietotāja aizsargkoda izveidošana

2.3.5.1.1. Mērķis:
Funkcija izveido specialo kodu atkarībā no lietotāja e-pasta adreses, paroles, internēta pārlūkprogrammas uzstādījumiem un IP adreses.
2.3.5.1.2. Ievaddati:
Lietotāja ID; Lietotāja paroles MD5 simbolu virkne, Lietotāja pārlūkprogrammas informācija; Lietotāja IP adrese;

Mājaslapas ID.
2.3.5.1.3. Apstrāde:
Visi ievaddati ir savienoti virknē pēc noteikta principa un ir pārtaisīti jauna, 32 simbolu garā MD5 virkne.
2.3.5.1.4. Izvaddati:
32 simbolu gara virkne.
2.3.5.1.5. Kļūdu paziņojumi:
-

2.3.5.2. Ievaddatu aizsargāšanas funkcija

2.3.5.2.1. Mērķis:
Funkcija aizsarga simbolu virkni no potenciāli bīstamiem HTML koda fragmentiem un skriptu injekcijām.
2.3.5.2.2. Ievaddati:
Simbolu virkne.
2.3.5.2.3. Apstrāde:
Simbolu virkne ir apstrādāta un aizsargāta no potenciāli bīstamiem HTML koda gabaliem un skriptu injekcijām.
2.3.5.2.4. Izvaddati:
Apstrādāta simbolu virkne.
2.3.5.2.5. Kļūdu paziņojumi:
-

2.3.5.3. Augšupielādēto failu pārbaude

2.3.5.3.1. Mērķis:
Funkcija pārbauda augšupielādēto failu atbilstību pieļautajiem formātiem un faila izmēra ierobežojumiem.

2.3.5.3.2. Ievaddati:
Fails.
2.3.5.3.3. Apstrāde:
Sistēma pārbauda faila MIME tipu, failā formātu un faila izmēra atbilstību noteiktām prasībām.
2.3.5.3.4. Izvaddati:
Apstrādāta simbolu virkne.
2.3.5.3.5. Kļūdu paziņojumi:
„Neizdevās ielādēt failu. Pārbaudiet faila izmēru (līdz 1 Mb) un veidu (ilustrācija, biroja dokuments, arhīvs).”

2.4. Nefunkcionālas prasības

2.4.1. Datu drošība

Sistēma atbilst visām „Fizisko personu datu aizsardzības likums” prasībām.

Sistēma veic pārbaudi ar Javascript palīdzību, kā arī pārbaudes ir veiktas servera pusē, kas nodrošina datu drošību.

2.4.2. Pieejamība

Sistēma un izstrādātas mājaslapas ir brīvi pieejamas apskatei katram iedzīvotājam. Lai veikt jebkuras izmaiņas mājaslapā, klientam ir jābūt pieteiktam izmantojot reģistrācijas laikā norādītos datus.

3. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS

3.1. Nefunkcionālas prasības

3.1.1. Nolūks

Mājaslapu izveides pašapkalpošanās sistēmas programmatūras projektējuma apraksta nolūks ir aprakstīt, kā dotās sistēmas programmatūras prasību specifikācijā iekļautās prasības tiks realizētas programmatūras izstrādes vidē, lai atvieglotu programmatūras projektēšanu un izstrādi.

3.1.2. Darbības sfēra

Mājaslapu izveides pašapkalpošanās sistēma ir paredzēta informatīvas mājaslapu izveidei internetā, kur veicot reģistrāciju, izvēloties noformējumu un pievienojot informāciju var apskatīt jaunizveidoto mājaslapu. Sistēma būs paredzēta gan brīva pieeja, gan tikai pieteiktiem lietotājiem.

3.1.3. Definīcijas

ER – Entity Relationship modelis attēlo datubāzes entītijuentes savstarpējās atkarības;

DPD – Datu Plūsmu Diagramma ir grafisks, informācijas sistēmas datu plūsmas attēlojums;

PK – Primary Key jeb primārā atslēga ir unikāls identifikators, kaskurš identificē kādu datubāzes tabulas ierakstu;

FK – Foreign Key ir saites atslēga, kas tiek izmantota kā norāde uz attiecīgo ierakstu kādā citā datubāzes tabulā;

Int – veselu skaitļu datu tips;

Text – rakstzīmju datu tips, kas var ietvert sevī 2,147,483,647 rakstzīmes.

Klients – persona, kas izmanto sistēmu ar minimālam piekļuves tiesībām.

Lietotājs – klients, kurš pieteicas sistēmā un veic izmaiņas mājas lapā.

Administrators – lietotājs, kas pārvalda sistēmu.

3.2. Saistība ar citiem dokumentiem

Šis programmatūras projektējuma apraksts ir izstrādāts uz „Mājaslapu izveides pašapkalpošanās sistēmas” programmatūras prasību specifikācijas 1.0 versijas bāzes un Latvijas standarta LVS 72:1996.

3.3. Moduļu dekompozīcija

Tabula	Apraksts
Blocks	Tabulas mērķis ir glabāt rediģējamo bloku sarakstu.
ContentTypes	Tabulas mērķis ir glabāt satura tipus.
HeaderImgs	Tabulas mērķis ir glabāt visas pieejamas augšienes.
HeaderSelectedImgs	Tabulas mērķis ir glabāt saites starp augšieni un noformējuma kategoriju.
ItemUploads	Tabulas mērķis ir glabāt augšupielādēto failu sarakstu.
MemberProfiles	Tabulas mērķis ir glabāt lietotāja profila informāciju.
Members	Tabulas mērķis ir glabāt lietotāja pamatdatus.
MemberSites	Tabulas mērķis ir glabāt saiti starp lietotāju un mājaslapu, kuru viņš administrē.
SiteContents	Tabulas mērķis ir glabāt satura bloka informāciju.
StaticPageContents	Tabulas mērķis ir glabāt saiti starp satura bloku un lapu.
StaticPages	Tabulas mērķis ir glabāt lapu sarakstu.
TemplateBlocks	Tabulas mērķis ir glabāt noformējuma bloku sarakstu.
TemplateCategories	Tabulas mērķis ir glabāt mājaslapas noformējuma kategorijas sarakstu.
TemplateCategorySelected	Tabulas mērķis ir glabāt saites starp noformējumu un kategoriju.
Templates	Tabulas mērķis ir glabāt noformējuma sarakstu.
TemplateTypes	Tabulas mērķis ir glabāt noformējuma tipus.
WebsiteDomains	Tabulas mērķis ir glabāt pievienotos domēna vārdus.
Websites	Tabulas mērķis ir glabāt mājaslapas sarakstu.
WebsiteStyles	Tabulas mērķis ir glabāt personalizētos mājaslapas noformējumus.

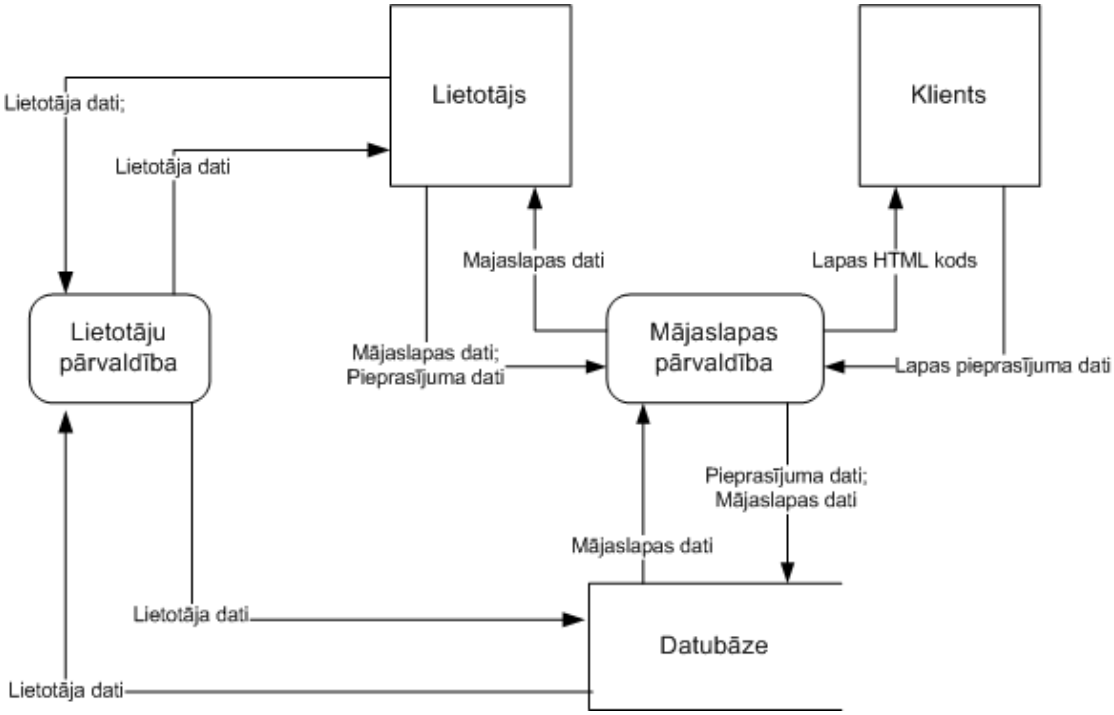
3.4. Starpprocesu atkarības

3.4.1. Datu plūsmas diagrammas

3.4.1.1. 0. līmeņa datu plūsmu diagramma - sistēmas kopskats

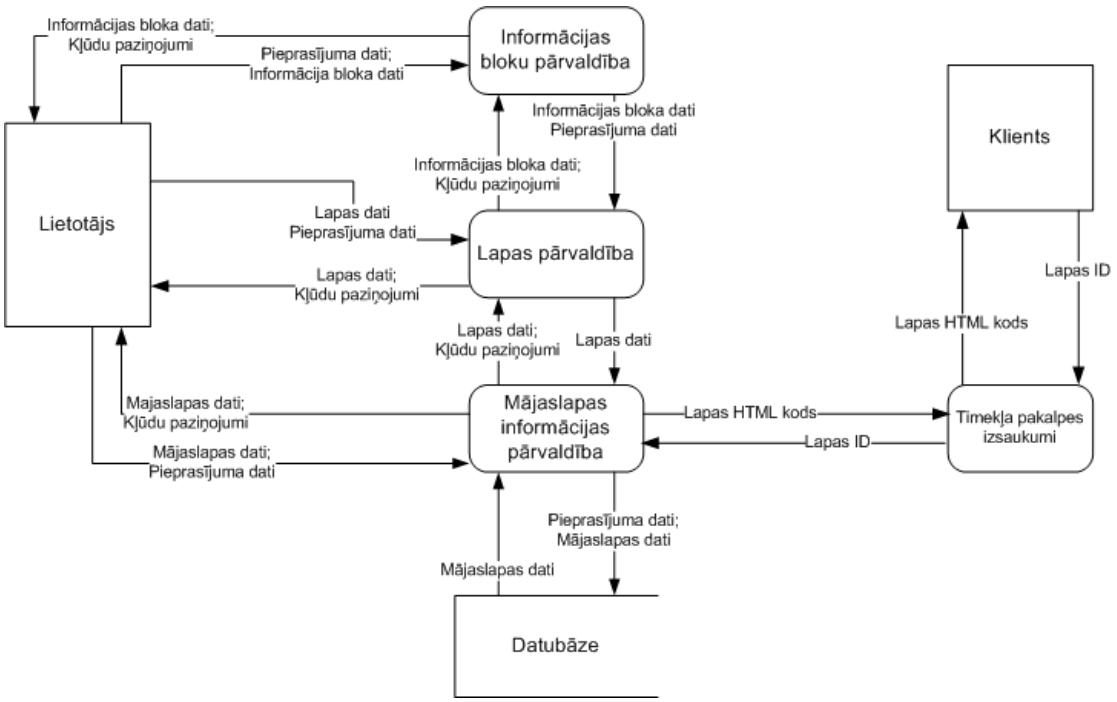
3.1 att. **0. līmeņa datu plūsmu diagramma - sistēmas kopskats**

3.4.1.2. Pirmā līmeņa datu plūsmu diagramma

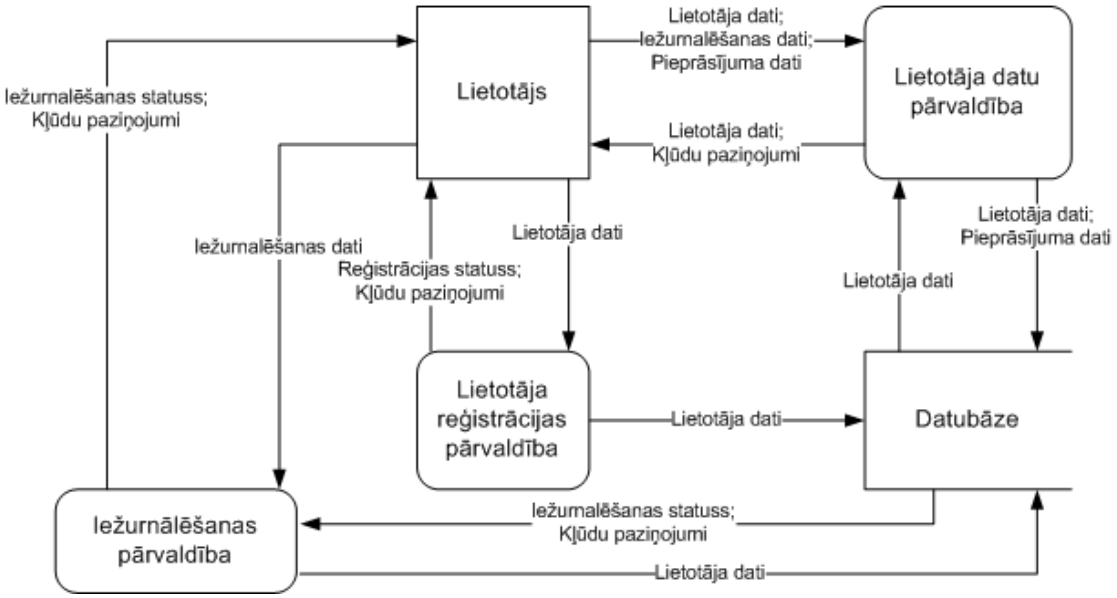


3.2 att. **Pirmā līmeņa datu plūsmu diagramma**

3.4.1.3. Otrā līmeņa datu plūsmu diagrammas



3.3 att. **Mājaslapas pārvaldības datu plūsmu diagramma**



3.4 att. **Lietotāju pārvaldības datu plūsmas diagramma**

3.5. Datu detalizēts projektējums

3.5.1. Tabula Blocks

Tabulas mērķis ir glabāt rediģējamo bloku sarakstu.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls

BlockID	tinyint	PK	Nē	Jā
Name	nvarchar (50)		Nē	Nē

3.5.2. Tabula ContentTypes

Tabulas mērķis ir glabāt satura tipus.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
ContentTypeID	tinyint	PK	Nē	Jā
Name	nvarchar (20)		Nē	Nē
IsEditable	bit		Nē	Nē

3.5.3. HeaderImgs

Tabulas mērķis ir glabāt visas pieejamas augšienes.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
HeaderImgID	smallint	PK	Nē	Jā
Name	char (1)		Jā	Nē

3.5.4. HeaderSelectedImgs

Tabulas mērķis ir glabāt saites starp augšieni un noformējuma kategoriju.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
HeaderSelectedImgID	smallint	PK	Nē	Jā
HeaderImgID	smallint	FK	Nē	Nē
TemplateCategoryID	tinyint	FK	Nē	Nē

3.5.5. ItemUploads

Tabulas mērķis ir glabāt augšupielādēto failu sarakstu.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
ItemUploadID	int	PK	Nē	Jā
StaticPageContenID	int	FK	Nē	Nē

ItemRandomID	int		Nē	Nē
FileName	varchar (55)		Nē	Nē
IsUploaded	bit		Nē	Nē

3.5.6. MemberProfiles

Tabulas mērķis ir glabāt lietotāja profila informāciju.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
MemberID	int	PK, FK	Nē	Jā
Phone	nvarchar (20)		Jā	Nē
Adress	nvarchar (50)		Jā	Nē
CompanyName	nvarchar (50)		Jā	Nē
InCompanyPosition	nvarchar (50)		Jā	Nē

3.5.7. Members

Tabulas mērķis ir glabāt lietotāja pamatdatus.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
MemberID	int	PK	Nē	Jā
Email	nvarchar (40)		Nē	Jā
Password	char (32)		Nē	Nē
Name	nvarchar (30)		Nē	Nē
IP	nvarchar (15)		Jā	Nē

3.5.8. MemberSites

Tabulas mērķis ir glabāt saiti starp lietotāju un mājaslapu, kuru viņš administrē.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
MemberSiteID	int	PK	Nē	Jā
MemberID	int	FK	Nē	Nē
WebsiteID	int	FK	Nē	Nē

Level	smallint		Nē	Nē

3.5.9. SiteContents

Tabulas mērķis ir glabāt satura bloka informāciju.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
SiteContentID	int	PK	Nē	Jā
WebsiteID	int	FK	Nē	Nē
ContentTypeID	tinyint	FK	Nē	Nē
Name	nvarchar (100)		Nē	Nē
Text	Nvarchar(MAX)		Jā	Nē
IP	varchar (15)		Jā	Nē
Date	smalldatetime		Nē	Nē

3.5.10. StaticPageContents

Tabulas mērķis ir glabāt saiti starp satura bloku un lapu.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
StaticPageContentID	int	PK	Nē	Jā
StaticPageID	int	FK	Nē	Nē
BlockID	tinyint	FK	Nē	Nē
SiteContentID	int	FK	Nē	Nē
Position	tinyint		Nē	Nē

3.5.11. StaticPages

Tabulas mērķis ir glabāt lapu sarakstu.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
StaticPageID	int	PK	Nē	Jā
WebsiteID	int	FK	Nē	Nē
ParentID	int	FK	Nē	Nē

Name	nvarchar (100)		Nē	Nē
Description	nvarchar (250)		Jā	Nē
IsEnabled	bit		Nē	Nē
Position	tinyint		Nē	Nē

3.5.12. TemplateBlocks

Tabulas mērķis ir glabāt noformējuma bloku sarakstu.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
TemplateBlockID	smallint	PK	Nē	Jā
TemplateTypeID	tinyint	FK	Nē	Nē
BlockID	tinyint	FK	Nē	Nē

3.5.13. TemplateCategories

Tabulas mērķis ir glabāt mājaslapas noformējuma kategorijas sarakstu.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
TemplateCategoryID	tinyint	PK	Nē	Jā
Name	nvarchar (40)		Nē	Nē

3.5.14. TemplateCategorySelected

Tabulas mērķis ir glabāt saites saistību starp noformējumu un kategoriju.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
TemplateCategorySelectedID	smallint	PK	Nē	Jā
TemplateID	smallint	FK	Nē	Nē
TemplateCategoryID	tinyint	FK	Nē	Nē

3.5.15. Templates

Tabulas mērķis ir glabāt noformējuma sarakstu.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls

TemplateID	smallint	PK	Nē	Jā
TemplateTypeID	tinyint	FK	Nē	Nē
IsPublic	bit		Nē	Nē

3.5.16. TemplateTypes

Tabulas mērķis ir glabāt noformējuma tipus.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
TemplateTypeID	tinyint	PK	Nē	Jā
Name	nvarchar (40)		Nē	Nē
TotalBlocks	tinyint		Nē	Nē
TotalCollums	tinyint		Nē	Nē

3.5.17. WebsiteDomains

Tabulas mērķis ir glabāt pievienotos domēna vārdus.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
WebsiteID	int	PK, FK	Nē	Jā
DomainName	varchar (20)		Nē	Jā

3.5.18. Websites

Tabulas mērķis ir glabāt mājaslapas sarakstu.

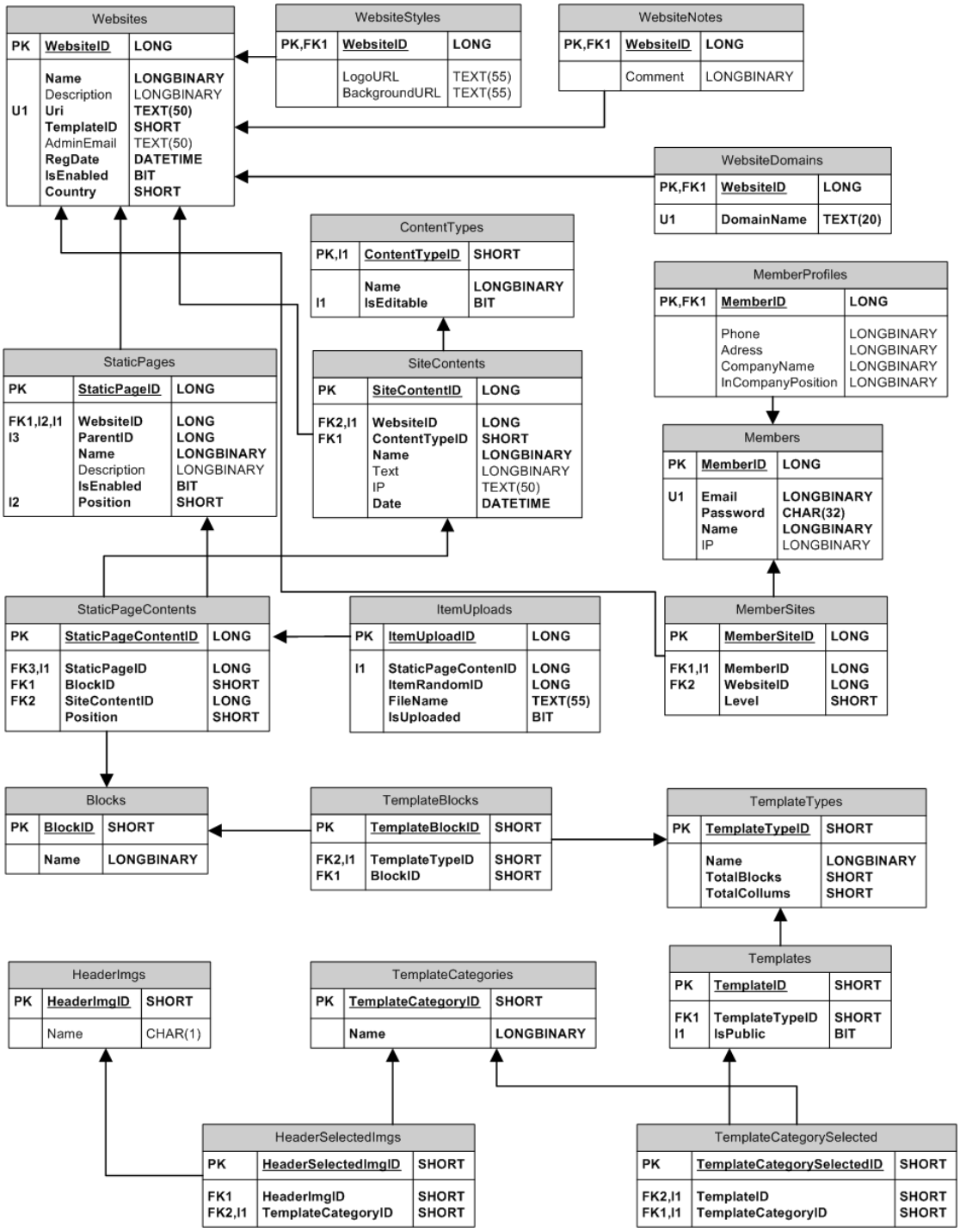
Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
WebsiteID	int	PK	Nē	Jā
Name	nvarchar (50)		Nē	Nē
Description	nvarchar (300)		Jā	Nē
Uri	varchar (50)		Nē	Jā
TemplateID	smallint	FK	Nē	Jā
AdminEmail	varchar (50)		Jā	Nē
RegDate	smalldatetime		Nē	Nē

3.5.19. WebsiteStyles

Tabulas mērķis ir glabāt personalizētos mājaslapas noformējumus.

Lauks	Lauka datu tips	Lauka tips	Var būt tukšs	Ir unikāls
WebsiteID	int	PK, FK	Nē	Jā
LogoURL	varchar (55)		Jā	Nē
BackgroundURL	varchar (55)		Jā	Nē

3.6. Fiziskais ER modelis



3.5 att. **Fiziskais ER modelis.**

3.7. Lietotāju saskarne

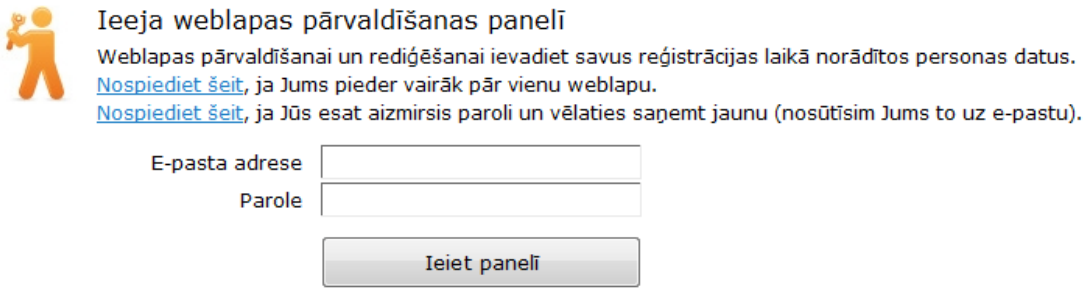
3.7.1. Lietotāja reģistrācija

Veicot reģistrāciju, klientam jāievada pieteikšanas un mājaslapas datus (3.6 att.). Pēc datu ievadīšanas, klientam ir jāpiekrīt lietošanas noteikumiem un jānospiež poga „Reģistrēt weblapu”. Pēc pogas nospiešanas tiek veikta ievadīto datu pārbaude un saglabāšana.

3.6 att. **Lietotāja reģistrācijas ievadlauki**

3.7.2. Lietotāja pieteikšanas

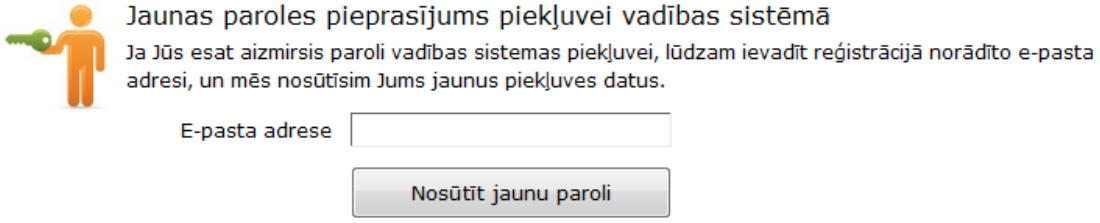
Veicot pieteikšanos, klientam jānorāda sava e-pasta adrese un paroli (3.7 att.).. Pēc datu ievadīšanas ir jānospiež poga „Ieiet panelī”. Nospiežot pogu tiek veikta ievadīto datu pārbaude un lietotāja pieteikšanas



3.7 att. **Lietotāja pieteikšanas ievadlauki**

3.7.3. Aizmirsta paroles maiņa

Pieprasot jauno paroli, lietotājam ir jāievada savu e-pasta adresi un jānospiež poga „Nosūtīt jaunu paroli” (3.8 att.). Nospiežot pogu tiek veikta ievadīto datu pārbaude un izvadīts paziņojums, ka ir nosūtīts e-pasts ar jauniem datiem.



3.8 att. **Aizmirsta paroles maiņas ievadlauki**

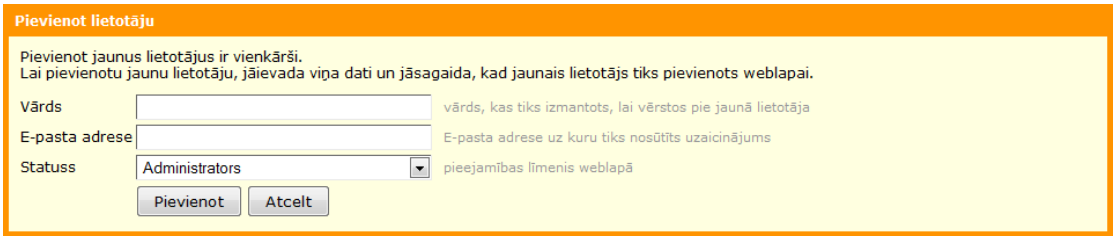
3.7.4. Profila rediģēšana

Lietotājs var izmainīt savu vārdu, paroli, uzņēmuma nosaukumu, norādīt savu amatu, adresi un tālruni. (3.9 att.). Nospiežot pogu „Saglabāt” tiek veikta ievadīto datu pārbaude un izvadīts paziņojums, ka dati ir atjaunoti un nospiežot „Atcelt” darbība it atcelta.

3.9 att. **Profila rediģēšanas ievadlauki**

3.7.5. Administratora pieaicināšana

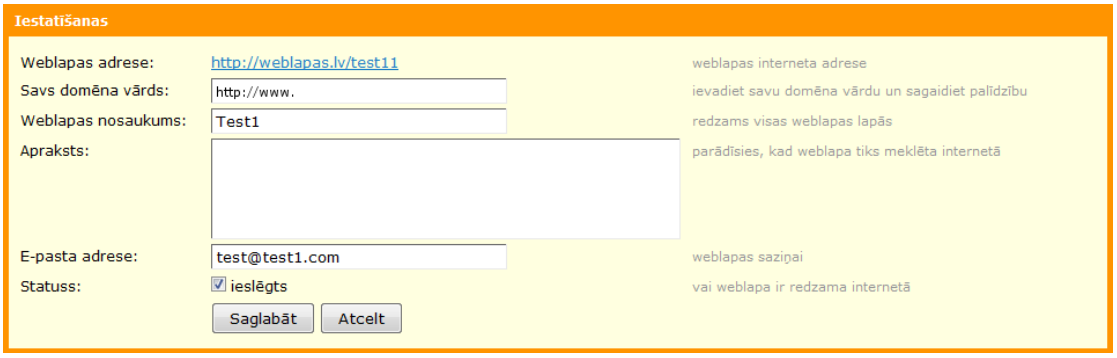
Lietotājs var pieaicināt jaunu administratoru norādot viņa vārdu, e-pasta adresi. Nospiežot pogu „Pievienot” ir pārbaudīti ievadītie dati un jaunam administratoram ir nosūtīts aicinājums pievienoties lapas administrēšanai. Nospiežot pogu „Atcelt” darbība ir atcelta.



3.10 att. **Administratora pieaicināšanas ievadlauki**

3.7.6. Mājaslapas datu rediģēšana

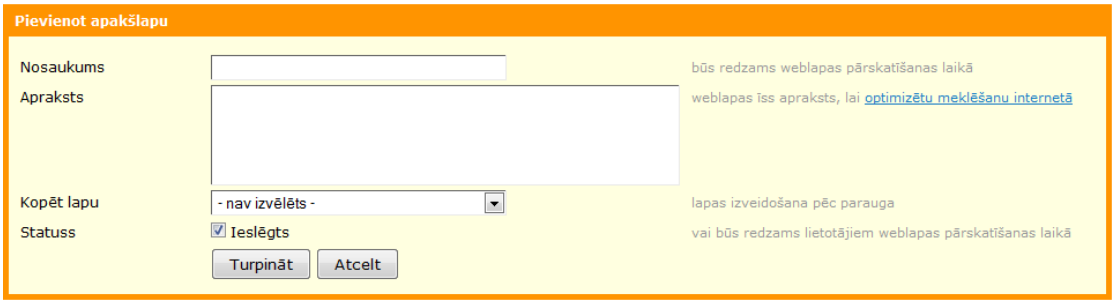
Mājaslapas rediģēšanas lapā var izmainīt mājaslapas domēna vārdu, nosaukumu, aprakstu, e-pasta adresi un statusu (3.11 att.). kā arī ir pieejamas divas pogas „Saglabāt” un „Atcelt”. Nospiežot pogu „Saglabāt” tiek veikta ievadīto datu pārbaude un lietotājam izvadīts paziņojums, ka dati ir atjaunoti, un nospiežot pogu „Atcelt” darbība ir atcelta.



3.11 att. **Mājaslapas datu rediģēšanas ievadlauki**

3.7.7. Jaunas lapas pievienošana

Pievienojot jaunu lapu ir jāievada lapas nosaukumu, aprakstu, izvēlēties bāzes lapu un statusu. Nospiežot pogu „Turpināt” tiek veikta ievadīto datu pārbaude un paradīta jauna lapa, un nospiežot pogu „Atcelt” lapa nav izveidota.



3.12 att. **Lapas pievienošanas ievadlauki**

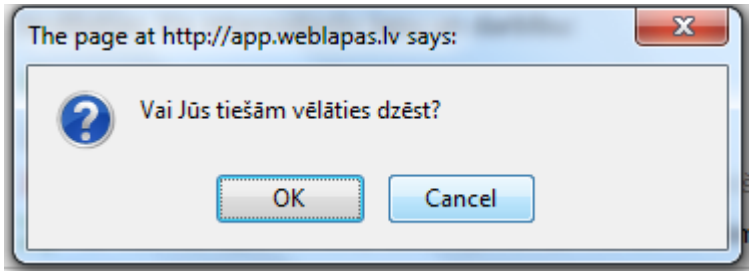
3.7.8. Lapas saraksta attēlošana

3.13 attēla parādīti visas mājaslapas lapas (3.13 att.). Katra lapa attēlota saites veidā un 3 ikonām. Saite norāda uz lapas rediģēšanas lapu, pirmā ikona norāda uz apakšlapas pievienošanu, otra ikona norāda uz lapas rediģēšanas lapu un trešā ikona dzēš lapu.

3.13 att. **Lapas saraksta attēlošana**

3.7.9. Lapas dzēšana

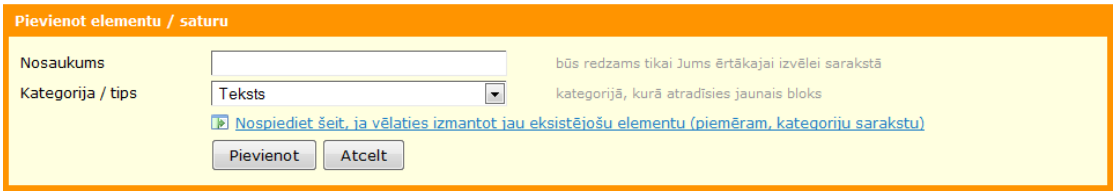
Pēc dzēšanas pogas nospiešanas (3.13 att.) ir parādīts lapas dzēšanas apstiprinājuma logs ar divām pogām (3.14 att.). Nospiežot pogu „Ok" lapa ir dzēsta un nospiežot pogu „Cancel" darbība ir atcelta.



3.14 att. **Lapas dzēšanas apstiprinājums**

3.7.10. Informācijas bloka pievienošana

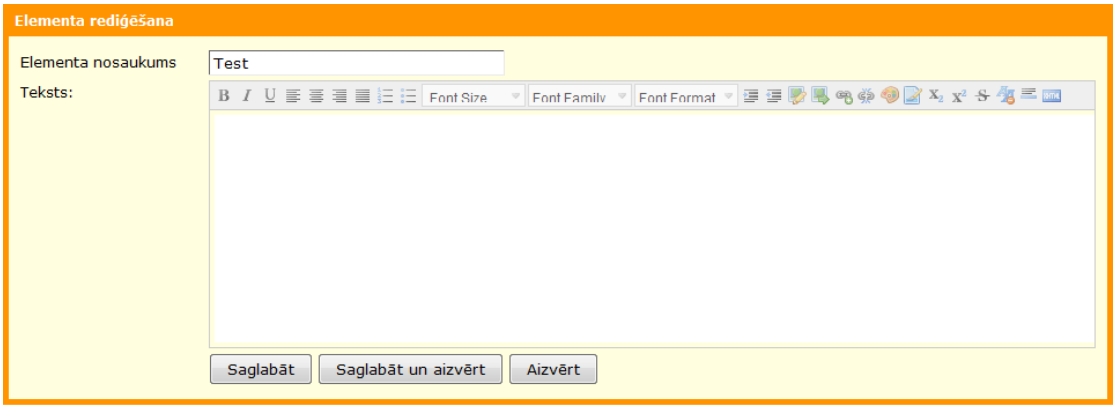
Pievienojot informācijas bloku jāievada nosaukumu un jāizvēlas informācijas bloka tipu (3.15 att.). Nospiežot pogu „Pievienot" tiek veikta ievadīto datu pārbaude un izveidots jauns informācijas bloks, un nospiežot pogu „Atcelt" darbība ir atcelta.



3.15 att. **Informācijas bloka pievienošanas ievadlauki**

3.7.11. Informācijas bloka saturas maiņa

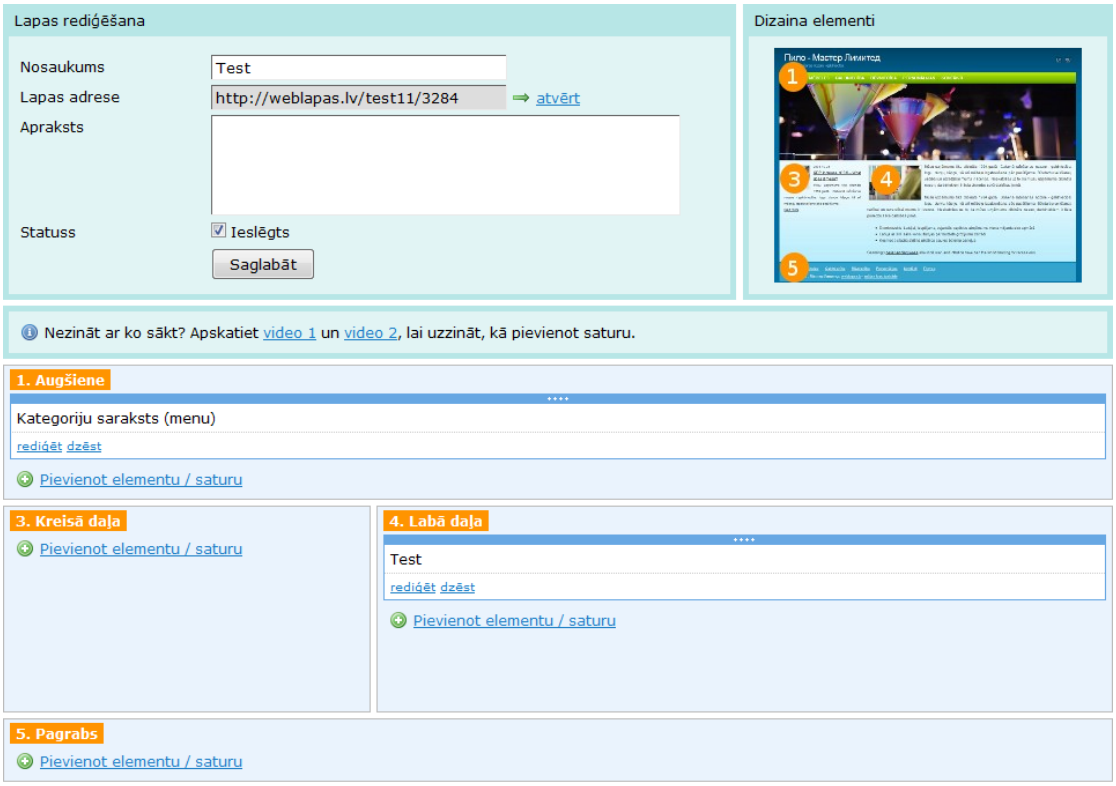
Informācijas bloka rediģēšanas lapā var izmainīt informācijas bloka nosaukumu un saturu (3.16 att.). Nospiežot pogu „Saglabāt” tiek veikta ievadīto datu pārbaude un izvadīts paziņojums, ka dati ir atjaunoti, nospiežot pogu „Saglabāt un aizvērt” tiek veikta ievadīto datu pārbaude un logs ir aizvērts, un nospiežot pogu „Atcelt” darbība ir atcelta.



Att. 3.16 **Informācijas bloka saturas maiņas ievadlauki**

3.7.12. Lapas informācijas apskatīšana

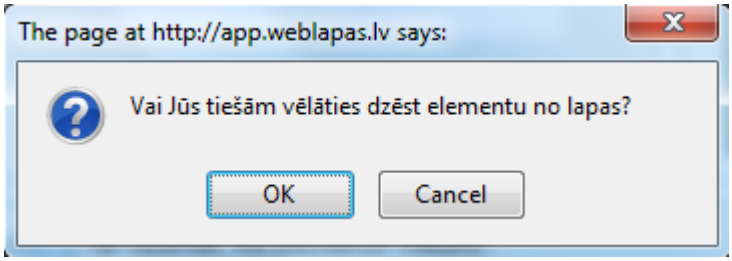
Lapas informācijas apskatīšanas lapā ir pieejama lapas rediģēšanas forma (3.7.7 att.) un informāciju bloku saraksts ar divām hipersaitēm „rediģēt” un „dzēst”. Nospiežot pogu „rediģēt” parādās informācijas bloka rediģēšanas lapa (3.7.11 att.) un nospiežot pēc pogu „dzēst” paradās lapas informācijas bloka dzēšanas apstiprinājums. (sk. 3.7.13).



Att. 3.17 **Lapas informācijas apskatīšanas attēlojums**

3.7.13. Lapas informācijas bloka dzēšana

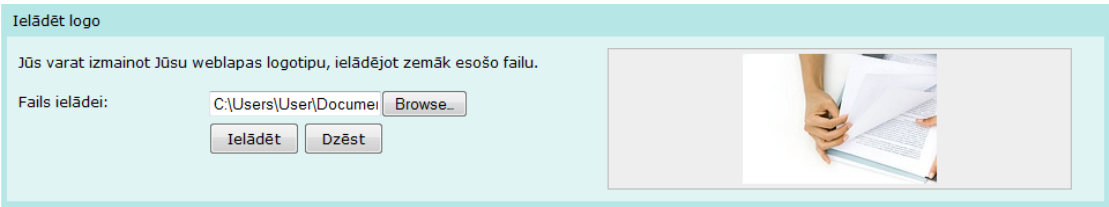
Pēc dzēšanas pogas nospiešanas (3.13 att.) parādās lapas informācijas bloka dzēšanas apstiprinājuma logs ar divām pogām (3.13 att.). Nospiežot pogu „Ok” informācijas bloks ir dzēsts un nospiežot pogu „Cancel” darbība ir atcelta.



Att. 3.18 **Lapas informācijas bloka dzēšanas apstiprinājums**

3.7.14. Mājaslapas logotipa augšupielāde

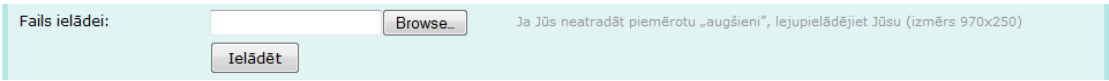
Augšupielādējot logotipu ir jānospiež poga „Browse” un ir jāizvēlas fails no cietā diskā (3.19 att.). Pēc pogas „Ielādēt” nospiešanas ir veikta failā augšupielāde, pārbaude un datu atjaunošana. Pēc veiksmīgas logotipa atjaunošanas tas tiek parādīts lietotājam. Pēc pogas „Dzēst” nospiešanas ir parādās logotipa dzēšanas apstiprinājums (3.7.18 att.).



Att. 3.19 **Mājaslapas logotipa augšupielādes ievadlauki**

3.7.15. Lielas bildes augšupielāde

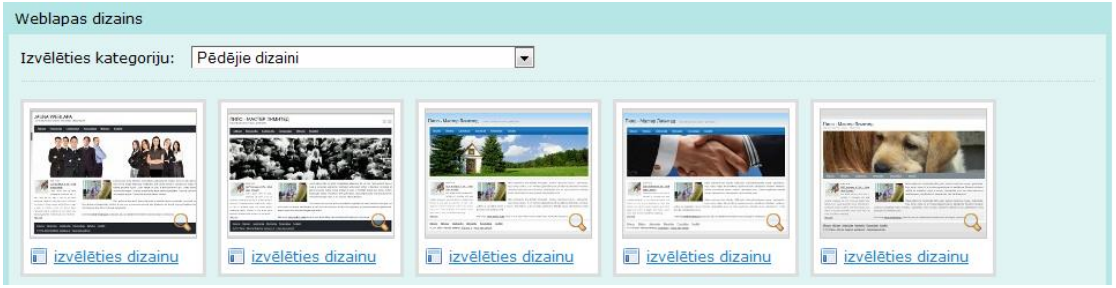
Aaugšupielādējot bildi ir jānospiež poga „Browse” un ir jāizvēlas fails no cietā diskā (3.20 att.). Pēc pogas „Ielādēt” nospiešanas ir veikta failā augšupielāde, pārbaude un datu atjaunošana.



Att. 3.20 **Lielas bildes augšupielādes ievadlauki**

3.7.16. Noformējuma saraksta izvade

Noformējuma saraksts ir izveidots atkarībā no izvēlētās kategorijas (3.21 att.). Lietotājs var izvēlēties citu kategoriju un saraksts tiks automātiski atjaunots. Saraksta elementam ir pieejama maza noformējuma bilde, noformējuma tipa ikona, kā arī saite noformējuma izvēlei.



Att. 3.21 **Noformējuma saraksta attēlojums**

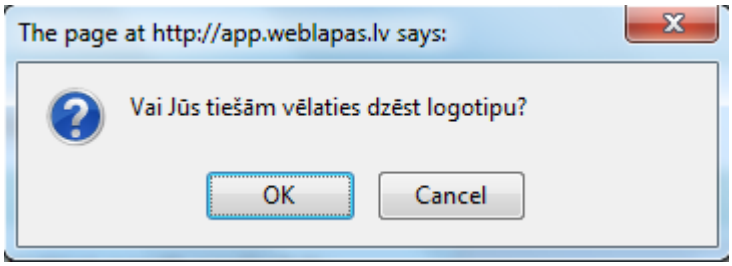
3.7.17.Lielo bilžu saraksta izvade

Noformējuma saraksts ir izveidots atkarībā no izvēlētas kategorijas (3.21 att.). Lietotājs var izvēlēties citu kategoriju un saraksts tiks automātiski atjaunots. Saraksta elementam ir pieejama maza noformējuma bilde, noformējuma tipa ikona, kā arī hipersaites noformējuma izvēlei.

Att. 3.22 **Lielo bilžu saraksta attēlojums**

3.7.18.Logotipa dzēšana

Pēc dzēšanas pogas nospiešanas (3.19 att.) ir parādās logotipa dzēšanas apstiprinājuma logs ar divām pogām (3.23 att.). Nospiežot pogu „Ok” logotips ir dzēsts un nospiežot pogu „Cancel” darbība ir atcelta.



Att. 3.23 **Logotipa dzēšanas apstiprinājums**

4. TESTĒŠANAS DOKUMENTĀCIJA

4.1. Testēšanas procesa apraksts

Visi testi tika veikti paralēli ar Javascript pārbaužu veidošanu, tādēļ pamanītas kļūdas uzreiz tika labotas.

Visi formas elementi ir automātiski konvertēti atbilstoši datubāzes laukuma tipam.

Visi formas elementi kur it atļauta brīva teksta ievade ir aizsargātas no HTML koda injekcijām.

Visi formas elementi tika konvertēti atbilstoši datu bāzes laukuma tipam.

4.2. Vienībtestēšanas rezultāti

4.2.1. Jauna lietotāja reģistrācija

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Obligāta datu lauku pārbaude.	Visi formas datu ievades lauki tiek atstāti tukši.	„Sekojošie lauciņi ir jāaizpilda obligāti: „vārds”, „e-pasts”, „parole”, „parole atkārtoti”, „mājas lapas adrese” un „mājas lapas nosaukums”.”	+
Obligāto datu lauku minimālā garuma ierobežojuma pārbaude.	Ievades formas datu laukos „vārds”, „e-pasts”, „parole”, „parole atkārtoti”, „mājas lapas adrese” un „mājas lapas nosaukums”.	Sekojošie lauciņi jāaizpilda obligāti: „vārds”, „e-pasts”, „parole”, „parole atkārtoti”, „mājas lapas adrese” un „mājas lapas nosaukums”.	+
E-pasta adreses pareizības pārbaude.	E-pasta adreses laukā tiek ievadīta simbolu virkne, kas: satur @, bet neatbilst e-pasta adreses regulārajai izteiksmei, satur punktu un domēna paplašinājumu beigās, bet neatbilst e-pasta adreses regulārajai izteiksmei, satur simbolus, kas neatbilst e-pasta adreses regulārajai izteiksmei.	„Ievadiet pareizu e-pasta adresi (piemērs: email@mail.com).”	+
Mājas lapas adreses	Mājaslapas adreses	„Diemžēl tīmekļa lapa	+

unikalitātes pārbaude.	laukā tiek ievadīta adrese, kas jau ir reģistrēta sistēmā.	ar tādu adresi jau reģistrēta. Mēģiniet reģistrēt tīmekļa lapu ar citu nosaukumu.”	
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Reģistrācija norisinās veiksmīgi – uz lietotāja norādīto e-pasta adresi tiek nosūtīts apstiprinājuma e- pasts.	+
Ar pakalpojuma lietošanas noteikumiem iepazīšanas.	Nav atzīmēts nepieciešamais lauks.	„Izlasiet un piekrītiet pakalpojuma lietošanas noteikumiem.”	+

4.2.2. Pieteikšanas

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Obligāta datu lauku pārbaude.	Visi formas datu ievades lauki tiek atstātai tukši.	„Lai ieietu, jāaizpilda laukums "E-pasts" un "Parole".”	+
E-pasta adreses pareizības pārbaude.	E-pasta adreses laukā tiek ievadīta simbolu virkne, kas: satur @, bet neatbilst e-pasta adreses regulārajai izteiksmei, satur punktu un domēna paplašinājumu beigās, bet neatbilst e-pasta adreses regulārajai izteiksmei, satur simbolus, kas neatbilst e-pasta adreses regulārajai izteiksmei.	„Ievadiet pareizu e-pasta adresi (piemērs: email@mail.com).”	+
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Klients tiek piesaistīts lietotāja sesijai un	+

		pāradresēts un administrācijas daļā.	
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi, bet to vērtība neatbilst datubāzē norādītiem.	„Lietotājs ar šādu e-pasta adresi un paroli neeksistē. Lūdzu, pārbaudiet ievadītos datus.	+

4.2.3. Profila rediģēšana

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Obligāta datu lauku pārbaude.	Obligātie formas datu ievades lauki tiek atstātai tukši.	„Sekojošie lauciņi i r jāaizpilda obligāti: „vārds”, „parole”, „parole atkārtoti”.”	+
Obligāto datu lauku minimālā garuma ierobežojuma pārbaude.	Ievades formas datu laukos „vārds”, „parole”, „parole atkārtoti”..	„Sekojošie lauciņi ir jāaizpilda obligāti: „vārds”, „parole”, „parole atkārtoti”.”	+
Jaunas paroles uzstādīšana.	Laukums „parole” ir aizpildīts, bet laukums „parole atkārtoti” ir atstāts tukšs.	„Jūs neesat atkārtoti ievadījis paroli.”	+
Jaunas paroles pārbaude.	Laukums „parole” nesakrīt ar laukumu „parole atkārtoti”.	„Ievadītās paroles nesakrīt.”	+
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Lietotāja profila dati ir atjaunoti.	+

4.2.4. Aizmirstas paroles maiņa

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
E-pasta adreses pareizības pārbaude.	E-pasta adreses laukā tiek ievadīta simbolu	„Ievadiet pareizu e-pasta adresi (piemērs:	+

	virknes, kas: satur @, bet neatbilst e-pasta adreses regulārajai izteiksmei, satur punktu un domēna paplašinājumu beigās, bet neatbilst e-pasta adreses regulārajai izteiksmei, satur simbolus, kas neatbilst e-pasta adreses regulārajai izteiksmei.	email@mail.com).”	
E-pasta adreses pārbaudes sistēma.	E-pasta adreses laukā tiek ievadīta adrese, kas nav reģistrēta sistēmā.	„Jūs ievadījāt nepareizu e-pasta adresi. Lietotājs ar šādu e-pasta adresi nepastāv.”	+
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Uz lietotāja norādīto e-pasta adresi tiek nosūtīts e- pasts ar jauniem pieteikšanas datiem.	+

4.2.5. Administratora pieaicināšana

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
E-pasta adreses pareizības pārbaude.	E-pasta adreses laukā tiek ievadīta simbolu virkne, kas: satur @, bet neatbilst e-pasta adreses regulārajai izteiksmei, satur punktu un domēna paplašinājumu beigās, bet neatbilst e-pasta adreses regulārajai izteiksmei, satur	„Ievadiet pareizu e-pasta adresi (piemērs: email@mail.com).”	+

	simbolus, kas neatbilst e-pasta adreses regulārajai izteiksmei.		
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Reģistrācija norisinās veiksmīgi – uz lietotāja norādīto e-pasta adresi tiek nosūtīts aicinājums.	+

4.2.6. Mājaslapas datu rediģēšana

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Obligāta datu lauku pārbaude.	Noteiktie formas datu ievades lauki tiek atstāti tukši.	„Sekojošie lauciņi jāaizpilda obligāti: „e-pasts.”	+
E-pasta adreses pareizības pārbaude.	E-pasta adreses laukā tiek ievadīta simbolu virkne, kas: satur @, bet neatbilst e-pasta adreses regulārajai izteiksmei, satur punktu un domēna paplašinājumu beigās, bet neatbilst e-pasta adreses regulārajai izteiksmei, satur simbolus, kas neatbilst e-pasta adreses regulārajai izteiksmei.	„Ievadiet pareizu e-pasta adresi (piemērs: email@mail.com).”	+
Domēna vārda pareizības pārbaude.	Domēna vārda laukā tiek ievadīta simbolu virkne s, kas neatbilst domēna vārda regulārajai izteiksmei.	„Nepareizs domēna vārds.”	+
Domēna vārda unikalitātes pārbaude.	Domēna vārda laukā tiek ievadīta adrese, kas jau reģistrēta sistēmā.	„Norādītais domēna vārds tiek izmantots citā projektā.”	+

Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Mājaslapas dati ir saglabāti.	+

4.2.7. Lapas pievienošana

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Obligāta datu lauku pārbaude.	Noteiktie formas datu ievades lauki tiek atstāti tukši.	„Sekojošie lauciņi jāaizpilda obligāti: „e-pasts.”	+
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Ir izveidota jauna lapa.	+

4.2.8. Lapas dzēšana

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Obligāta datu lauku pārbaude.	Visi formas datu ievades lauki tiek atstāti tukši.	„Lapa nav atrasta.”	+
Datu pareizības pārbaude.	Visi ievadlauki tiek aizpildīti pareizi, bet to vērtība neatbilst datubāzē norādītiem.	„Lapa nav atrasta.”	+
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Lapa ir dzēsta no datubāzes.	+

4.2.9. Informācijas bloka pievienošana

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Obligāta datu lauku pārbaude.	Noteiktie formas datu ievades lauki tiek atstāi tukši.	„Lauciņš „Elementa nosaukums” ir obligāti jāaizpilda.”	+
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Ir izveidots jauns informācija bloks.	+

4.2.10.Informācijas bloka saturas maiņa

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Obligāta datu lauku pārbaude.	Noteiktie formas datu ievades lauki tiek atstāti tukši.	Paziņojumi: „Informācijas bloks nav atrasts.”, „Lauciņš „Elementa nosaukums” ir obligāti jāaizpilda.”.	+
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Informācijas bloks ir saglabāts datubāze.	+

4.2.11.Lapas kopēšanas

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Obligāta datu lauku pārbaude.	Noteiktie formas datu ievades lauki tiek atstāti tukši.	„Lapa nav atrasta.”	+
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Ir izveidota lapas kopija.	+

4.2.12.Lapas informācijas atjaunošana

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Obligāta datu lauku pārbaude.	Noteiktie formas datu ievades lauki tiek atstāti tukši.	„Sekojošie lauciņi jāaizpilda obligāti: „e-pasts.”	+
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Ir izveidota jauna lapa.	+

4.2.13.Lapas informācijas bloka dzēšana

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Obligāta datu lauku	Noteiktie formas datu	„Lapa nav atrasta.”	+

pārbaude.	ievades lauki tiek atstāti tukši.		
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Lapa ir dzēsta no datubāzes.	+

4.2.14.Mājaslapas logotipa augšupielāde

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Obligāta datu lauku pārbaude.	Nav izvēlēts fails.	„Neizdevās ielādēt failu. Pārbaudiet faila izmēru (līdz 1 Mb) un veidu (bilde).”	+
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Mājaslapas logotips ir atjaunots un paradīts lietotājam.	+

4.2.15.Lielas bildes augšupielāde

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Obligāta datu lauku pārbaude.	Nav izvēlēts fails.	„Neizdevās ielādēt failu. Pārbaudiet faila izmēru (līdz 1 Mb) un veidu (ilustrācija).”	+
Reakcija uz pareiziem ievaddatiem.	Visi ievadlauki tiek aizpildīti pareizi.	Mājaslapas liela bilde ir atjaunota un paradīta lietotājam.	+

4.2.16.Spriedzes testēšanu

Testa apraksts	Ievaddati	Sagaidāmais rezultāts	Statuss
Lapas izveidošanas laika testēšana.	Datubāzes ieraksti: 200 reģistrētie lietotāju ieraksti, 200 reģistrētie	Lapas izveidošanas laiks mazāk par 2 sekundēm.	+

	mājaslapu ieraksti, 5000 izveidotie mājaslapu lapu ieraksti, 10000 informācija bloku ieraksti.		

5. PROGRAMMATŪRAS PROJEKTA ORGANIZĀCIJA

Pirms sistēmas izstrādes bija izveidota un saskaņota neliela programmatūras prasību specifikācija, kas norādīja, kādiem ir jābūt ievaddatiem un izvaddatiem, to īpašības un datu drošības prasības. Projekts tika organizēts pēc Agile metodes, kas pieļauj sīkāku dokumentācijas neveidošanu, tā vietā, lai pēc iespējas ātrāk izveidotu darbojošos sistēmu.

Sistēma izstrādāta C# programmēšanas valodā, ASP.NET vidē, kas apstrādā aizpildītās formas. Papildus tiek izmantota SQL Server 2005 datubāze datu glabāšanai, HTML un Javascript, lai aizpildītu un attēlot nepieciešamas lapas un formas. Papildus sistēmā tik izmantota divas dažādas datu attēlošanas sistēmas ASP.NET MVC Template sistēmas un StringTemplate.org Template sistēmas.

Pie sistēmas izstrādes strādāja kvalifikācijas darba autors, kas arī veica datubāzes projektēšanu, programmatūras koda izstrādi un prasību saskaņošanu.

6. KVALITĀTES NODROŠINĀŠANA

Programmatūra tika izstrādāta sazinoties ar pasūtītāju (WG Agency), ik pēc kāda laika tika atrādīts pašreizējais sistēmas stāvoklis. Sistēmas izstrādes laikā tika veikta vienībtestēšana. Pēc sistēmas pilnīgas izstrādes, tika veikta apjomīga sistēmas testēšana, kurā piedalījās arī pasūtītājs. Atrodot kļūdas vai nepilnības, tās tika labotas uzreiz. Pēc pasūtītāja apstiprinājuma sistēma kļuva par produkcijas vidi un pieejama klientiem. Tika izveidota izstrādes vide, izmantojot to pašu programmatūru, kura gandrīz pilnībā atbilst produkcijas videi. Izstrādes vidē tika pārbaudītas atklātās kļūdas un izlabotas. Pēc labošanas un atkārtotas testēšanas izmaiņas tika pārnestas arī produkcijas vidē.

Sistēma tika pārbaudīta uz visiem pašlaik biežāk izmantotajiem interneta pārlūkprogramma un to dažādām versijām.

7. KONFIGURĀCIJU PĀRVALDĪBA

Versiju kontrolei tika izmantots Subversion un Tortoise SVN modulis priekš Visual Studio 2008 Professional. Pastāvīgi tiek veidotas datubāzes kopijas izmantojot SQL Server 2005 iebūvēto. Pašreizējā sistēmas versija ir 1.0. Pēc jaunu funkciju pievienošanas, sistēmai tiks piešķirta versija 1.1. utt.

8. DARBIETILPĪBAS NOVĒRTĒJUMS

Darbietilpības novērtējumam izmantoju Basic COCOMO metodi. Darbietilpību kalendārajos mēnešos var izrēķināt pēc formulām: DPM = A*(KLOC/1000)^B,

DKM = C*(DPM)^ D, kur:

A, B, C, D – konstantes;

KLOC – koda rindiņu skaits tūkstošos;

DPM – darbietilpība personmēnešos;

DKM – darbietilpība kalendārajos mēnešos.

Pēc darba pabeigšanas reālais rindiņu skaits? pēc Visual Studio 2010 aprēķina ir 9300. Tas ir rindiņu skaits neiekļaujot komentārus, kompilatora direktīvas, atverošas un aizverošas iekavas un tukšas rindas. Rindiņas, kas nodrošina sistēmas funkcionalitāti? un ir darba autora rakstīti bija aptuveni 5000 un to skaits tika izvēlēts kā koda rindiņu daudzums.

DPM = 2.4 * (5000/1000)^1.05 = 13.00

DKM = 2.5 * (13.00)^0.38 = 6.62

Reāli sistēmas un dokumentu izstrādē tika patērēti aptuveni 5 mēneši Iespējams, šī neprecizitāte ir skaidrojama ar darba autora zināšanām un pieredzi līdzīgos projektos, kas atviegloja sistēmas izstrādi.

9. PROGRAMMATŪRAS PIRMKODS

Šajā sadaļā ir ievietoti MIPS programmatūras koda fragmenti. Bāzes lapu kontrolleris, zema līmeņa datu struktūra un rekursīvas funkcijas realizācija.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Web;
using System.Web.Mvc;

using Weblapas.Core.Common;
using Weblapas.Core.Enums;
using Weblapas.Core.Models;
using Weblapas.Core.Web;

namespace Weblapas.Core.Controllers.App
{
    /// <summary>
    /// The base controller class for site controllers
    /// </summary>
    [CompressFilter]
    [SetCulture(CultureStore = SetCultureAttribute.CultureLocation.Cookie,
CookieName = "lang")]
    public abstract class BaseSiteController : Controller
    {
        /// <summary>
        /// Required access level
        /// </summary>
        private EAccessLevel accessLevel = EAccessLevel.Any;

        /// <summary>
        /// Current Site Member
        /// </summary>
        private Member SiteMember { get; private set; }

        /// <summary>
        /// Current Site Member Level
        /// </summary>
        private EAccessLevel SiteMemberLevel { get; private set; }

        /// <summary>
        /// tekuwij sait
        /// </summary>
        public Website CurrentSite { get; private set; }

        /// <summary>
        /// This method initializes a user session, so this is the first
place where
        /// we can check if the user is authorized.
        /// </summary>
        /// <param name="requestContext"></param>
        protected override void
Initialize(System.Web.Routing.RequestContext requestContext)
        {
            base.Initialize(requestContext);
            LoadMember();
        }

        /// <summary>
        /// Empty base controller constructor
```

```
        /// </summary>
        public BaseSiteController()
        {
        }

        /// <summary>
        /// Base controller constructor
        /// </summary>
        /// <param name="level">required access level</param>
        public BaseSiteController(EAccessLevel level)
        {
            accessLevel = level;
        }

        /// <summary>
        /// Load current member
        /// </summary>
        private void LoadMember()
        {
            // If user do not matters
            if (accessLevel.Equals(EAccessLevel.Any))
                return;

            // If user do not have authorization details, throw an error
            if ((Session.IsNewSession ||
Validation.IsNull(Session["MemberID"])) &&
Validation.IsNull(Request.Cookies["MemberID"]))
            {
                throw new HttpException(403, "Access denied.");
            }

            int memberID = 0;
            int siteID = 0;

            // Get current user details from user session/cookies
            var result = HttpQuery.GetMemberSessionValues(Request, Session,
ref memberID, ref siteID);

            // If user details are wrong
            if (!result)
                throw new HttpException(500, "Access denied.");

            // Current user site element
            MemberSite memberSite = null;

            // Connect to DataBase wrapper class
            using (var db = new NorthwindDataContext())
            {
                // Check if user is site administrator
                memberSite = QueryLib.GetMemberSiteBySiteAndMemberID(db,
siteID, memberID);

                // If it's ok, save details
                if (!Validation.IsNull(memberSite))
                {
                    CurrentSite = memberSite.Website;
                    SiteMember = memberSite.Member;
                    SiteMemberLevel = (EAccessLevel)memberSite.Level;
                }
                else throw new HttpException(500, "Access denied.");
            }
```

```
            // Compare current user details and site informations with
security string in session
            if (!HttpQuery.IsLogged(SiteMember, siteID,
(EAccessLevel)memberSite.Level, Request, Session))
                throw new HttpException(403, "Access denied.");

            // Setup username in session
            Session["MemberName"] = SiteMember.Name;
        }

        /// <summary>
        /// Get numeric "ObjectID" value from QueryString
        /// </summary>
        /// <returns></returns>
        protected int GetObjectID()
        {
            return GetObjectID("objectID");
        }

        /// <summary>
        /// Get numeric value from QueryString
        /// </summary>
        /// <param name="name">object name</param>
        /// <returns></returns>
        protected int GetObjectID(string name)
        {
            var value = Request.QueryString[name];
            return Validation.IsNumeric(value) ? int.Parse(value) : 0;
        }

        /// <summary>
        /// Generate valid text box
        /// </summary>
        /// <param name="s"></param>
        /// <returns></returns>
        protected ActionResult ContentValid(string title, string s)
        {
            return ShowDiv(title, s, EMessageResultType.Valid);
        }

        /// <summary>
        /// Generate valid content box
        /// </summary>
        /// <param name="s"></param>
        /// <returns></returns>
        protected ActionResult ContentValid(string s)
        {
            return ContentValid(String.Empty, s);
        }

        /// <summary>
        /// Generate error text box
        /// </summary>
        /// <param name="title"></param>

        /// <param name="s"></param>
        /// <returns></returns>
        protected ActionResult ContentInValid(string title, string s)
        {
            return ShowDiv(title, s, EMessageResultType.InValid);
        }

        /// <summary>
```

```
        /// Generate error list box
        /// </summary>
        /// <param name="title"></param>
        /// <param name="s"></param>
        /// <returns></returns>
        protected ActionResult ContentInValid(string title, List<string>
list)
        {
            StringBuilder s = new StringBuilder("<ul>");
            list.ForEach(e => s.AppendFormat("<li>{0}</li>", e));
            s.Append("</ul>");

            return ContentInValid(title, s.ToString());
        }

        /// <summary>
        /// Generate HTML div with response
        /// </summary>
        /// <param name="title"></param>
        /// <param name="content"></param>
        /// <param name="type"></param>
        /// <returns></returns>
        private ActionResult ShowDiv(string title, string content,
EMessageResultType type)
        {
            return Content(HTMLSnippets.ShowResultDiv(title, content,
EMessageResultType.InValid));
        }

        /// <summary>
        /// Checks if Form element is empty
        /// </summary>
        /// <param name="name">element name</param>
        /// <returns></returns>
        protected bool IsEmptyFormField(string name)
        {
            return Validation.IsEmpty(Request.Form[name]);
        }
    }
}

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Web.Mvc;

using Weblapas.Core.Extensions;
using Weblapas.Core.Items;

namespace Weblapas.Core.Common
{
    /// <summary>
    /// Lightweight category tree element
    /// </summary>
    public struct CategoryTreeItem
    {
        public int CategoryID { get; set; }
        public string Name { get; set; }
    }

    /// <summary>
    /// Category tree generator
```

```
    /// </summary>
    public class CategoryTree
    {
        /// <summary>
        /// Result category list
        /// </summary>
        private List<CategoryTreeItem> resultList = null;

        /// <summary>
        /// Source category list
        /// </summary>
        private List<CategoryItem> inputList = null;

        /// <summary>
        /// Children padding element
        /// </summary>
        private string paddingElement = "· ";

        /// <summary>
        /// Do we need to make padding
        /// </summary>
        private bool makePadding = false;

        /// <summary>
        /// Constructor with category elements and settings
        /// </summary>
        /// <param name="list">source list</param>
        /// <param name="padding">padding list</param>
        public CategoryTree(List<CategoryItem> list, bool padding
        {
            inputList   = list;
            makePadding = padding;
            resultList  = new List<CategoryTreeItem>(inputList.Count);
        }

        /// <summary>
        /// Create category tree
        /// </summary>
        /// <param name="selectedValue">selected value</param>
        /// <returns>ASP.NET MVC Select List</returns>
        public SelectList CreateTree(int selectedValue = 0)
        {
            Tree(0, 0);

            return new SelectList(resultList, "CategoryID", "Name",
selectedValue);
        }

        /// <summary>
        /// Generate category tree (based on parent id)
        /// </summary>
        /// <param name="parentID">parent category id</param>
        /// <param name="level">current level</param>
        private void Tree(int parentID, int level)
        {
            if (level > 4) // Max recursive level
                return;

            // Get all child categories
            var items = inputList.Where(e =>
e.ParentCategoryID.Equals(parentID));

            // Process all child categories if available
```

```
            foreach (var item in items)
            {
                // Skip category if it is already processed
                if (resultList.Contains(f =>
f.CategoryID.Equals(item.CategoryID)))
                    continue;

                // Create new element with standard name and id
                var e = new CategoryTreeItem() { CategoryID =
item.CategoryID, Name = item.Name};

                // Add padding element if required
                if (makePadding)
                    e.Name = paddingElement.Clone(level + 1) + e.Name;

                // Add current category to end list and make
                // recursive call to process child categories
                resultList.Add(e);
                Tree(e.CategoryID, level + 1);
            }
        }
    }
}
```

Kvalifikācijas darbs „***Mājaslapu izveides pašapkalpošanās sistēma***” izstrādāts Latvijas Universitātes Datorikas fakultātē.

Ar savu parakstu apliecinu, ka darbs izstrādāts patstāvīgi, izmantoti tikai tajā norādītie informācijas avoti un iesniegtā darba elektroniskā kopija atbilst izdrukai.

Autors: ***Dmitrijs Čudihins _____ 31.05.2010.***

Rekomendēju darbu aizstāvēšanai
Darba vadītājs: ***Mg.dat. Krišs Rauhvargers _____ 31.05.2010.***

Recenzents: ***Ceriņa-Bērziņa Jana***

Darbs iesniegts 31.05.2010.
Kvalifikācijas darbu pārbaudījumu komisijas sekretārs: *Dainis Dosbergs* _____

Darbs aizstāvēts kvalifikācijas darbu pārbaudījuma komisijas sēdē
___.06.2010. prot. Nr. _____, vērtējums _____ (_____)

Komisijas sekretārs(-e): _____

Vadītāja/recenzenta atsauksme
par LU Datorikas fakultātes
pirmā līmeņa profesionālās augstākās izglītības studiju programmas „Programmēšana un datortīklu administrēšana” specializācijā SE (Programmētājs)

studenta .....
(vārds, uzvārds)
kvalifikācijas darbu

.....

Vai kvalifikācijas darbam ir obligātās sastāvdaļas:	jā	nē
-labi komentēts pretendenta patstāvīgi izstrādāts programmas kods;		
-testēšanas dokumentācija, kas apliecina pretendenta patstāvīgi veiktu vienībtestēšanu;		
-pretendenta patstāvīgi izstrādāts zema līmeņa (datu struktūru un algoritmu) projektējums;		
-pretendenta izstrādātajam programmatūras produktam atbilstoša prasību specifikācija un projektējums (vai to fragmenti), kas drīkst nebūt paša pretendenta izstrādāti;		
-paskaidrojošs teksts, kurā atspoguļota konkrētā programmatūras projekta organizācija, kvalitātes nodrošināšana, konfigurāciju pārvaldība un dots darbietilpības novērtējums saskaņā ar izplatītām metodēm.		
Vai datorprogramma atbilst vismaz 3 personmēnešu darbietilpībai?		
Vai programmatūras produkts ir noformēts saskaņā ar valsts un starptautiskiem standartiem?		
Vai programmprodukta izstrāde notikusi saskaņā ar labo programmēšanas praksi?		
Darba noformējuma kvalitāte		
	(1-10)	
Darba publiskā pasnieguma kvalitāte (norāda tikai tad, ja vadītājs/recenzents personīgi piedalās kvalifikācijas pārbaudījumos)		
	(1-10)	
Darba aizstāvēšanas (atbildes uz jautājumiem) kvalitāte (norāda tikai tad, ja vadītājs/recenzents personīgi piedalās kvalifikācijas pārbaudījumos)		
	(1-10)	
Galīgais vērtējums		
	(1-10)	

Komentāri:

Vadītājs/recenzents_____
(Vārds, Uzvārds, paraksts, datums)