



**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
MULTIMEDIJOS INŽINERIJOS KATEDRA**

Saulius Daraška

**Nuotolinio mokymosi kursų kokybės technologinio
įvertinimo ir pritaikymo neįgaliesiems analizė ir
įrankio kūrimas**

Magistro darbas

Darbo Vadovė
doc. dr. D.Rutkauskienė

Kaunas, 2006



**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
MULTIMEDIJOS INŽINERIJOS KATEDRA**

Saulius Daraška

**Nuotolinio mokymosi kursų kokybės technologinio
įvertinimo ir pritaikymo neįgaliesiems analizė ir
įrankio kūrimas**

Magistro darbas

Kalbos konsultantas

Dėst. J. Jonušas
2006-05-____

Vadovė

Doc. Dr. D. Rutkauskienė
2006-05-____

Recenzentė

Doc. Dr. R. Kulvietiene
2006-05-____

Atliko

IFN-4/2 gr. stud.
Saulius Daraška
2006-05-22

Kaunas, 2006

Turinys

Reziumė.....	4
Summary.....	4
Įvadas.....	5
Darbo tikslas	5
Darbo uždaviniai	6
1. Tiriamojo darbo objektas ir dalykas.....	7
2. Tiriamojo darbo metodai ir technologijos	9
3. Detalus darbo planas (2-4 semestrams).....	10
4. Neįgaliųjų darbo su mokymosi medžiaga būdai ir problemos.....	11
4.1. Hiperteksto dokumentų vartojimo scenarijai	11
4.2. Ekranų skaitytojai (Screen Readers).....	11
5. Medžiagos pateikimas hipertekstu	12
6. HTML nuotolinio mokymosi studijose	16
7. Hipertekstu pateiktos medžiagos pritaikymo neįgaliesiems taisyklių sudarymas	17
8. Analizės rezultatai	26
9. Panašių programų ir sprendimų analizė bei palyginimas	27
9.1. Nagrinėti sprendimai.....	27
9.2. Išvados	29
10. Hipertekstu pateiktos medžiagos analizės įrankio projektavimas.....	30
10.1. Technologijų analizė.....	30
10.2. Formalus taisyklių aprašymas	32
10.3. Technologijų pasirinkimas	37
11. Hipertekstu pateiktos medžiagos analizės įrankio kūrimas	39
11.1. Specifikacija	39
11.2. Architektūra.....	40
11.3. Klasių diagrama.....	42
11.4. Taisyklių aprašymo modelis.....	48
11.5. Formalus įvertinimo pagal taisykles algoritmas.....	48
11.6. Duomenų struktūros.....	50
11.7. Vartotojo vadovas.....	53
12. Analizė ir taisyklių sudarymo principai.....	56
12.1. Taisyklių sudarymo principai ir apribojimai:.....	56
12.2. Taisyklių grupės	56
12.3. Taisyklių sąlygos ir pataisymai	57
12.4. Taisyklių pavyzdžiai	58
12.5. Taisyklių įvedimo įrankis.....	59
Išvados ir rezultatai	65
Santrumpų sąrašas	66
Literatūros sąrašas.....	67
Priedai.....	68

Reziუმэ

Neigalių mokymosi būdai ir priemonės nėra labai plačiai nagrinėjamos, todėl neigalieji labai dažnai susiduria su problemomis naudodami nepritaiktą jiems mokomąją medžiagą. Populiarėjant nuotolinėms studijoms vis daugiau neigalių žmonių renkasi tokią studijų formą, nes tai patogiausias būdas mokytis neišeinant iš namų. Medžiaga tokio tipo studijose dažniausiai pateikiama hipertekstu. Kadangi trūksta techninių priemonių hiperteksto dokumentams pritaikyti neigaliesiems, pastarųjų galimybės dirbti su tokia medžiaga tampa labai mažos ar net neįmanomos. Atsižvelgiant į tai, kad paruošti pedagogus darbui su technine HTML kalba yra pakankamai sudėtinga ir užima daug laiko, tai šiame darbe bandoma pasukti kitu keliu ir pateikti sprendimą, leidžiantį net ir visiškai HTML nežinančiam žmogui pritaikyti šia kalba sukurtus dokumentus žmonėms su negalia. Sudaryta taisyklių aibė ir lankstus įrankis leidžia efektyviai ir kokybiškai pritaikyti hipertekstu pateiktą medžiagą neigaliesiems.

Summary

Education methodology of people with disabilities is not very well developed and it is the reason why disabled people are facing problems in using of unadapted educational materials. More and more disabled people are choosing distance learning due to its convenience and possibility to learn from home. In most cases materials are presented in hypertext format. There are no sufficient tools for adaptation of hypertext documents for disabled people usage and it makes work with those documents very hard or even impossible. Due to the fact, that preparation of teachers and professors for work with hypertext may be very time consuming and difficult, this thesis's where trying to use different approach - find a solution for non HTML literate person to adapt HTML documents for disabled people. Set of rules was created and flexible tool lets effectively adapt hypertext materials for disabled.

Įvadas

Darbo tikslas

Išanalizuoti nuotoliniu būdu besimokančių žmonių su negalia problemas, mokantis iš hipertekstu pateiktos mokymosi medžiagos. Ištirti priemones, naudojamas mokymosi procesui palengvinti.

Išanalizuoti mokymosi medžiagos pateikimą hipertekstu. Nustatyti esamas problemas ir suprojektuoti bei pateikti įrankį, leidžiantį lengvai ir greitai pritaikyti hipertekstu pateiktą mokymosi medžiagą neįgaliesiems. Hiperteksto dokumentų analizei sudaryti formalų taisyklių aprašymo modelį. Pateiktas įrankis turi būti orientuotas į hiperteksto vidinės – techninės struktūros neišmanančius vartotojus.

Padaryti sistemos taisyklių įvedimo galimybę universalia – tinkančią įvesti taisykles, susijusias ne tik su neįgalių žmonių mokymu. Taip sistemą bus galima pritaikyti įvairioms reikmėms.

Darbo uždaviniai

Probleminės srities analizė

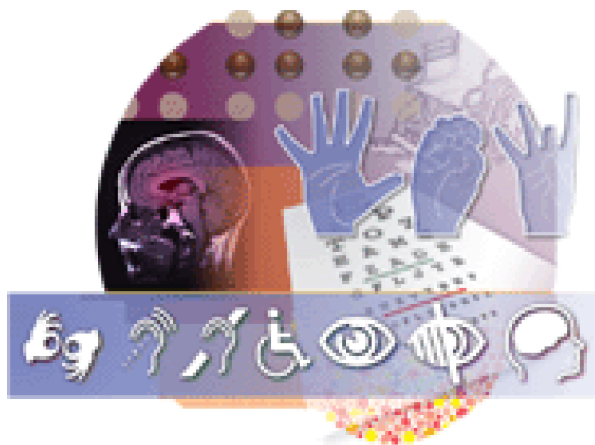
1. Surinkti ir susisteminti žmonių negalias, trukdančias mokytis.
2. Išanalizuoti Hiperteksto HTML kalbos struktūrą, privalumus ir trūkumus.
3. Išanalizuoti galimas problemas pateikiant medžiagą Hipertekstu neįgaliesiems.
4. Sudaryti taisyklių rinkinį, kuris turi atitikti hipertekstu pateiktą medžiagą.

Projektavimas ir programinės įrangos kūrimas

1. Suprojektuoti duomenų struktūras, skirtas formaliai aprašyti taisyklėms.
2. Suprojektuoti programą (įrankį), analizuojantį hipertekstinį dokumentą pagal turimas taisykles (profilius), ir, jei galima, automatiškai atliekantį korekcijas - ištaisanti esamas klaidas.
3. Išanalizuoti galimybę įrankį integruoti į esamas interneto naršyklės.
4. Testuoti sukurtą įrankį.

1. Tiriamojo darbo objektas ir dalykas

Pagrindinis tiriamojo darbo objektas yra mokymosi terpės pateikimas žmonėms su negalia, studijuojantiems nuotoliniu būdu. Kadangi įprastose, sveikiems žmonėms pritaikytose paskaitose, neįgaliesiems sunku dalyvauti, pagrindinis ir tinkamiausias mokymo būdas jiems yra nuotolinis mokymas. Tada žmogus gali mokytis net neišeidamas iš namų. Tačiau nuotolinėse studijose populiariausias medžiagos pateikimas ir bendravimas tarp kurso dalyvių realizuojamas Hipertekstu. Taigi atsiranda papildomų problemų, jei nesilaikoma (kaip dažniausiai ir būna) hipertekstinių dokumentų sudarymo taisyklių. Šios problemos dažniausiai kyla dėl kvalifikacijos trūkumo ir automatinių įrankių naudojimo, kurie ne visada atitinka numatytus standartus. Medžiagą ruošiantys pedagogai dažniausiai neišmano hiperteksto dokumento vidinės struktūros. Populiariausia naudojama hiperteksto metakalba yra HTML. Ja kurti dokumentus, nepasitelkus papildomų priemonių, yra sudėtingas ir dažniausiai neefektyvus darbas, todėl naudojamos pagalbinės priemonės (dažniausiai neturinčios įrankių dokumento pritaikymo neįgaliesiems), kuriomis dokumentas paruošiamas ir tik tuomet išsaugomas kaip hipertekstas. Taip paruoštą dokumentą pritaikyti neįgaliesiems be papildomų techninių hiperteksto kalbos žinių yra sudėtinga. Taigi kyla problema, kaip galima paprasčiau ir efektyviau padėti žmonėms, ruošiantiems medžiagą nuotolinėms studijoms ir žmonėms su negalia. Efektyviausias sprendimas – įrankis, leidžiantis pagal turimą duomenų bazę išanalizuoti dokumentą ir nustatyti esamas klaidas, tuo pačiu jas ir ištaisyti. Kadangi hipertekstas yra dokumento struktūrą aprašanti meta kalba, formali analizė yra pakankama, norint nustatyti, kaip dokumentas atitinka reikalavimus. Dokumento turinys šiuo atveju yra neaktualus.



1 pav.

1 pav. paveikslėlis parodo negalias, trukdančias žmonėms mokytis. Paveiksle pavaizduotos problemos (tiriomojo darbo objektai ir dalykai), kurias reikia spręsti, norint pritaikyti NM kursą neįgaliesiems.

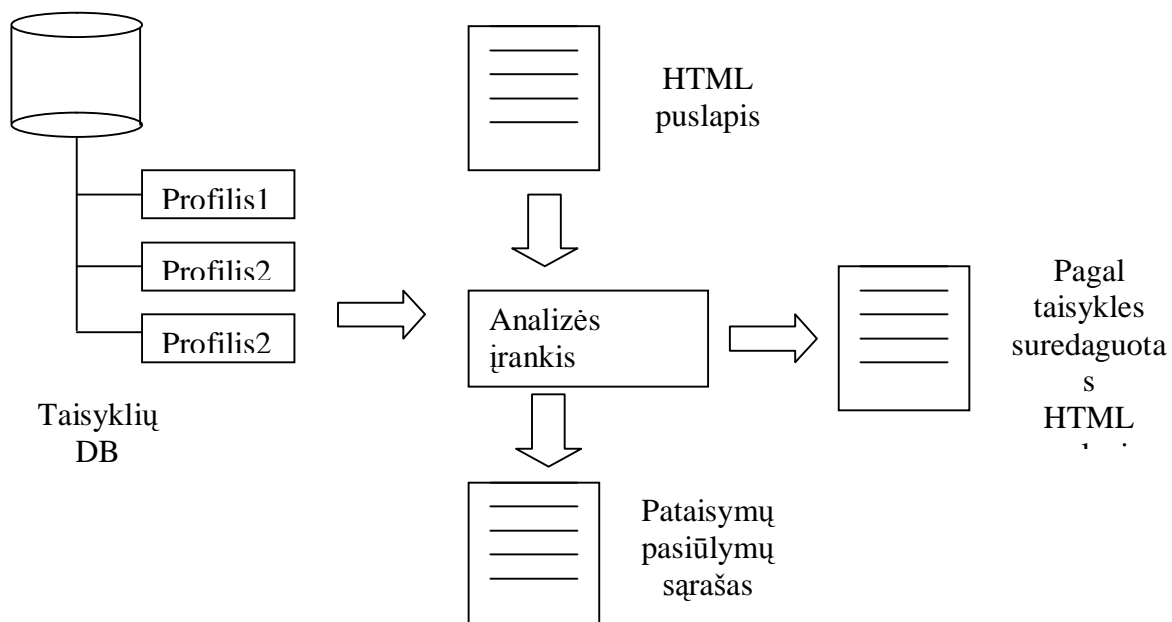
- Gestų kalba kurtiesiems;
- Supaprastintas valdymas ir specialūs įrenginiai;
- Garsai;
- Invalido vežimėliai;
- Regėjimas;
- Aklumas arba silpnas regėjimas;
- Intelektu atsilikimas.

Įrankis turėtų apimti tik dalį problemų, kadangi jis kuriamas Hipertekstu pateiktos mokymosi terpės analizei. Įrankis yra kaip tarpininkas tarp internete hipertekstu sukurtu puslapio ir interneto naršyklės. Jis, pagal turimą taisyklių rinkinį įvertinęs puslapį, pasiūlo arba atlieka tam tikras korekcijas pagal pasirinktą negalią. Juo remiantis sukurtą medžiagą turi būti galima patikrinti, ar ji atitinka norimus standartus. Taisyklių rinkinys turi būti papildomas ir redaguojamas.

2. Tiriomojo darbo metodai ir technologijos

Pagrindiniai darbo metodai:

- Pagrindinių problemų, išskylančių žmonėms su negalia, mokantis iš hipertekstu pateiktos medžiagos, analizė;
- Įrankių analizė ir jų privalumų bei trūkumų analizavimas pagal sudarytų kriterijų sąrašą;
- Mokymo įstaigų sukurtos patirties panaudojimas, kuriant taisykles ir profilius;
- Programinės įrangos projektavimas, atliekamas remiantis UML metakalba;
- Programavimui naudojamos NET technologijos;
- Orientavimasis į kliento-serverio(tarnybinės stoties) technologiją;
- Įvairių naršyklių ir OS sistemų išbandymas;
- PĮ kūrimo procesui naudojamas “krioklio” metodas.



2 pav. Principinė įrankio darbo schema

3. Detalus darbo planas (2-4 semestrams)

Tiriamasis darbas 2

- ü Analizuojamos technologijos, reikalingos darbo įgyvendinimui (HTML, NET client-server, MS SQL Server).
- ü Literatūros apžvalga patvirtintai tiriamojo darbo temai.
- ü Neįgalumų analizė ir profilių bei taisyklių pradinis juodraštis.

Tiriamasis darbas 3

- ü Analizuojami ir lyginami sprendimų variantai bei pagrindžiamas pasirinktas variantas.
- ü Įrankio projektavimas (UML) ir kūrimas (NET, MS SQL Server).
- ü Sukurta veikianti sistema (įrankis).

Magistro baigiamasis darbas

- ü Atliekamas bandomasis NM įvertinimo įrankio testavimas.
- ü Įrankis tobulinamas pagal testavimo rezultatus.
- ü Analizuojami gauti rezultatai, nurodant pavykusius ir nepavykusius aspektus.
- ü Detalaus magistro baigiamojo darbo aprašymas.

4. Neįgaliųjų darbo su mokymosi medžiaga būdai ir problemos

4.1. Hyperteksto dokumentų vartojimo scenarijai

- Silpnai regintis žmogus gali naudotis Web naršyklėmis, kuris skaito Web puslapio turinį garsiai, tačiau ši programinė įranga dažnai „susipainioja“ dėl vaizduojamų žemėlapių, JavaScriptų ir kitų Web dizaino elementų.
- Kurčiam žmogui reikia tekstinės audio informacijos versijos.
- Kai kurie žmonės pastebi skirtingas spalvas, tačiau gali nematyti šviesaus kolorito spalvų tamsiame fone.
- Yra žmonių, šiek tiek gebančių valdyti ranką, bet negalinčių paspausti daugiau nei vieną mygtuką tuo pačiu metu klaviatūroje. Gerai sukurtas Web meniu suteiks alternatyvius naršymo metodus.
- Silpnai regintiems žmonėms būtinas didelis teksto šriftas, dideli valdymo komponentai.
- Daltonizmu sergantys žmonės, atsižvelgiant į ligos pobūdį, nemato arba silpnai skiria žalią, raudoną ir violetinę spalvas.

4.2. Ekranų skaitytojai (Screen Readers)

Ekranų skaitytojai - tai tokios programos, kurios skaito internetinį puslapį vartotojui. Viena iš populiariausių programų yra JAWS for Windows. Kiti populiariausi ekranų skaitytuvai Window-Eyes, HAL ir out-SPOKEN, by GW-Micro, Dolphin ir Alva AG.

Kaip ir kiti ekranų skaitytojai, JAWS paverčia medžiagą kompiuterio vaizduoklyje į dirbtinę kalbą ir skaito ją garsiai, eidamas iš kairės į dešinę ir iš viršaus į apačią. Kai skaitomas Web turinys, ekranų skaitytojai ir kalbančios naršyklės (tokios kaip IBM's Home Page Reader) apie nuorodas (links) vartotojui praneša pasakydamos žodį *link* arba perskaitydamos ją kitokiu balsu. JAWS skiria keletą nuorodų tipų. Kai jis susiduria su teksto nuoroda, pasako tik žodį *link*. "*This page link*" reiškia, kad čia nuoroda į tame pačiame puslapyje esančią dalį. "*link graphic*" identifikuoja grafinio objekto arba mygtuko nuorodą ir "*image map link*" nurodo, kad tai - paveikslėlių žemėlapių (image map) regiono nuoroda.

JAWS skaito ALT tekstą (tai alternatyvus tekstas vizualiniams ir garsiniams puslapio objektams), jei prie objekto ALT teksto nėra, tada skaitoma nuoroda arba rinkmenos pavadinimas (į kur ta nuoroda). Taigi klaidingas arba neegzistuojantis ALT tekstas gali labai apriboti neįgalių žmonių darbą.

5. Medžiagos pateikimas hipertekstu

Informacijos pateikimui internete geriausiai tinka HTML kalba. HTML kalba aprašo dokumento vaizdavimo taisykles, bet neaprašo pačios dokumento loginės prasmės, todėl medžiaga paruošiama atskirai ir tik paskui apipavidalinama HTML kalbos pagalba. HTML kalba aprašytus dokumentų vaizdavimui naudojamos interneto naršyklės. HTML dokumentams perduoti naudojamas HTTP protokolas kuri yra pakankamai paprastas ir nereikalaujantis daug resursų. Praktiškai kiekviena operacinė sistema turi joje veikiančią interneto naršyklę. Taigi taip apipavidalinta medžiaga yra nesunkiai prieinama visiems.

Pagrindinės HTML komponentės:

1. Stilių lentelių rinkmenos (css) ;
2. Skriptai (js, vbs);
3. Dokumento tekstas ir struktūra (html).

HTML dokumento variacijos:

1. Stilių lentelės, skriptai ir dokumento tekstas vienoje byloje;
2. Stilių lentelės ir/arba skriptai nenaudojami;
3. Stilių lentelės ir/arba skriptai, bei dokumento tekstas atskirose rinkmense.

Peržiūrint HTML dokumentą vartotojui jo vidinė struktūra yra nematoma. Programos, skirtos hipertekto dokumento peržiūrai, pačios vaizduodamos dokumentą, surenka informaciją iš reikiamų rinkmenų.

HTML rezultatas, gaunamas vienu iš šių būdų:

1. Viską html rašome rankomis;
2. Dalį rašome rankomis, dalį - su automatiniais įrankiais;
3. Viską generuojame su automatiniais įrankiais.

HTML problemos:

1. Su MS Office sukurti ir kaip Web Page išsaugotos rinkmenos;
2. Su FrontPage sukurtos rinkmenos;
3. Kitomis automatinio html generavimo priemonėmis sukurti dokumentai.

Kiekviena automatizuota HTML generavimo programa dažnai prideda daug savo žymių, todėl dokumentas tampa sunkiai analizuojamas. Dažnai automatinės priemonės naudoja savo žymes (tagus) ir nesilaiko W3C nustatytų standartų.

Automatinės HTML generavimo sistemos sukurtas dokumentas yra labai sudėtingas ir po to net ir HTML gerai išmanantis žmogus gali susidurti su rimtomis problemomis redaguodamas dokumentą.

Kaip pavyzdį palyginimui pateiksime dokumentą, sukurtą tiesiogiai programuojant HTML ir sudarytą naudojant MS Word XP programą:

```
<html>
<p align=center >
    <b><i>Sveiki gyvi</i></b>
</p>
</html>
```

```
<html>
<head>
<meta http-equiv=Content-Type content="text/html;
charset=windows-1257">
<meta name=Generator content="Microsoft Word 11 (filtered)">
<title>Sveiki gyvi</title>
<style>
<!--
```

```

/* Style Definitions */
p.MsoNormal, li.MsoNormal, div.MsoNormal
    {margin:0cm;
margin-bottom:.0001pt;
font-size:12.0pt;
font-family:"Times New Roman";}
@page Section1
    {size:612.0pt 792.0pt;
margin:72.0pt 90.0pt 72.0pt 90.0pt;}
div.Section1
    {page:Section1;}
-->
</style>

</head>

<body lang=EN-US>
<div class=Section1>
<p class=MsoNormal align=center style='text-align:center'><b><i><span lang=LT>Sveiki
gyvi</span></i></b></p>
</div>
</body>
</html>

```

Abiem atvejais sukurtas dokumentas atrodo taip pat.

Sveiki gyvi!

Kadangi dauguma varotojų, ruošiančių mokymosi medžiagą nemoka HTML kalbos, tai jų paruošti hiperteksto dokumentai yra sukurti automatinėmis priemonėmis. Todėl pataisyti juos rankiniu būdu tampa labai sudėtinga. Šiai problemai spręsti reikalingos specialios programos, nereikalaujančios iš vartotojo HTML programavimo žinių.

Formali hiperteksto patikra

HTML dokumento patikrai naudojamos DTD rinkmenos, kuriuose aprašytos HTML taisyklės. Aprašyti visi elementai ir atributai bei jų reikšmių aibės.

Hiperteksto dokumentų peržiūra

HTML dokumentų peržiūrai naudojamos specialios programos, vadinamos interneto naršyklėmis.

Populiariausios interneto naršyklės:

1. Internet Explorer;
2. Mozilla;
3. Netscape;
4. FireBird;
5. Kongueror.

Problemos, kai su skirtingomis naršyklėmis peržiūrimas tas pats dokumentas, kyla todėl, kad, konkuruodamos tarpusavyje, jos nesilaiko standartų ir naudoja savo atskirus žymėjimus. Tokių žymėjimų nesupranta kitos naršyklės, ir dokumentas negali būti teisingai matomas visiems vartotojams. Formaliai analizuojant HTML dokumentą, patartina remtis W3C konsorciumo standartais.

6. HTML nuotolinio mokymosi studijose

Kadangi nuotolinėse studijose studentas ir dėstytojas dažniausiai tiesiogiai nebendrauja, reikalinga paprasta galimybė perduoti informaciją dideliais atstumais. Studentas turi turėti galimybę pasinaudoti mokymosi medžiaga jam prieinamu laiku norimoje vietoje. Medžiaga turi neužimti daug vietos kompiuterio laikmenose. Mokymosi 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 ir vaizdo įrašus. Hierarchinis medžiagos išdėstymas, išsiskojimai ir detalizacijos principas leidžia imituoti intelektualią mokymosi aplinką ir prisitaikyti prie individualių studentų savybių bei skirtingo jų pasiengimo lygio. Iškilus neaiškumams, studentas gali paprašyti pateikti detalesnę informaciją, kuri gali būti aktuali ne visiems, nes įvairiems studentams 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 naudojamas ateityje plėsis.

7. Hipertekstu pateiktos medžiagos pritaikymo neįgaliesiems taisyklių sudarymas

.

Paaiškinimai

Prioritetai suskirstyti pagal W3C konsorciumo reikalavimus:

1 Prioritetas – Privaloma taisyklė. Jos nesilaikymas gali būti vertinamas kaip klaida. Žmonės su negalia praktiškai negalėtų naudodami tam tikromis dokumento dalimis.

2 Prioritetas – Būtina taisyklė. Žmonės su negalia turės nemažą sunkumą dirbdami su dokumentu.

3 Prioritetas – Rekomenduojama taisyklė. Žmonės su negalia gali turėti nepatogumų dirbdami su dokumentu.

HTML žymėjimai - stulpelis, nurodantis, kuriems žymėjimams taikoma sudaryta taisyklė.

Sąlyga – formali sąlyga, kuri turi būti tenkinama pagal taisyklę.

WCAG 1.0

TAISYKLĖS	Prioritetas	HTML žymėjimai	Sąlyga
Grupė 1. Aprašyti alternatyvas garsiniams ir vaizdiniam komponentams.			
1.1. Pateikti teksto atitikmenį ne tekstiniams komponentams.	P1	IMG, AREA, OBJECT,APPLET, FRAME, FRAMESET, INPUT	Ar egzistuoja atributai alt,longdesc?
		OBJECT	Ar egzistuoja vidinė HTML žymė NOEMBED?
1.2. Pateikti papildomas teksto nuorodas kiekvienam aktyviam serverio pusės paveikslelių žemėlapiu (server-side image map) regionui.	P1		
1.3. Kol vartotojo agentas automatiškai balsu skaito vaizdinės informacijos tekstinę alternatyvą, pasirūpinkite ne tik turinio, bet ir vaizdinės informacijos (paveikslėlių, nuotraukų) apibūdinimu balsu.	P1		
1.4. Kiekvienai laiku paremtai multimedia prezentacijai reikia pateikti sinchronizuotą alternatyvą (titrai ir t.t.).	P1		
Grupė 2. Protingai naudoti spalvas.			
2.1. Užtikrinti, kad visa informacija, išreikšta spalvomis, taip pat būtų prieinama ir be spalvų.			
2.2. Užtikrinti, kad fono ir teksto ar objektų spalva netrukdytų žmonėms, turintiems negalių arba žiūrintiems per juodai baltą monitorių. Daltonizmu sergantys žmonės blogai skiria RAUDONĄ, ŽALIĄ ir VIOLETINĘ spalvas,	P2 paveikslėliai	Visos HTML žymės, turinčios atributus, nurodančius fono ir teksto spalvas.	Atributai, nurodantys teksto ir fono spalvas, nelygūs raudonai, žaliai ir violetinei

todėl geriausia visai jų atsisakyti.	P3 tekstas		spalvoms.
Grupė 3. Tinkamai naudoti žymes (tagus) ir stilių lenteles			
3.1. Jeigu atitinkama formali žymėjimo kalba egzistuoja naudoti vietoj paveikslėlių informacijai pateikti. MATHML kalba skirta matematinių formulių aprašymui.	P2	IMG	Ar paveikslėlis nevaizduoja matematinių formulių?
3.2. Sukurkite dokumentus, kurie tikrina formalų gramatikos teisingumą. Tai DTD rinkmenos HTML kalbai. Turėtų būti tokia eilutė dokumento pradžioje: <code><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd"></code>	P2	!DOCTYPE	Ar egzistuoja HTML žymė?
3.3. Naudoti stilių lenteles turinio struktūros apibrėžimui ir pateikimui. Reikėtų naudoti stilių lenteles vietoje B I STRONG	P2	LINK	Ar atributo rel reikšmė lygi STYLESHEET ?
3.4. Naudoti reliatyvias matavimo vienetų reikšmes vietoj absoliučių. Pvz: Procentus pagal bendrą lango dydį.	P2	Visos HTML žymės, turinčios atributus, nurodančius skaitinius dydžius.	Ar atributai, kurie aprašo skaitinius dydžius, gale turi % ženklą?
3.5. Naudoti antraštės elementus, išreiškiančius dokumento struktūrą. Naudoti juos pagal specifikaciją.	P2	H1 – H6	Ar turi šią žymę kiekviena dokumento dalis?
3.6. Tvarkingai žymėti sąrašus ir sąrašų elementus. Nesunumeruoti sąrašai gali sukelti problemų žmonėms, naudojantiems ekrano skaitymo priemones. Nes ne visada aišku, kur baigiasi vienas elementas ir prasideda kitas. Pataisymas - pakeisti sunumeruotą sąrašą į OL.	P2	OL vietoj UL	Ar egzistuoja HTML žymė?
3.7. Citavimas. Nenaudokite citavimo žymėjimo efektams, tokių kaip atitraukimas nuo krašto formatavimui	P2	BLOCKQUOTE, Q	Ar egzistuoja bent vienas atributas, pvz.: cite ?
Grupė 4. Aiškiai naudoti kalbos struktūras			

4.1. Aiškiai nustatyti dokumento kalbą ir kalbos pasikeitimus.	P1	Teksto žymės kaip P ir t.t.	Ar egzistuoja atributas lang , jei naudojama kita nei pagr. dok. kalba?
4.2. Nurodykite kiekvieno akronimo arba sutrumpinimo aprašymą, kai jis pirmą kartą panaudojamas. Dažniausiai naudojamas duomenų lentelių antraščių sutrumpinimams.	P3	TH	Ar egzistuoja vidinė HTML žymė ABBR ?
4.3. Nurodyti dokumente naudojamą kalbą.	P3	HTML	Ar egzistuoja atributas lang ?
Grupė 5. Taisyklingai naudoti lentelės.			
5.1. Duomenų lentelėms nurodyti eilučių bei stulpelių antraštes (headers).	P1	TABLE, TR	Ar egzistuoja vaikinis HTML žymėjimas TH
5.2. Duomenų lentelėms, kurios turi du ir daugiau antraščių loginių lygių, naudokite žymėjimą susieti duomenų ir antraščių celėms.	P2	TD	Ar egzistuoja atributas headers ?
5.3. Nenaudoti lentelių puslapio struktūrai nurodyti.	P2	TABLE	Neegzistuoja HTML žymė arba ji naudojama duomenims vaizduoti.
5.4. Lentelėms, naudojamoms dokumento struktūrai apibrėžti, aprašymui nenaudoti duomenų lentelėms naudojamas žymas, tokias kaip antraštės, apibendrinimas.	P2	TABLE	Ar HTML žymė neturi vidinių žymių CAPTION, TH ?
5.5. Pateikti lentelių antraštę ir santraukas.	P3	TABLE	Ar HTML žymė turi atributus title ir summary ?
5.6. Pateikti antraščių sutrumpinimus.	P3	TH	Ar HTML žymė turi vidinę HTML žymę ABBR ?
Grupė 6. Užtikrinti taisyklingą puslapio transformaciją.			
6.1. Organizuoti dokumentus taip, kad juos būtų galima peržiūrėti be stilių	P1		

lentelių.			
6.2. Užtikrinti, kad ir tekstiniai dinaminio turinio ekvivalentai kinta kintant vizualiniam turiniui. Skaidant hiperteksto dokumentus į freimus, reikėtų nurodyti tik hiperteksto dokumentus.	P3	FRAME	Ar HTML žymės atribute src nurodytas hiperteksto dokumentas?
6.3. Užtikrinti, kad puslapiai su skriptais, apletais ar kitais programiniais objektais yra prieinami ir turi alternatyvias priemones tokiu atveju, jei dinaminiai objektai būtų nepalaikomi arba išjungti. Skriptams reikalingi atitikmenys skriptų nesuprantančioms programoms.	P1	OBJECT, APPLET	Ar HTML žymės turi tekstinius atitikmenis?
		SCRIPT	Ar egzistuoja kita HTML žymė NOSCRIPT ?
6.4. Skriptuose užtikrinti, kad įvykių apdorojimo mechanizmas būtų nepriklausomas nuo platformos.	P2		
6.5. Užtikrinti, kad dinaminis turinys yra prieinamas neįgaliems arba yra pateiktas alternatyvus puslapis. Būtinai alternatyvūs dokumentai naršyklėms, identifikuojančioms save kaip nurodyta LINK žymės MEDIA atributo reikšmėje, t.y., jei programa padeda aklam žmogui, tai turi atsidaryti vienoks puslapis, jei kurčias - kitoks.	P2	FRAMESET	Ar HTML žymė turi vidinę žymę NOFRAME ?
		LINK	Ar HTML žymė turi atributą rel="alternate" ?
Grupė 7. Užtikrinti pilnas puslapio valdymo galimybes.			
7.1. Išvengti ekrano mirgėjimo.	P1		
7.2. Išvengti dokumento turinio mirgėjimo.	P2		
7.3. Išvengti nuolat judančių dokumento dalių (animacija ir t.t.).	P2	IMG	Ar nurodomi gif tipo animuoti paveikslėliai src atribute?
		EMBED	Ar nurodomi flash tipo animuoti paveikslėliai src atribute?
7.4. Išvengti periodiškai atnaujinamų (perpiešiamų) puslapių.	P2	META	Ar atributas http-equiv

			lygus "refresh"?
7.5. Nenaudoti HTML elementų, automatiškai nukreipiančių į kitą dokumentą. Geriau tokioms operacijoms konfiguruoti WEB serverį.	P2	META	Ar atributai http-equiv lygus "refresh" content="5;http://www.example.com/newpage" .
Grupė 8. Užtikrinti tiesiogini priėjimą prie išplėsto vartotojo interfeiso.			
8.1. Programinius elementus, tokius kaip skriptai ir apletai, padaryti tiesiogiai pasiekiamus arba suderinamus su pagalbinėmis programomis.	P1/P2		
Grupė 9. Kurti nuo įrangos nepriklausomą dokumentą.			
9.1. Naudoti kliento pusės paveikslėlių žemėlapius vietoj serverio pusės, nebent regiono neįmanoma apibrėžti geometriškai.	P1	IMG	Nenaudojamas atributas ismap .
9.2. Užtikrinti, kad kiekvienas elementas turėtų savo interfeisą ir galėtų veikti nepriklausomai nuo platformos.	P2		
9.3. Skriptams naudoti loginias įvykių apdorojimo funkcijas (event handlers) vietoj nuo platformos priklausomų įvykių apdorojimo funkcijų (event handlers).	P2		
9.4. Sukurti logiškas TAB perėjimo sekas nuorodoms, formų elementams ir objektams.	P3	INPUT	HTML žymė turi atributą tabindex .
9.5. Sukurti klaviatūros klavišų sąryšius ypatingoms nuorodoms, kliento pusės paveikslėlių žemėlapiams, formų komponentams ir jų grupėms	P3	A , formos elementai	HTML žymė turi atributą accesskey .
Grupė 10. Naudoti tarpinius pranešimus.			
10.1. Nenaudoti iššokančių pop-up langų ir nekeisti aktyvaus lango be papildomo vartotojo perspėjimo.	P2	A, AREA, FORM, FRAME	HTML žymės turi atributą target=_blank .
10.2. Visiems formos elementams užtikrinti vienareikšmišką aprašymą ir	P2	LABEL	HTML žymė turi atributą

išdėstymą. Įvedimo elementus surišti su juos aprašančiomis antraštėmis.			for.
10.3. Priskirti pradines, vartotojui palengvinančias formos įvedimo elementų reikšmes.	P3	INPUT	HTML žymė turi atributą value.
10.4. Prie kievienos nuorodų grupės pridėti vieną nuorodą, leidžiančią praleisti nuorodų grupę ir peršokti prie kitos informacijos.	P3	A	Pirma HTML žymė nuorodų grupėje skirta ją praleisti.
Grupė 11. Naudoti W3C technologijas ir taisykles.			
11.1. Naudoti W3C technologijas (jeigu tokios yra) ir stengtis naudoti tik paskutines versijas	P2		
11.2. Išvengti W3C technologijoms prieštaraujančių priemonių naudojimo.	P2	APPLET, BASEFONT, CENTER, DIR, FONT, ISINDEX, MENU, S, STRIKE, U	Nenaudojamos nurodytos su HTML 4.1 specifikacija nesuderinamos HTML žymės.
11.3. Pateikti informaciją, kad vartotojas galėtų gauti dokumentą pagal tam tikrus parametrus.	P3	LINK	Ar HTML žymė turi atributą rel , lygų "alternate"?
11.4. Jeigu dėl tam tikrų priežasčių nepavyksta dokumento padaryti prieinamu neįgaliesiems, pateikite nuorodą į alternatyvų dokumentą, turintį ekvivalenčią informaciją ir atnaujinamą kartu su pagrindiniu.	P1		
Grupė 12. Pateikti konteksto ir navigacijos informaciją.			
12.1. Pavadinti kiekvieną freimą, kad būtų palengvinta informacijos indentifikacija ir navigacija.	P1	FRAMESET, FRAME	HTML žymė turi atributą title.
12.2. Pateikti kiekvieno rėmelio (frame) aprašymą ir paskirtį bei ryšį su kitais rėmeliais	P2	FRAME	HTML žymė turi atributą name.

12.3. Suskaidyti didelius informacijos blokus į mažesnius, stengtis grupuoti ir padaryti suprantamesnius.	P2	FORM	HTML žymė turi vidines žymes FIELDSET , OPTGROUP .
12.4. Aiškiai susieti žymėjimus su jų komponentais.	P2	LABEL	HTML žymė turi atributą for .
Grupė 13. Pateikti aiškų navigacijos mechanizmą.			
13.1. Suprantamai apibrėžti tai, į ką rodo nuoroda.	P2	A	HTML žymė turi atributą title .
13.2. Pateikti meta HTML žymes, tam kad būtų galima pridėti semantinę informaciją. Būtina nurodyti dokumento antraštę ir autorių.	P2	TITLE, ADDRESS	Ar egzistuoja HTML žymės?
13.3. Pateikti informaciją apie puslapio išdėstymą ir turinį.	P2		
13.4. Pateikti lengvai suprantamą ir nuoseklų navigacijos mechanizmą.	P2		
13.5. Navigacijos priemonės pateikti paryškintas ir gerai matomas.	P3		
13.6. Grupuoti susijusias nuorodas, identifikuoti grupes (vartotojo agentams). Pateikti galimybę praleisti grupę.	P3		
13.7. Jeigu naudojama paieškos funkcija, sukurti kelių lygių paieškos galimybes apsižvelgiant į vartotojo galimybes ir patyrimą.	P3		
13.8. Pateikti būdingą informaciją prieš kiekvieną paragrafą, sąrašą ir t.t.	P3		
13.9. Pateikti informaciją apie visas į dokumentą įeinančias dalis, esančias kituose dokumentuose.	P3	LINK	Ar egzistuoja HTML žymė ir jos atributuose nurodyti visi susiję dokumentai?
13.10. Pateikti galimybę praleisti ASCII tipo grafinius elementus.	P3		
Grupė 14. Užtikrinti, kad dokumentas yra aiškus ir paprastas.			
14.1. Naudoti kuo aiškesnę ir paprastesnę kalbą dokumento turiniui.	P1		

14.2. Pridėti papildomus grafinius elementus, jei tai padeda suprasti dokumento tekstą	P3		
14.3. Sukurti bendrą stilių lentelę visoms dokumento dalims, sukurtoms atskirose rinkmenose.	P3		
14.4 Užtikrinti, kad dokumento tekstas būtų suskaidytas į logiškus vienetus. Labai ilgas vientisas tekstas sudėtingai skaitomas ir įsisavinamas.	P1	P ir kitos tekstą formatuojančios HTML žymės.	Teksto ilgis pastraipoje (tarpą atidarantios ir uždarančios HTML žymės) mažesnis už tam tikrą reikšmę.

8. Analizės rezultatai

Buvo išanalizuoti:

- Neigalių žmonių darbo su hiperteksto dokumentais būdai ir problemos;
- Mokymosi medžiagos pateikimo hipertekstu būdai ir problemos;
- Hiperteksto dokumentų svarba nuotoliniame mokyme.

Sudarytas formalių taisyklių rinkinys, leidžiantis technologiškai įvertinti mokymosi medžiagos tinkamumą neįgaliesiems.

Išvada: Reikalingas įrankis, leidžiantis mokymosi medžiagą ruošiančiam žmogui efektyviai pritaikyti ją neįgaliesiems.

Analizės metu išryškėjo pagrindiniai įrankio kriterijai:

- Paprastumas naudoti – nereikalingos HTML programavimo žinios;
- Galimybė plėsti be įrankio kodo perrašymo – lankstus konfigūravimas;
- Integracija su interneto naršyklėmis, WYSIWYG darbo principas.

9. Panašių programų ir sprendimų analizė bei palyginimas

Atsižvelgiant į analizės metu suformuotus kriterijus buvo atrinkti galimi jau egzistuojantys sprendimai. Visi sprendimai realizuoti kaip įrankiai, skirti padėti hiperteksto dokumentus pritaikyti žmonėms su negalia. Įrankių analizės metu buvo bandoma sudaryti kiekvieno iš jų privalumų bei trūkumų sąrašą, bei atitikimų analizės suformuluotiems reikalavimams lentelę. Pagrindinė sprendimų paieškos vieta buvo interneto svetainės:

- www.google.lt - populiariausia interneto paieškos svetainė;
- www.w3.org - World Wide Web konsorciumas (W3C). Tai tarptautinis konsorciumas, kurio organizacijų - narių tikslas - dirbti kartu, kuriant interneto standartus.

9.1. Nagrinėti sprendimai

[AccRepair™](#)

Skirtas internetinių puslapių įgalinimo (Accessibility) įvertinimui ir rastų neatitikimų koregavimui. Programa palaiko 3 WCAG lygius. AccRepair gali veikti ir kaip atskira programa, ir kaip Microsoft FrontPage, Microsoft Office 2000 ir XP programų įskiepis. Su programa pateikiamas API funkcijų ir pavyzdžių rinkinys. Programa mokama.

[CSS Accessibility Analyser](#)

Ši programa leidžia atlikti stilių lentelių analizę pagal W3C reikalavimus. Pagrindiniai atliekami testai yra spalvų ir šriftų dydžių.

[Web Accessibility Toolbar](#)

Programa sukurta kaip interneto naršyklės Internet Explorer įskiepis. Ji skirta internetinių puslapių įgalinimo įvertinimui. Palaiko WCAG 1.0 standartą. Turi galimybę imituoti vartotojo elgseną. Pateikia nuorodas į galimus problemų sprendimus. Su programa kartu platinamas plati pagalbos rinkmena.

[AccessEnable™](#)

Tai per internetą pasiekiamą programą, skirtą internetinių puslapių įgalinimo įvertinimui ir rastų neatitikimų koregavimui. Palaiko WCAG internetinių puslapių įgalinimo standartus.

Accessibility Wizard

Tai programa, skirta internetinių puslapių kūrėjams ir kūrėjų komandoms. Programa sukurta „Flash“ technologijos pagrindu. Programos pagalba galima paskirstyti darbus projekto dalyviams. Palaiko WCAG 1.0 ir 508 standartus.

1 lentelė. Sprendimų palyginimas pagal analizės kriterijus

Sprendimai Kriterijai	AccRepair	Web Accessibility Toolbar	CSS Accessibility Analyser	AccessEnable	Accessibility Wizard
Nereikalingos HTML žinios	TAIP	NE	NE	TAIP	TAIP
WYSIWYG darbo principas	TAIP	NE	NE	TAIP	NE
Nemokama	NE	TAIP	TAIP	NE	TAIP
Automatinis klaidų taisymas	TAIP	NE	NE	NE	NE
Galimybė pridėti ir keisti taisykles	NE	NE	NE	NE	NE
Integracija su interneto naršyklėmis	NE	TAIP	TAIP	NE	NE

Taigi analizuodami lentelę matome, kad nei vienas iš nagrinėtų sprendimų neatitiko analizės dalyje iškeltų reikalavimų. Nei vienas įrankis neturi priemonių papildomos taisyklės įvesti, todėl norėdami tai padaryti vartotojai turi kreiptis į programos kūrėjus kurie gali ir nesutikti tai padaryti. Tik „AccRepair“ ir „AccessEnable“ turi WYSIWYG vartotojo sąsają ir nereikalauja HTML žinių, tačiau jie yra mokami.

9.2. Išvados:

Kadangi nei vienas iš nagrinėtų sprendimų neatitiko visų keliamų reikalavimų, nuspresta suprojektuoti ir sukurti naują įrankį, kuris atitiktų visus analizės metu iškeltus reikalavimus. Pagrindiniai kriterijai, kurių neatitiko nagrinėti įrankiai, buvo:

- Ar įrankis nemokamas;
- Galimybė pridėti ir keisti taisykles.

10. Hipertekstu pateiktos medžiagos analizės įrankio projektavimas

10.1. Technologijų analizė

Ü HTML ir XHTML

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 sukurtas dar 1980-1984 metais ir patvirtintas ISO 8779 standartu. SGLM kalba vartojama pavyzdžiui, siekiant standartizuoti didelių tarptautinių organizacijų raštvedybą ir tarpusavio susirašinėjimą.

"Interneto" standartu tapusi HTML - tai ne programavimo kalba ir ne griežtas dokumento formatas. HTML visų pirma aprašo loginę WWW puslapių struktūrą: dokumentų bei juos sudarančių skyrių ir skirsnių antraštes, pastraipas, iliustracijas, lenteles, nuorodas į kitus dokumentus ar kitokius duomenis ir t.t.

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 fizinės dokumento savybes kaip antai, vartojamo šrifto parametrus, lentelių, iliustracijų bei kitų elementų dydžius ir pan.

W3C XHTML, HTML tęsėjas. XHTML turi daug tokių pat elementų kaip ir HTML. Sintaksė buvo truputį pakeista, kad atitiktų XML taisykles. Formatas, "XML pagrindu" paveldėjo XML sintaksę ir kai kuriais atvejais ją draudžia (pvz., XHTML leidžia "<p>", tačiau ne "<r>"); jis taip pat suteikia prasmę šiai sintaksei (XHTML sako, jog "<p>" reiškia "paragrafą", o ne "parametrą", "puslapį" ar dar ką nors).

Ü XML

XML skirta duomenims sisteminti. Sistemizuotiems 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 sistemizuoti jūsų duomenis. XML nėra programinė kalba ir jūs neprivalote būti programuotojas, norėdamas ją naudoti ar jos išmokyti. XML 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 atrodo naršyklėje), XML naudoja tegus tik atskirti duomenų dalis. Duomenų interpretavimą palieka taikomajai programai, kuri juos nuskaityti. Kitaip tariant, jei matote "<p>" XML byloje, nemanysite, kad tai paragrafas. Priklausomai nuo konteksto, tai gali būti parametras, puslapis, p... (kai kas sako, jog tai turi būti žodis su "p"?).

Ü Microsoft .NET Framework

Microsoft .NET arba paprastai .NET - nauja technologija ir naujas programų kūrimo ir palaikymo būdas, kuris naudoja tokius standartus kaip HTTP ir XML, kad užtikrintų neribojamą IT sistemų mastą. Ypatinga NET dalis yra .NET Framework, tai yra platforma .NET programų kūrimui ir eksploatavimui. .NET Framework nėra būtinas kuriant programas, bet jis padaro programų kūrimą lengvesnį ir greitesnį. .NET Framework - tai tam tikrų klasių rinkinys, skirtas palengvinti programuotojų darbą.

Ü Microsoft SQL Server

Microsoft SQL Server 2000 yra duomenų bazių valdymo sistema, sudaryta iš duomenų bazės valdymo sistemos ir galingų įrankių. Microsoft SQL Server 2000 - integruotas įrankių rinkinys XML palaikymui. Microsoft SQL Server 2000 yra patikima,

lanksti ir lengvai valdoma DB sistema. Microsoft SQL Server 2000 MSDE tam tikra versija, skirta programoms, kurioms nereikia pilno Microsoft SQL Server 2000 paketo. Microsoft SQL Server 2000 MSDE yra nemokama programa. Microsoft SQL Server 2000 visiškai suderinama su Microsoft Windows Active Directory, kuri leidžia visiškai integruoti duomenų bazių serverį į operacinę sistemą.

Schematinis įrankio veikimo planas:

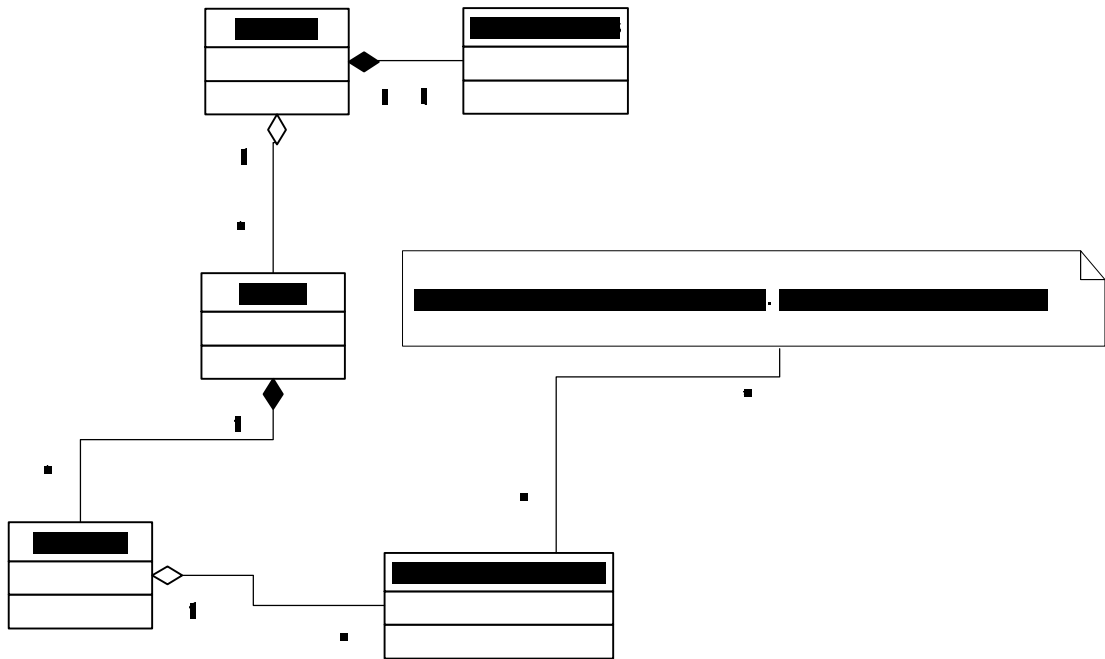
1. Atliekame html sintaksės patikrą;
2. Sudarome HTML žymių hierarchiją;
3. Kiekvienai HTML žymei pagal šalia jo parašytus atributus užpildome atributų lentelę;
4. Pagal turimas stilių lenteles pridedame į HTML žymės atributų lentelę papildomas reikšmes;
5. Turime HTML žymes ir kiekvieno iš atributų lentelę tokią, pagal kurią jis atvaizduojamas;
6. Pritaikome taisykles ir pildome pakeitimų / pasiūlymų sąrašą;
7. Pritaikome / parodome galimus / pasirinktus pataisymus.

10.2. Formalus taisyklių aprašymas

HTML dokumento vidinė struktūra susideda iš apibrėžtos aibės objektų. Visi objektai yra žymės. Jų pagalba formuojama dokumento struktūra. Kiekviena žymė turi atributus, jų aibė taip pat griežtai apibrėžta kiekvienai žymei. Norint formaliai išanalizuoti dokumentą ir jį įvertinti, reikia analizuoti:

- Ø Kiekviena žymė ir jos atributus;
- Ø Žymių išdėstymą ir jų hierarchiją.

Taisyklės turi turėti savyje informaciją, pagal kurią galima būtų išpildyti aukščiau iškeltas sąlygas. Norėdami schematiškai pavaizduoti taisyklių struktūrą, sudarome UML diagramą. Joje geriausiai atsispindi esybės ir ryšiai tarp jų.



3 pav. UML klasių diagrama

Pagrindinės esybės:

- Taisyklė;
- Sąlyga;
- Korekcija.

Taisyklė – formalus analizės metu suformuluotos taisyklės aprašymas. Kiekviena taisyklė susideda iš sąlygos. Taisyklė yra tam tikro tipo.

Pagrindiniai taisyklių atributai:

- Ø Pavadinimas;
- Ø Taisyklės tipas (WCAG ir t.t., 508, EU);
- Ø Prioritetas (P1 P2 P3);
- Ø Svarbumas (warning error info);
- Ø Aprašymas;
- Ø Ryšys su sąlygomis ir korekcija.

Sąlyga – tai tam tikra operacija iš galimų operacijų aibės. Pagal sąlygą nustatoma, ar analizuojamas objektas ją tenkina, jei ne, vadinasi, radome neatitikimą taisyklei. Tikrinant sąlygą naudojama operacija.

Operacijos parametrai gali būti:

- Ø Žymės atributai ir jų reikšmės;
- Ø Žymės;
- Ø HTML struktūros išdėstymai;
- Ø Žymių eilės tvarka ir hierarchija.

Pagrindinės operacijos gali būti:

- Ø | < | = | != | parent | count;
- Ø count< count> (kiekis atributų arba HTML žymių visoje hiperteksto tipo rinkmenoje);
- Ø insidesize (info kiekis tarp žymės atidarymo ir uždarymo);
- Ø size (atributo reikšmės dydis).

Jei parametras nurodomas kaip ANY, tada tikrinimas vyksta su visais objektais.

Korekcija – tai taisyklės sąlygos neatitinkančio objekto modifikavimas taip, kad jis ją atitiktų. Korekcija nurodo, kokią formalią operaciją su hiperteksto objektais reikia atlikti.

Pagrindinės korekcijos operacijos:

- Ø Pridėti objektą;
- Ø Pašalinti objektą su visais vaikiniais elementais;
- Ø Pašalinti objekto atributą;
- Ø Pridėti atributą;
- Ø Pridėti vaikinį objektą;
- Ø Tik patarimas, jokių konkrečių pataisymų;
- Ø Pakeisti objekto atributo reikšmę.

Korekcijų vienai problemai gali būti keletas. Vartotojas privalo išsirinkti vieną iš jų. Korekcijos esybėje sagoma, kiek ir kokių paametrų (jei reikia) ji užklausia vartotojo. Pvz., jei taisoma problema, kai paveikslėlis neturi alternatyvaus teksto, tai vartotojo reikia paklausti vieno parametro – alternatyvaus teksto eilutės.

Taisyklės bus aprašytos duomenų bazėje arba XML formatu.

Problema, kaip aprašyti taisyklės pritaikymą, kai neužtenka vien formalaus sutapimo pagal taisyklę (pvz., lentelė skirta formatui palaikyti ar duomenims rodyti, nes, jei pirmas variantas, nereikia lentelės keisti). Tai išsprendžiama su esybe – *korekcijos klausimu*.

Korekcijos klausimas – tai esybė, kuri apibrėžia klausimą, kurio atsakymas tik taip arba ne. Pagal tai, koks atsakymas, vienareikšmiškai galima nustatyti, ar galima pritaikyti korekciją.

Rastų klaidų pateikimas

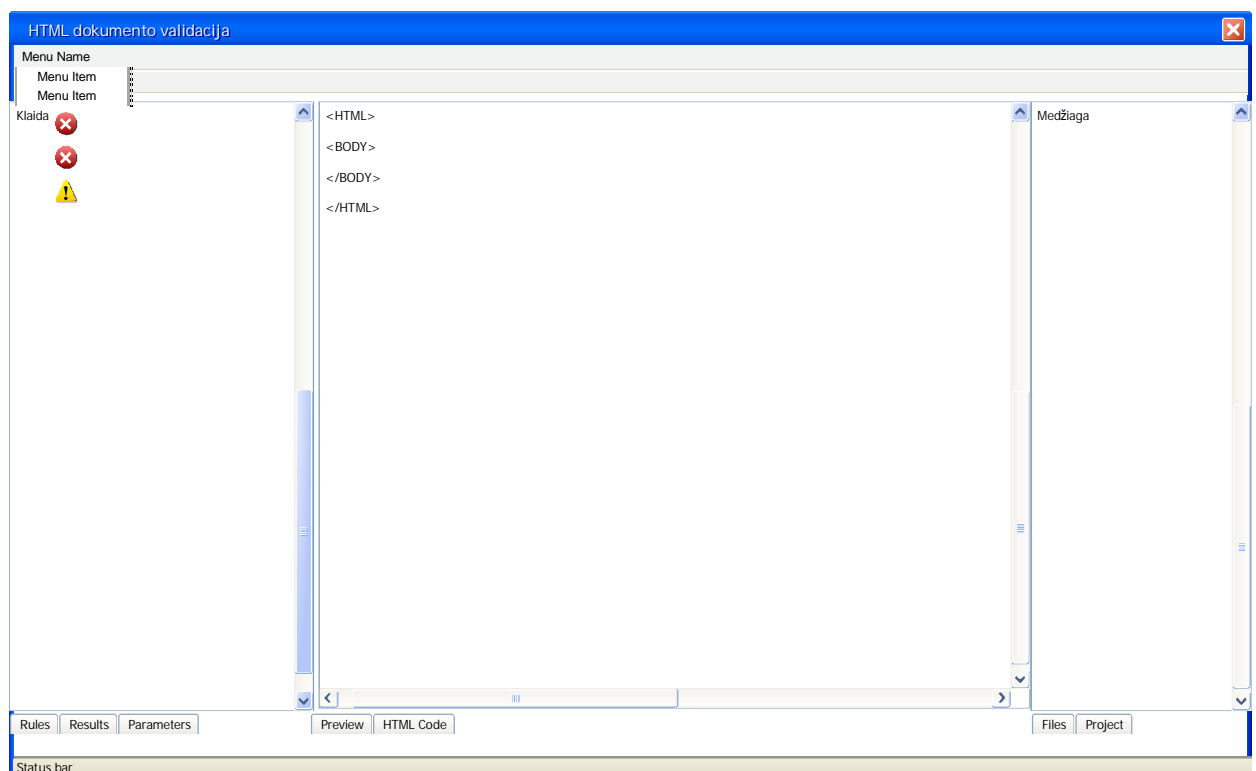
HTML gali būti vaizduojamas 2 pavidalais:

- Ø Tekstinių – parodomas jo vidinis kodas;
- Ø Grafinių – interneto naršyklės pagalba.

Tekstiniame pavidale klaida gali būti rodoma kaip eilutės numeris.

Grafiniame režime rodomas raudonas rėmelis arba fonas aplink objektą.

Programos lango prototipas



4 pav.

Paveikslėlyje Nr. 4 pateiktas įrankio darbo lango prototipas. Jis sudarytas iš 3 dalių:

- Ø Įrankio meniu, kuriame galima pasirinkti, kokią norime atlikti operaciją;
- Ø Klaidų korekcijų sąrašas kairėje lango pusėje, skirtas parodyti analizuojamame dokumente rastas klaidas;
- Ø Analizuojamas dokumentas rodomas vyduryje;
- Ø Dokumento struktūra rodoma dešinėje lango dalyje;
- Ø Apatinėje lango dalyje rodoma būsenų juosta. Joje atsispindi įrankio būsenos ir darbo režimai.

10.3. Technologijų pasirinkimas

- **Dokumento struktūra**

2 Lentelė. DOM technologijų palyginimas pagal kriterijus

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

MSHTML DOM

Technologija pasirinkta, nes tai - populiariausios interneto naršyklės IE branduolys, turintis plačiausias galimybes ir suderinamumą su Microsoft .NET Framework.

- **Taisyklių duomenų struktūra**

3 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.

XML

Technologija pasirinkta, nes tai - pati pažangiausia ir labiausiai palaikoma formalus aprašymo kalba.

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.

MS SQLServer MSDE

Technologija pasirinkta, nes Microsoft SQL Server 2000 MSDE yra nemokama ir visiškai su XML formatu suderinama DBVS.

- **Programavimo kalba ir aplinka**

2 Lentelė. Programavimo technologijų palyginimas pagal kriterijus

Technologija	Kaina	Privalumai	Trūkumai
Microsoft .NET Framework	Nemokama	Platus priemonių pasirinkimas.	Veikia tik windows OS.
JAVA	Nemokama	Nuo platformos nepriklausoma.	Lėta.

Microsoft .NET Framework

Technologija pasirinkta, nes Microsoft .NET Framework pateikia pilną rinkinį klasių darbui su XML ir HTML technologijomis.

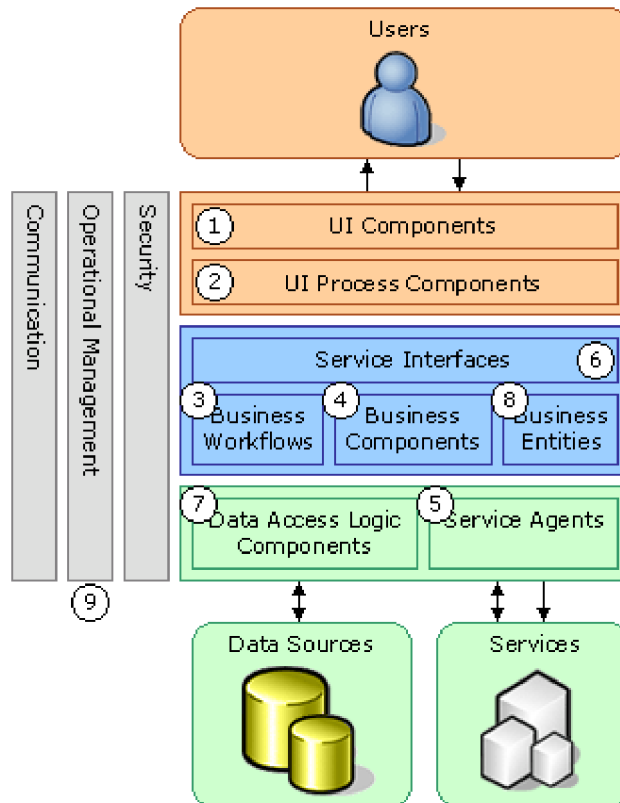
11. Hipertekstu pateiktos medžiagos analizės įrankio kūrimas

11.1. Specifikacija

Įrankis kuriamas hipertekstu pateiktos mokymosi medžiagos analizei. Jis, pagal turimą taisyklių rinkinį įvertinęs puslapį, pateikia problemų sąrašą. Kiekviena problema gali turėti vieną ar kelis pataisymus, kuriuos vartotojas gali pritaikyti ir taip ištaisyti problemas. Įrankis turi suprasti ir tekstiniu redaktoriumi, ir automatiniais įrankiais sukurtus / konvertuotus dokumentus. Įrankis turi suprasti W3C standartų nesilaikančius dokumentus. Vartotojas interaktyviu režimu gali peržūrėti dokumento struktūrą ir iš karto pamatyti atliktus pataisymus komponente. Jo pagalba sukurtą medžiagą bus galima patikrinti, ar ji atitinka norimus standartus. Taisyklių rinkinį bus galima pildyti ir redaguoti. Taisyklių kūrimui ir redagavimui bus sukurtas specialus įrankis. Taisykles jungiamos į profilius. Taisyklių apsikeitimas vyksta per SQL DB arba XML rinkmenas.

11.2. Architektūra

Programos architektūra projektuota pagal 3 lygių architektūrą:

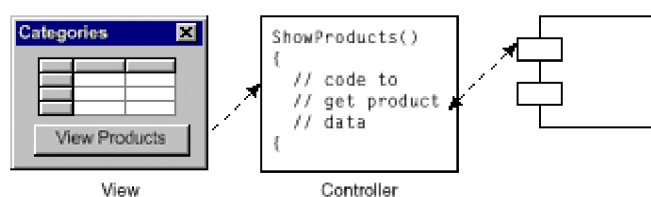


5 pav.

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 objekcinio programavimo priemones.

Architektūrą sudaro 3 lygiai:

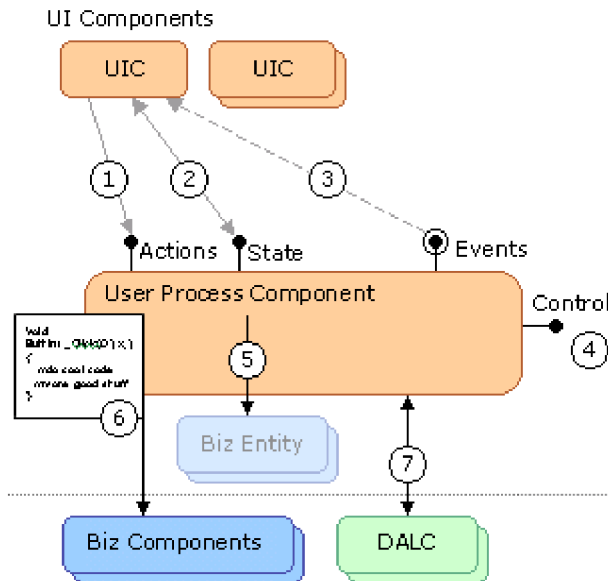
- Prezentacijos lygis (vartotojo sąsaja (5 pav. (1)) ir vartotojo sąsajos valdiklis (5 pav. (2));



6 pav.

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.

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

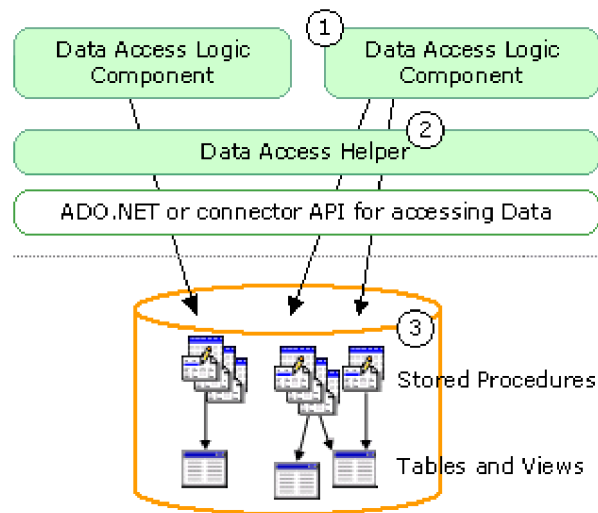


7 pav.

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 taisyklių sąrašą iš duomenų bazės. Iš vartotojo lygio perduodamas hiperteksto dokumento objektas, kuris taip pat patalpinamas veiklos komponentų lygio objektuose. Jeigu įrankis realizuotas kaip Internet Explorer įskiepis, tai veiklos komponentų lygis realizuotas kaip COM + objektas.

Vartotojo sąsajų lygiui operacijų (7 pav. (1)), būsenų (7 pav. (2)) ir įvykių (7 pav. (3)) informacija perduodama per interfeisą.

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

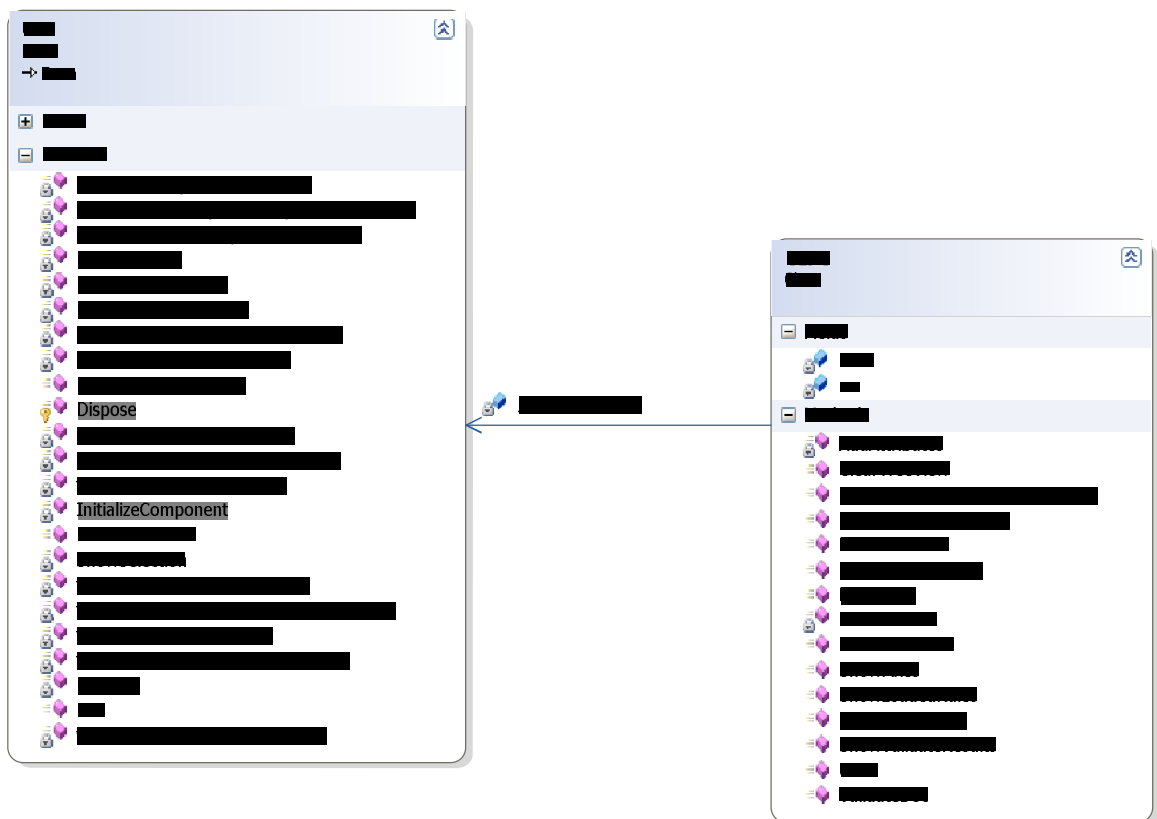


8 pav.

Duomenų lygis skirtas išgauti taisyklių sąrašą iš duomenų bazės. Jis abstrahuoja kitų komponentų darbą su duomenų šaltiniais. Pagal paduotą parametą – taisyklių grupės ID, šis komponentas atlieka užklausą duomenų bazėje ir gražina užklaususiam komponentui iš DB išrinktą duomenų rinkinį.

11.3. Klasių diagrama

Ü Prezencijos lygis



9 pav.

Klasė	Aprašymas
UIC	Vartotojo sąsajos pagrindinė forma.
UIPC	Vartotojo sąsajos valdiklis. Valdo visą sąsają ir papildomas formas. Veikia kaip tarpininkas tarp vartotojo sąsajos ir veiklos komponentų.

FixValues
Class
→ Form

- Fields
 - _fixValuesCount
 - buttonCancel
 - buttonOK
 - components
 - errorProvider
 - groupBox1
 - groupBoxQuestion
 - labelFixVal
 - textBoxFixVal
 - textBoxQuestion
- Properties
 - FixVal
 - FixValuesCount
 - Question
- Methods
 - buttonOK_Click
 - Dispose
 - FixValues
 - FixValues_Load
 - InitializeComponent
 - PrepareFixValBoxes
 - textBox_Validating

FixesForm
Class
→ Form

- Fields
 - _fixes
 - _fixesValues
 - _rowWithFixVal
 - buttonAddFixVal
 - buttonApply
 - buttonCancel
 - components
 - dataGridViewFixes
 - FixName
 - ID
 - richTextBoxFixDesc
- Properties
 - Fixes
 - FixValues
 - SelectedFix
- Methods
 - AddFixValues
 - buttonAddFixVal_Click
 - buttonApply_Click
 - buttonCancel_Click
 - dataGridViewFixes_CellDoubleClick
 - dataGridViewFixes_SelectionChanged
 - Dispose
 - FixesForm
 - FixesForm_Load
 - InitializeComponent

AboutBox
Class
→ Form

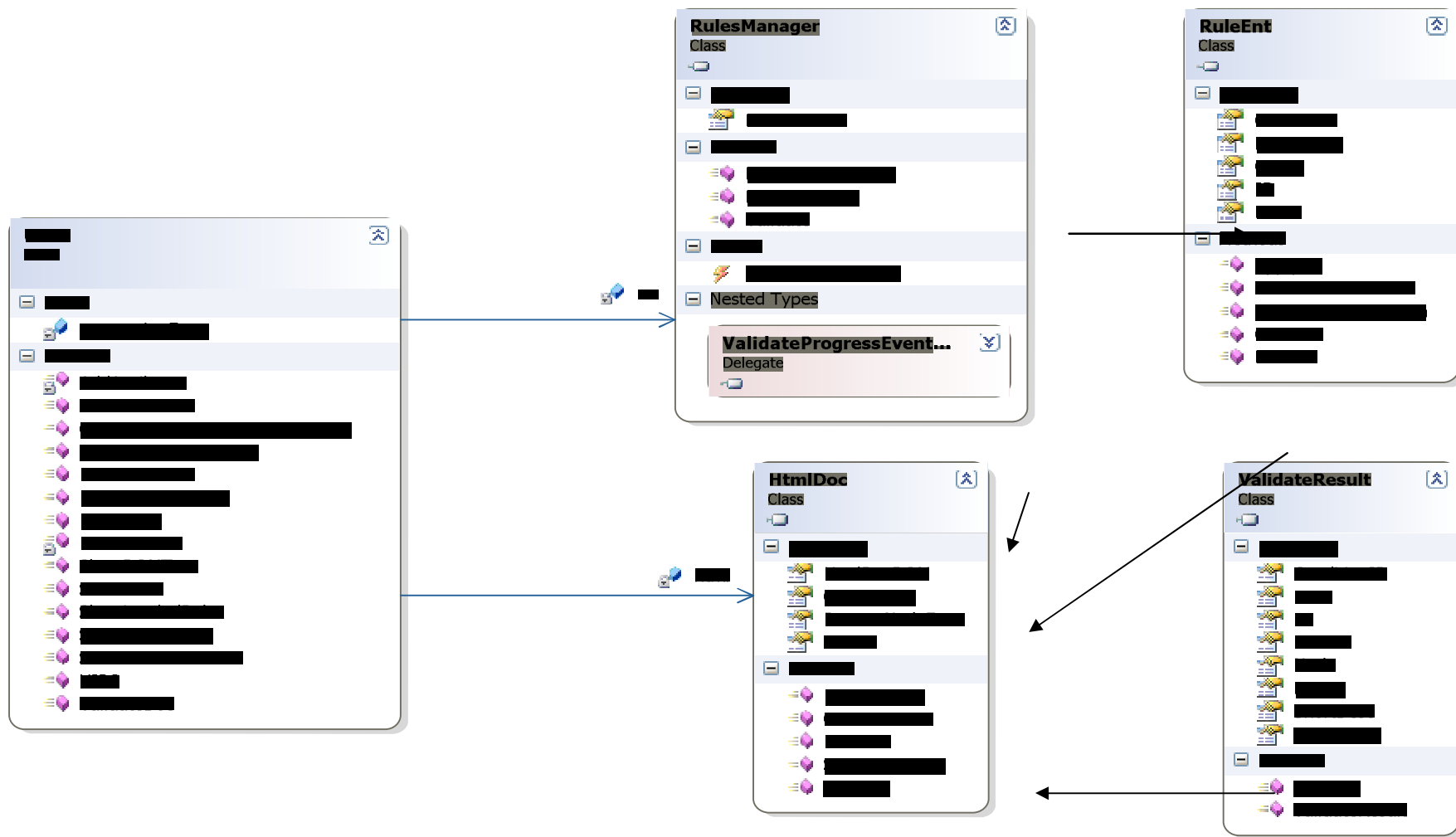
- Fields
 - components
 - labelCompanyName
 - labelCopyright
 - labelProductName
 - labelVersion
 - logoPictureBox
 - okButton
 - tableLayoutPanel
 - textBoxDescription
- Properties
 - AssemblyCompany
 - AssemblyCopyright
 - AssemblyDescription
 - AssemblyProduct
 - AssemblyTitle
 - AssemblyVersion
- Methods
 - AboutBox
 - Dispose
 - InitializeComponent

ValidatingProgress
Class
→ Form

- Fields
 - components
 - groupBoxInfo
 - labelInfoNode
 - labelInfoRule
 - progressBar
- Methods
 - Dispose
 - InitializeComponent
 - UpdateProgress
 - ValidatingProgress

Klasė	Aprašymas
FixValues	Pataisymų reikšmių įvedimo forma.
FixesForm	Pataisymų forma.
AboutBox	Programos aprašymo forma.
ValidatingProgress	Forma, rodanti patikros progresą.

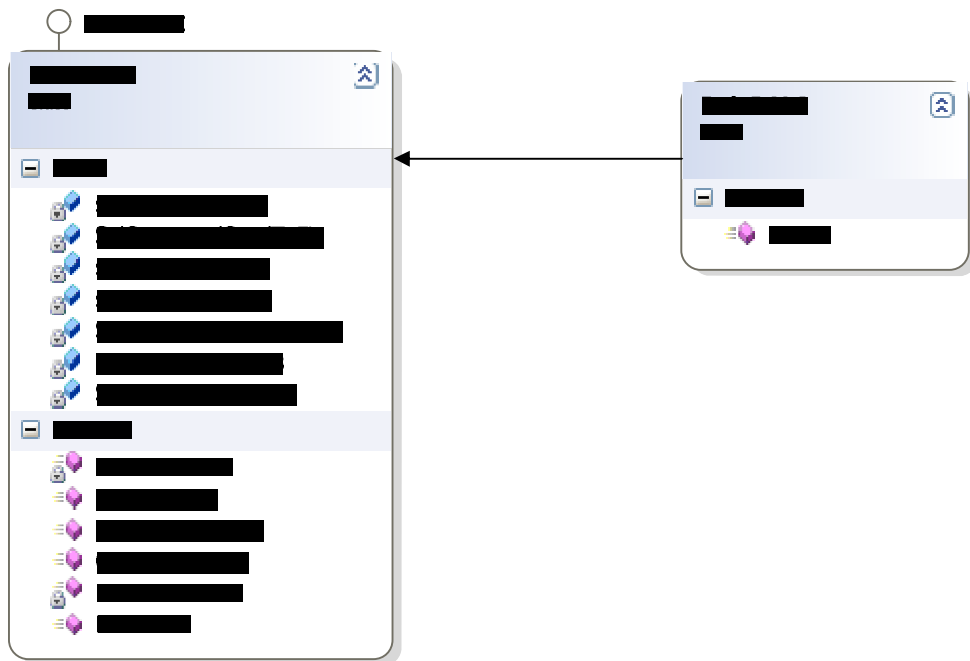
Ü Veiklos komponentų lygis



11 pav.

Klasė	Aprašymas
RulesManager	Pagrindinė klasė, valdanti duomenų užkrovimą ir dokumento patikrą.
HtmlDoc	Dokumentą aprašanti klasė.
ValidateResult	Rastą problemą aprašanti klasė.
RuleEnt	Taisyklę aprašanti klasė.

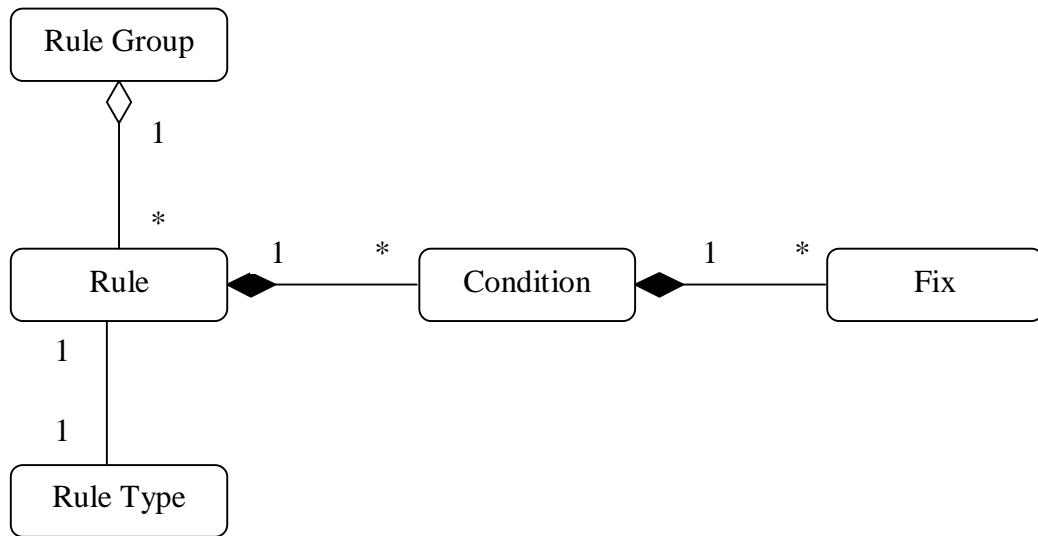
Ü Duomenų lygis



12 pav.

Klasė	Aprašymas
RuleDALC	Klasė, dirbanti su duomenų bazės serveriu.

11.4. Taisyklių aprašymo modelis



13 Pav. Taisyklių aprašymo UML klasių diagrama

Kiekviena taisyklė gali priklausyti taisyklių grupei ir yra tam tikro tipo. Taisyklės, kurios nepriklauso nei vienai taisyklių grupei, negali būti panaudotos patikros metu. Taisyklė susideda iš vienos arba keletos sąlygų. Atskirai be taisyklės sąlyga negali būti naudojama. Kiekviena sąlyga turi vieną ar kelias korekcijas. Korekcijos griežtai susietos su sąlygomis ir atskirai nenaudojamos.

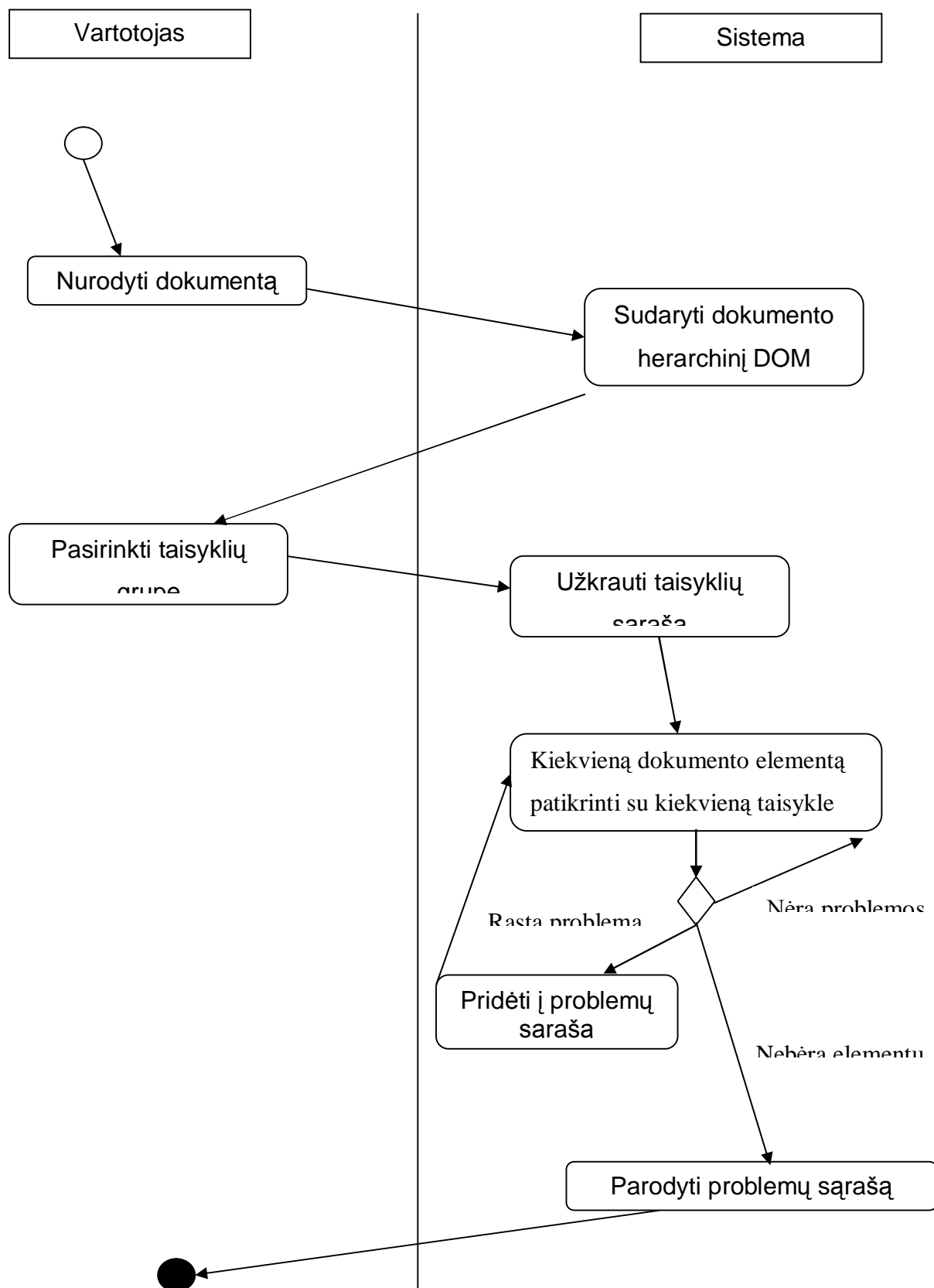
11.5. Formalus įvertinimo pagal taisykles algoritmas

Algoritmo žingsniai

1. Nurodyti dokumentą;
2. Sudaryti dokumento hierarchinį DOM modelį;
3. Pasirinkti taisyklių grupę;
4. Užkrauti taisyklių sąrašą;
5. Kiekvieną dokumento elementą patikrinti su kiekviena taisykle;
6. Jei rasta problema, pridėti ją į problemų sąrašą;
7. Pavaizduoti problemų sąrašą;
8. Pasirinkti problemą;
9. Parodyti pataisymų sąrašą;
10. Išsirinkti pataisymą;
11. Parodyti parametrų įvedimo langą;
12. Įvesti parametrus;

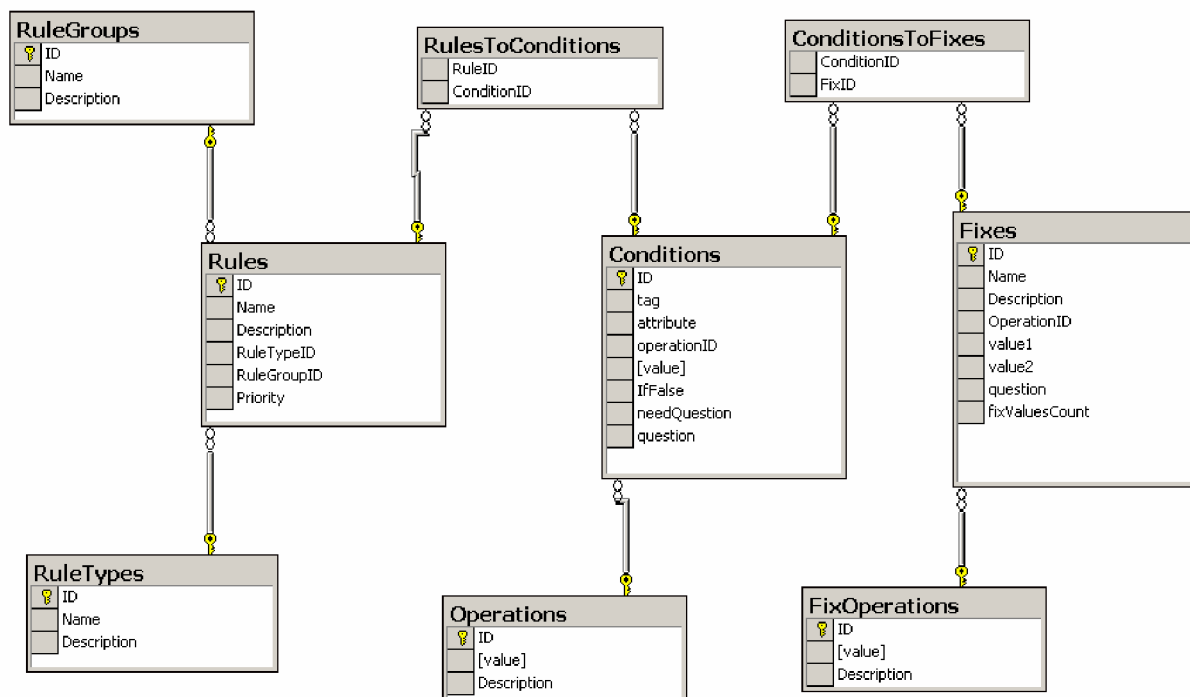
13. Atlikti pataisymą;
14. Atnaujinti pataisyto dokumento vaizdą;
15. Pradėti nuo 7 punkto.

UML diagramoje algoritmas pavaizduotas struktūriškai.



11.6. Duomenų struktūros

Duomenų bazės struktūra:



15 pav.

🚩 Lentelė *Rules*

Saugomos formalaus įvertinimo taisyklės.

Stulpelis	Duomenų tipas	Aprašymas
ID	int	Unikalus identifikatorius
Name	nvarchar	Taisyklės pavadinimas
Description	nvarchar	Taisyklės aprašymas
RuleTypeID	int	Taisyklės tipas, išorinis raktas su lentele <i>RuleTypes</i>
RuleGroupID	int	Taisyklės grupė, išorinis raktas su lentele <i>RuleGroups</i>
Priority	nvarchar	Taisyklės prioritetas

🚩 Lentelė *Conditions*

Saugomos taisyklių sąlygos.

Stulpelis	Duomenų tipas	Aprašymas
ID	int	Unikalus identifikatorius.
tag	nvarchar	Žymė, kuri bus tikrinama.
attribute	nvarchar	Žymės atributas, kuris bus tikrinamas.
operationID	int	Tikrinimo operacija, išorinis raktas ryšiui su lentele <i>Operations</i> .
value	nvarchar	Operacijos argumentas.
ifFalse	nvarchar	Klaidos pranešimas, jei neatitinka sąlygos.
needQuestion	bool	Ar reikia klausti vartotojo?
question	nvarchar	Klausimas vartotojui.

🚩 Lentelė *Fixes*

Saugomi pataisymai taisyklių sąlygoms.

Stulpelis	Duomenų tipas	Aprašymas
ID	int	Unikalus identifikatorius.
Name	nvarchar	Korekcijos pavadinimas.
operationID	int	Korekcijos operacija, išorinis ratas ryšiui su lentele <i>FixOperations</i> .
Value1	nvarchar	Operacijos argumentas.
Value2	nvarchar	Operacijos argumentas.
Question	nvarchar	Klausimas parametrų įvedimui.
fixValuesCount	Int	Kiek parametrų reikia iš vartotojo.

🚩 Lentelė *Operations*

Saugomos sąlygų operacijos.

Stulpelis	Duomenų tipas	Aprašymas
ID	int	Unikalus identifikatorius.
value	nvarchar	Operacijos pavadinimas.
Description	nvarchar	Operacijos aprašymas.

🚩 Lentelė *RuleGroups*

Saugomos taisyklių grupės.

Stulpelis	Duomenų tipas	Aprašymas
ID	int	Unikalus identifikatorius.

Name	nvarchar	Grupės pavadinimas.
Description	nvarchar	Grupės aprašymas.

 Lentelė *RuleTypes*

Saugomi taisyklių tipai.

Stulpelis	Duomenų tipas	Aprašymas
ID	int	Unikalus identifikatorius.
Name	nvarchar	Grupės pavadinimas.
Description	nvarchar	Grupės aprašymas.

 Lentelė *FixOperations*

Saugomos pataisymo operacijos.

Stulpelis	Duomenų tipas	Aprašymas
ID	int	Unikalus identifikatorius.
value	nvarchar	Operacijos pavadinimas.
Description	nvarchar	Operacijos aprašymas.

 Lentelė *RulesToConditions*

Saugomas ryšys tarp taisyklių ir sąlygų.

Stulpelis	Duomenų tipas	Aprašymas
RuleID	int	Taisyklės ID.
ConditionID	int	Sąlygos ID.

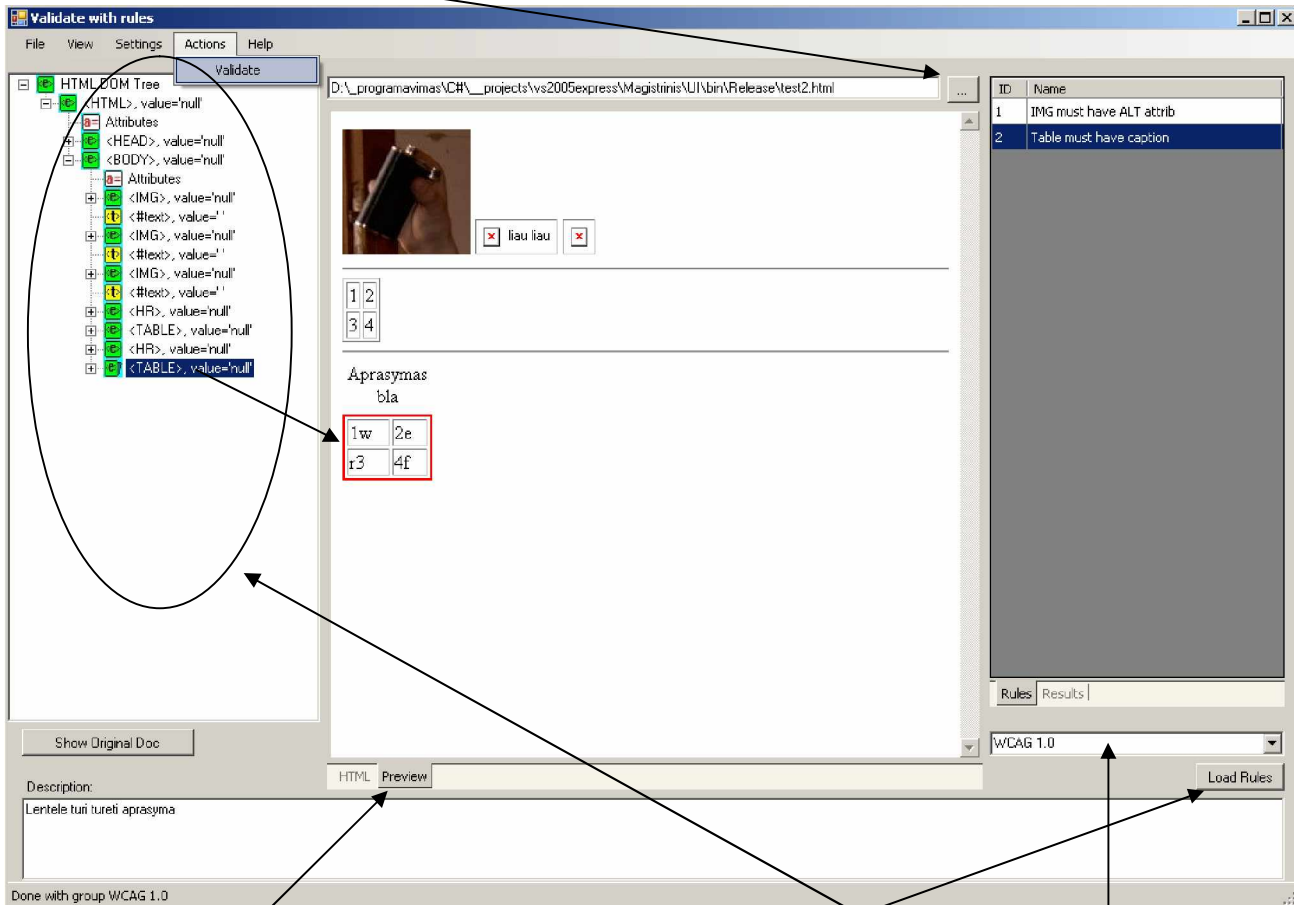
 Lentelė *ConditionsToFixes*

Saugomas ryšys tarp sąlygų ir pataisymų.

Stulpelis	Duomenų tipas	Aprašymas
ConditionID	int	Sąlygos ID.
FixID	int	Pataisymo ID.

11.7. Vartotojo vadovas

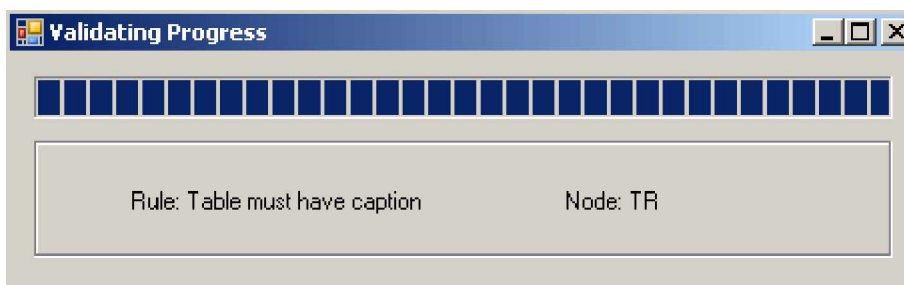
Paleidžiame programą ir, paspaudę mygtuką [...], viršuje pasirenkame dokumentą, kurį vertinsime



Dokumentas atidaromas, ir kairėje pusėje suformuojamas objektų medis. Viduryje parodomas pats dokumentas. Paspaudus ant objekto, medyje objektas pažymimas dokumente. Išsirinkę taisyklių grupę, spaudžiame mygtuką Load Rules. Užkraunamas taisyklių sąrašas.

Pasirinkę dokumentą ir taisyklių grupę, pasirenkame meniu: Action->Validate.

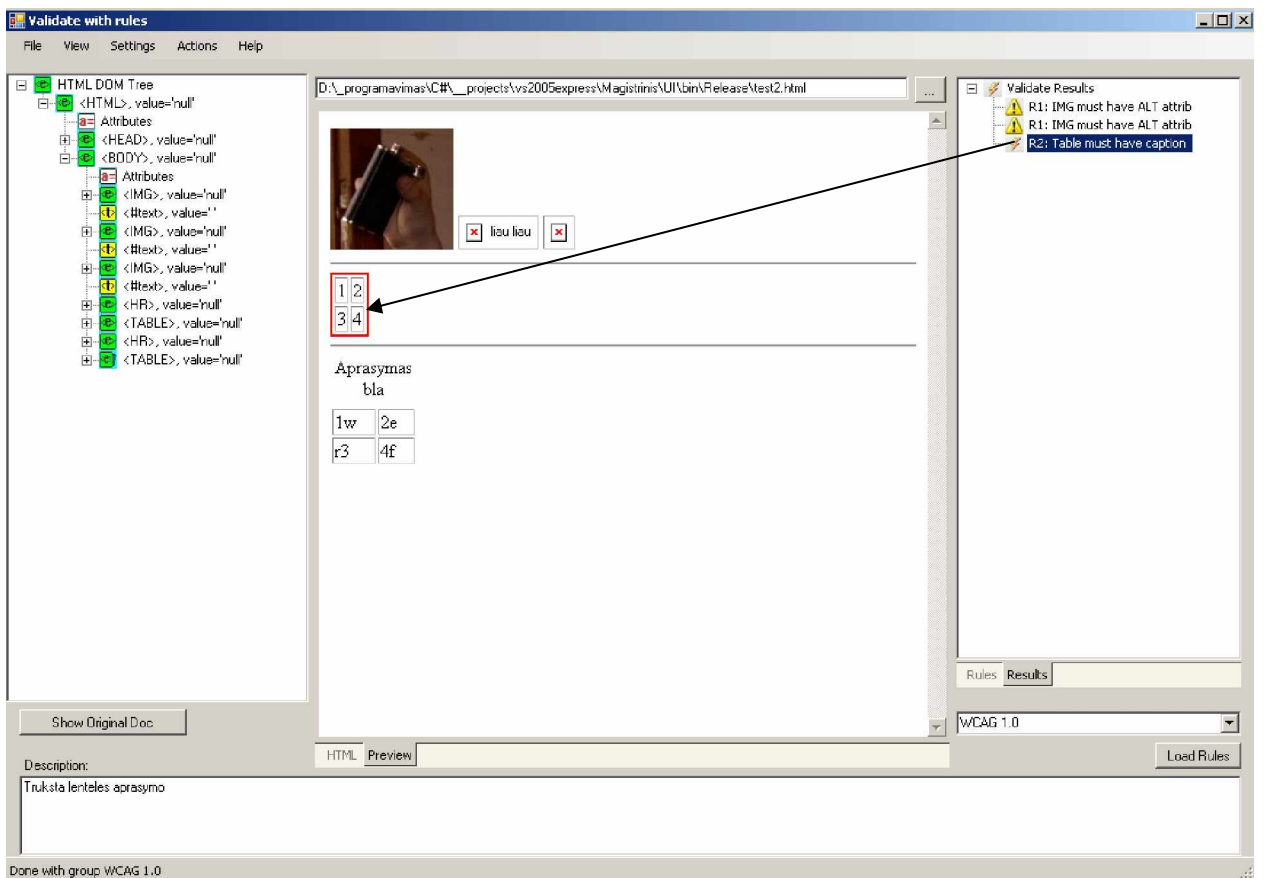
Prasideda dokumento patikra



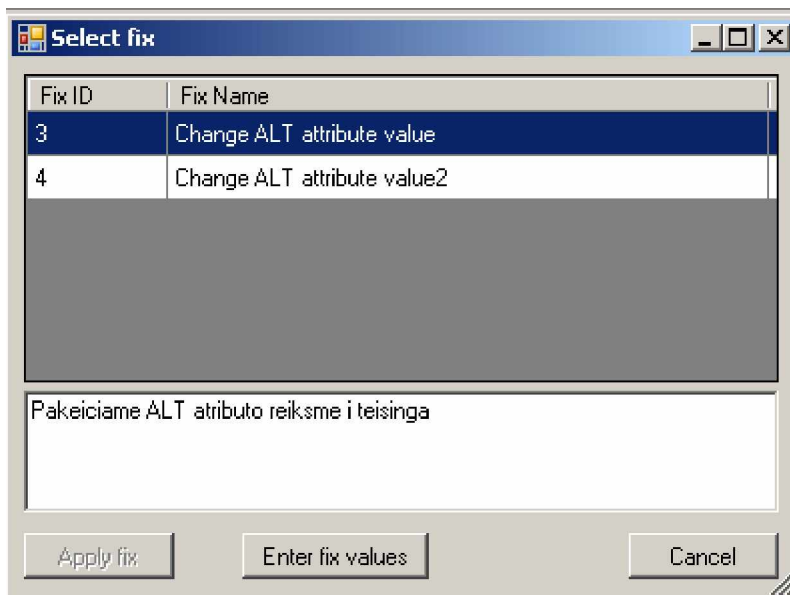
Po patikros parodomi rezultatai:



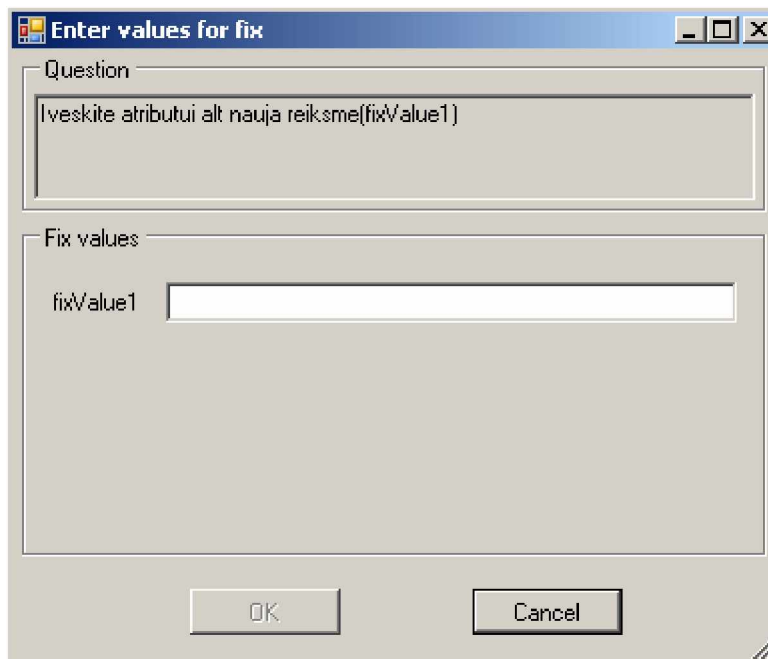
Ekrano kairėje parodomos visos rastos problemos. Paspaudus ant problemos, dokumente pažymimas elementas, kuriame rasta problema.



Paspaudus dešinį klavišą ant problemos ir pasirinkus fix, parodomi visi galimi pataisymai rasti problemai.



Pasirenkame vieną iš siūlomų sprendimų ir spaudžiame Enter fix values tam, kad įvestume papildomas reikšmes, reikalingas pataisymui.



Įvedę spaudžiame OK, paskui Apply. Pasirinktas pataisymas pritaikomas.

Pataisymą galima iš karto pamatyti dokumente.

12. Analizė ir taisyklių sudarymo principai

12.1. Taisyklių sudarymo principai ir apribojimai:

- Taisyklė gali priklausyti vienai arba kelioms taisyklių grupėms;
- Galima jau iš turimų taisyklių sudaryti tam tikrą grupę. Taip galime sudaryti įvertinimo taisyklių rinkinius įvairioms situacijoms;
- Kiekviena taisyklė turi vieną arba keletą sąlygų. Taisyklės nesilaikoma, jei nepritaikoma nors viena iš sąlygų;
- Sąlyga gali priklausyti vienai arba keletai taisyklių;
- Sąlygą sudaro objektas, operacija ir reikmė naudojama operacijoje su objektu. Operacija gali turėti klausimą, jeigu vienareikšmiškai nėra aišku, ar sąlyga tinkama. Jis užduodamas vartotojui, kuris ir nusprendžia, ar sąlyga pritaikyta teisingai;
- Objektai, dalyvaujantys sąlygoje, gali būti **HTML žymės** ir **atributai**. Jei tinka bet koks elementas, vartojamas žodis **ANY**;
- Kiekviena sąlyga gali turėti vieną ar kelis pataisymus sąlygos pagalba rastai problemai;
- Pataisymas susideda iš pataisymo operacijos, fiksuotų reikšmių (naudojamų pataisyje) ir reikšmių skaičiaus, reikalingo gauti iš vartotojo. Jis taip pat saugo pranešimą, nurodantį vartotojui, kokias reikšmes jam reikia įvesti.

12.2. Taisyklių grupės

Kadangi įrankis buvo kuriamas kaip universali hipertekstu pateiktos medžiagos testavimo priemonė, jis nėra susietas nei su viena iš taisyklių grupių (kaip kad dauguma nagrinėtų įrankių). Taisyklės galima bet kada papildyti, pakeisti arba pritaikyti savo reikmėms. Taip pat realizuoti naujus standartus ar sprendimus. Programos keisti keičiant taisyklės nereikia.

- Pagrindinės numatytos taisyklių grupės:

Grupė	Aprašymas
WCAG 1.0	W3C standartų 1 versija hiperteksto

	dokumentams pritaikyti žmonėms su negalia.
WCAG 2.0	W3C standartų 2 versija hiperteksto dokumentams pritaikyti žmonėms su negalia.
Section 508	JAV vyriausybės nustatyti standartai hiperteksto puslapiams pritaikyti žmonėms su negalia.

- Papildomos grupės:

Grupė	Aprašymas
No advertisements	Taisyklių rinkinys, pašalinantis internetines šiukšles (spam) iš dokumento.
For WML	Taisyklių rinkinys, skirtas pritaikyti dokumentą mobiliems telefonams.
Text only	Taisyklių rinkinys, skirtas pritaikyti dokumentą tekstinėms interneto naršyklėms.

Įrankio vartotojas bet kada gali susikurti savo taisyklių grupę ir į ją itraukti savo sukurtas arba jau egzistuojančias taisykles.

12.3. Taisyklių sąlygos ir pataisymai

Ü Galimos sąlygos

Operacija	Aprašymas
hasChild	Ar objektas turi vaikinį objektą?
countless	Tam tikrų HTML žymių / atributų skaičius ir t.t.
sizeless	HTML žymės / atributo reikšmės dydis mažesnis nei nurodyta parametre.
tagHasAttrib	Ar HTML žymė turi atributą?
hasSubString	HTML žymė / atributas savo reikšmėje turi šią teksto dalį.

sizeMore	HTML žymės / atributo reikšmės dydis didesnis nei nurodyta parametre.
sizeEq	HTML žymės / atributo reikšmės dydis lygus nurodytam parametre.
countMore	Tam tikrų HTML žymių/ atributų skaičius ir t.t.
countEq	Tam tikrų HTML žymių/ atributų skaičius ir t.t.
TagHasChildWithAttr	HTML žymė turi vidinę žymę su atributu, nurodytu parametre.
TagHasChildInFullDeep	Vidinės HTML žymės ieškoti visu gyliu.
NotHasSubString	Negyvena eilutės dalis.
NextTagIs	Ar kita HTML žymė yra tokia, kaip nurodoma parametre?

Ü Galimi pataisymai

Operacija	Aprašymas
removeTag	Pridėti objektą.
removeTagWithChilds	Pašalinti objektą su visais vaikiniais elementais.
removeAttribute	Pašalinti objekto atributą.
addAttribute	Pridėti atributą.
addChildTag	Pridėti vaikinį objektą.
suggestion	Tik patarimas, jokių konkrečių pataisymų.
changeAttributeVal	Pakeisti objekto atributo reikšmę.

12.4. Taisyklių pavyzdžiai

Aprašymas

Taisyklė, skirta nustatyti, ar kiekvienas paveikslėlis dokumente turi suprantamą alternatyvų tekstą.

Taisyklė:

ID	Name	Type	Group	Priority	Description
1	IMG must have ALT attrib	Error	WCAG 1.0	1	Image must contain alternative text

Sąlygos:

ID	tag	attribute	Operation	value	IfFalse	needQuestion	Question
2	IMG	ALT	tagHasAttrib	ANY	Trūksta ALT atributo, jo p...	0	NULL
3	IMG	ALT	sizeless	4	ALT atributo reikšmė trump...	0	NULL

Tikrinamos 2 sąlygos: ar iš viso toks tekstas yra ir ar jis ilgesnis nei 3 simboliai.

Pataisymai:

ID	Name	Description	value	value1	value2	question	fixValues
1	Add ALT attri...	Pridedame AL...	addAttribute	ALT	NULL	Iveskite atributui al...	1

Pataisymas 2 sąlygai – pridedamas alternatyvaus teksto atributas. Jo reikšmę prašoma įvesti vartotojo.

12.5. Taisyklių įvedimo įrankis

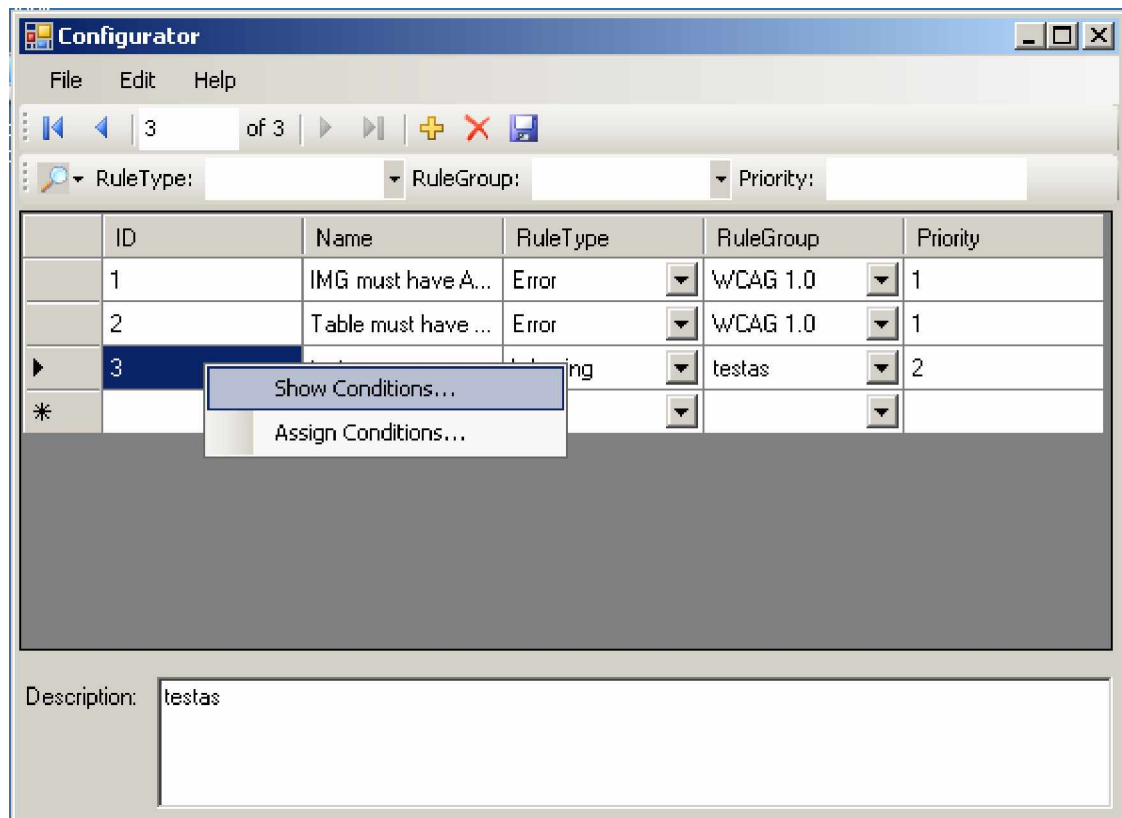
Taisyklių įvedimo įrankis yra skirtas įvesti formalioms taisyklėms, kurių pagalba analizavimo įrankis analizuoja hiperteksto dokumentus. Taisyklės įvedamos į duomenų bazę. Iš jos vėliau galima (jei reikia) eksportuoti į XML formatą.

Įrankio funkcijos:

- Ø Įvesti taisyklių grupes
- Ø Įvesti taisykles ir jų sąlygas
- Ø Grupuoti taisykles į grupes
- Ø Įvesti korekcijas
- Ø Koreguoti taisykles
- Ø Pateikti paieška tarp įvestų taisyklių

Įrankio langai:

- Ø Pagrindinis langas
- Ø Sąlygų įvedimo, koregavimo ir priskirimo taisyklėms langas
- Ø Korekcijų įvedimo, koregavimo ir priskirimo taisyklėms langas
- Ø Taisyklių tipo ir grupės įvedimo langai
- Ø Korekcijų ir sąlygų operacijų įvedimo langai



Pav. Pagrindinis langas

Pagrindinis taisyklių įvedimo programos langas susideda iš 4 dalių:

- Ø Menių, kuriame galima pasirinkti norimas operacijas
- Ø Filtravimo juostos
- Ø Taisyklių sąrašo
- Ø Taisyklės aprašymo

Taisyklių grupių įvedimas

Pagrindiniame lange spaudžiame *Edit* pasirenkame *Rule Groups*. Įvedame Taisyklių grupę ir spaudžiame *Save* mygtuką.

Taisyklių tipų įvedimas

Pagrindiniame lange spaudžiame *Edit* pasirenkame *Rule Types*. Įvedame Taisyklių tipą ir spaudžiame *Save* mygtuką.

Patikros operacijų įvedimas

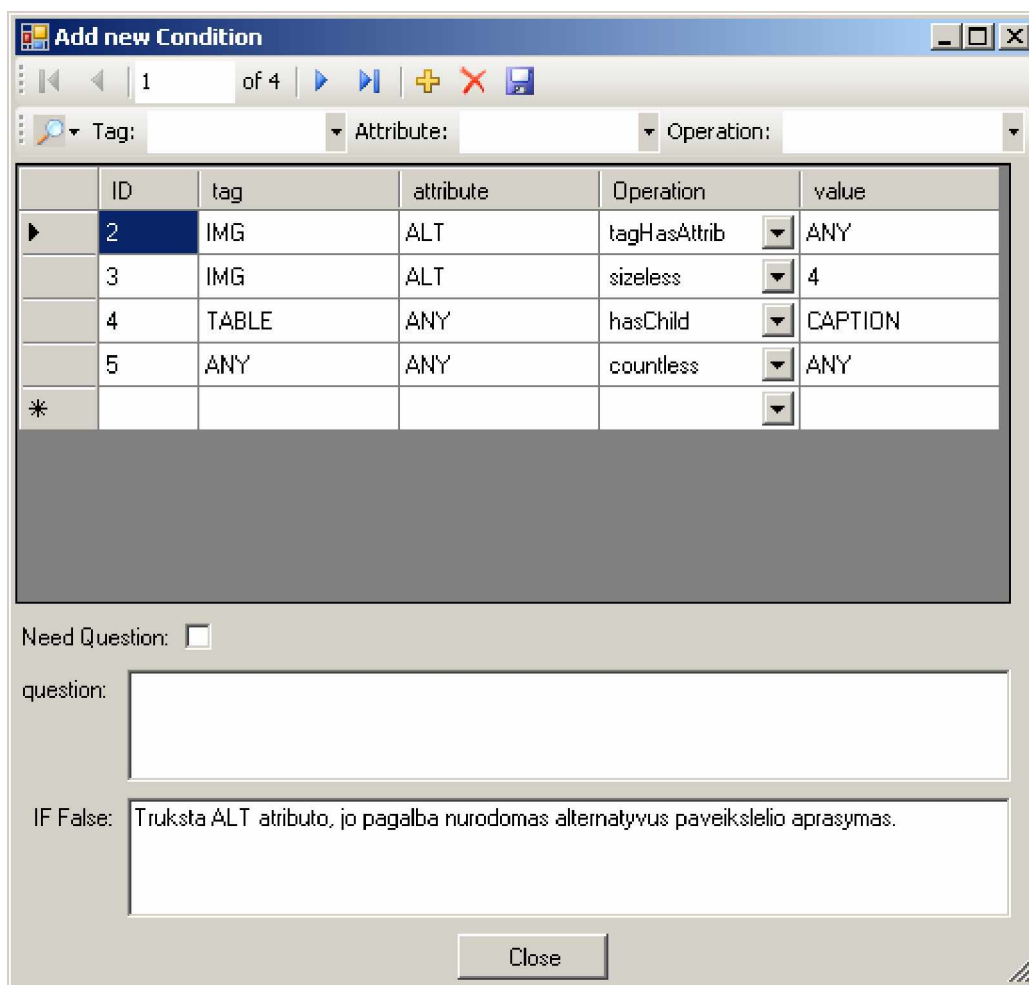
Pagrindiniame lange spaudžiame *Edit* pasirenkame *Operations*. Įvedame patikros operacijos pavadinimą. Šis pavadinimas turi atitikti funkciją realizuota įrankyje.

Korekcijos operacijų įvedimas

Pagrindiniame lange spaudžiame *Edit* pasirenkame *Fix Operations*. Įvedame korekcijos operacijos pavadinimą. Šis pavadinimas turi atitikti funkciją realizuota įrankyje.

Taisyklės įvedimas

Pagrindiniame lange spaudžiame mygtuką *New* ir įvedame bei pasirenkame norimas laukų reikšmes. Viską įvedę spaudžiame *Save* mygtuką.

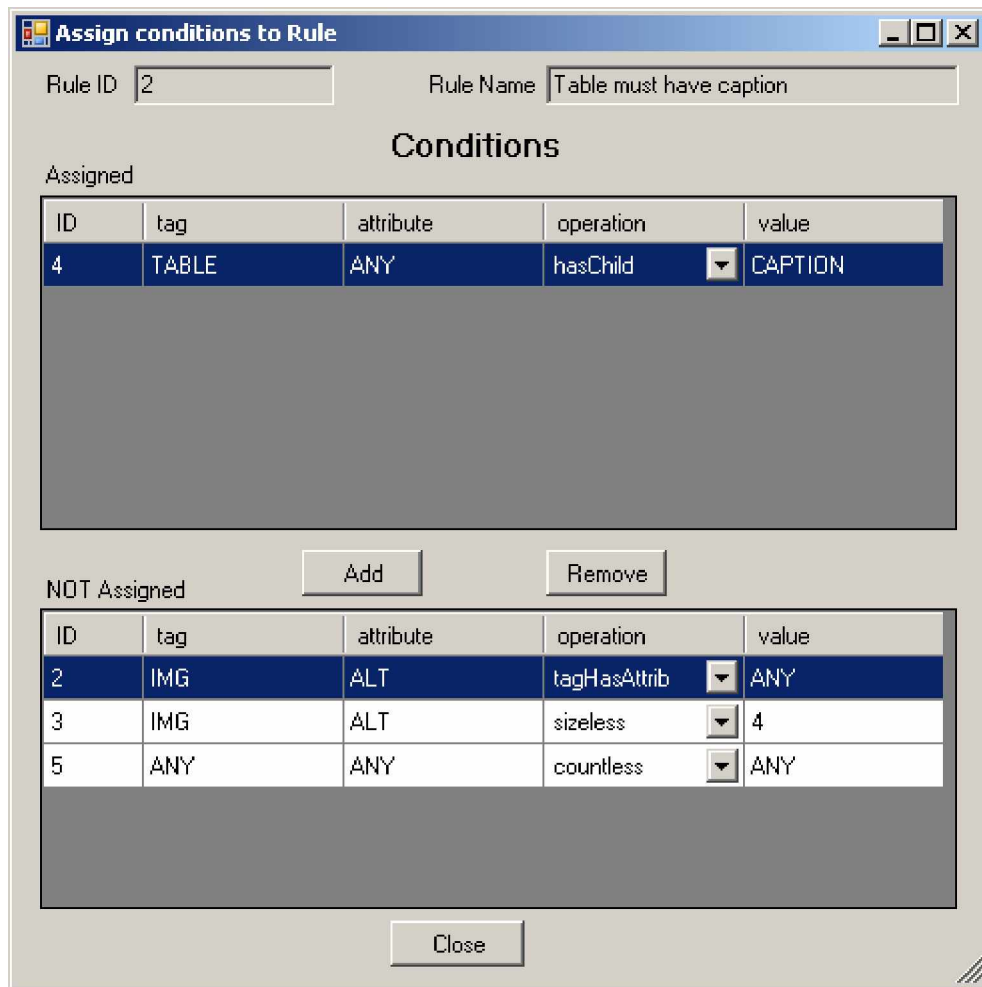


Pav. Sąlygų įvedimas

Sąlygų įvedimas ir priskyrimas

Pagrindiniame lange spaudžiame *Edit* pasirenkame *Conditions*. Atsidaro sąlygų įvedimo langas. Spaudžiame *New* ir įvedame reikšmes. Uždarome langą.

Norėdami priskirti sąlygą taisyklei pagrindiniame lange pažymime taisyklę ir spaudžiame dešinį klavišą. Atsiradusiame meniu pasirenkame *Assign conditions*. Atsidaro sąlygų pasirinkimo langas.



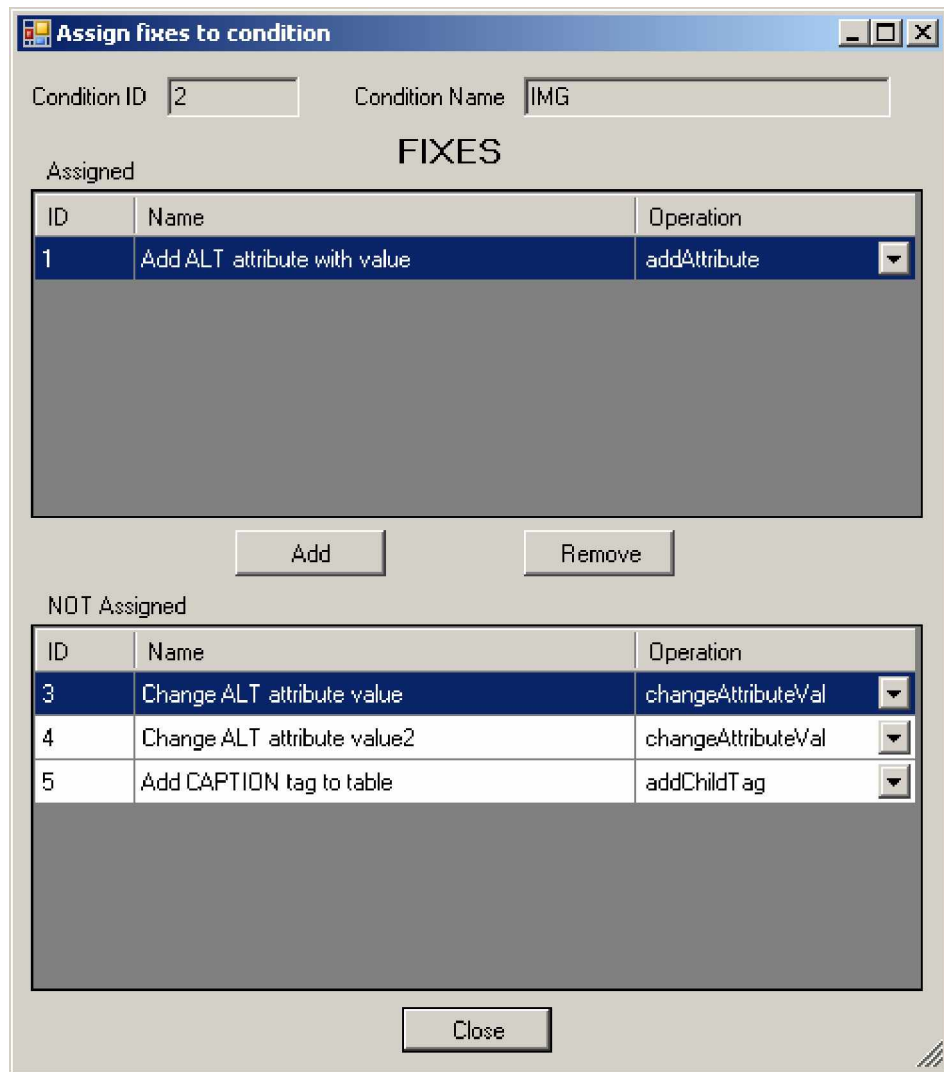
Pav. Sąlygų priskyrimas

Pažymime norimas priskirti sąlygas ir spaudžiame *Add*.

Korekcijų įvedimas ir priskyrimas

Pagrindiniame lange spaudžiame *Edit* pasirenkame *Fixes*. Atsidaro korekcijų įvedimo langas. Spaudžiame *New* ir įvedame reikšmes. Uždarome langą.

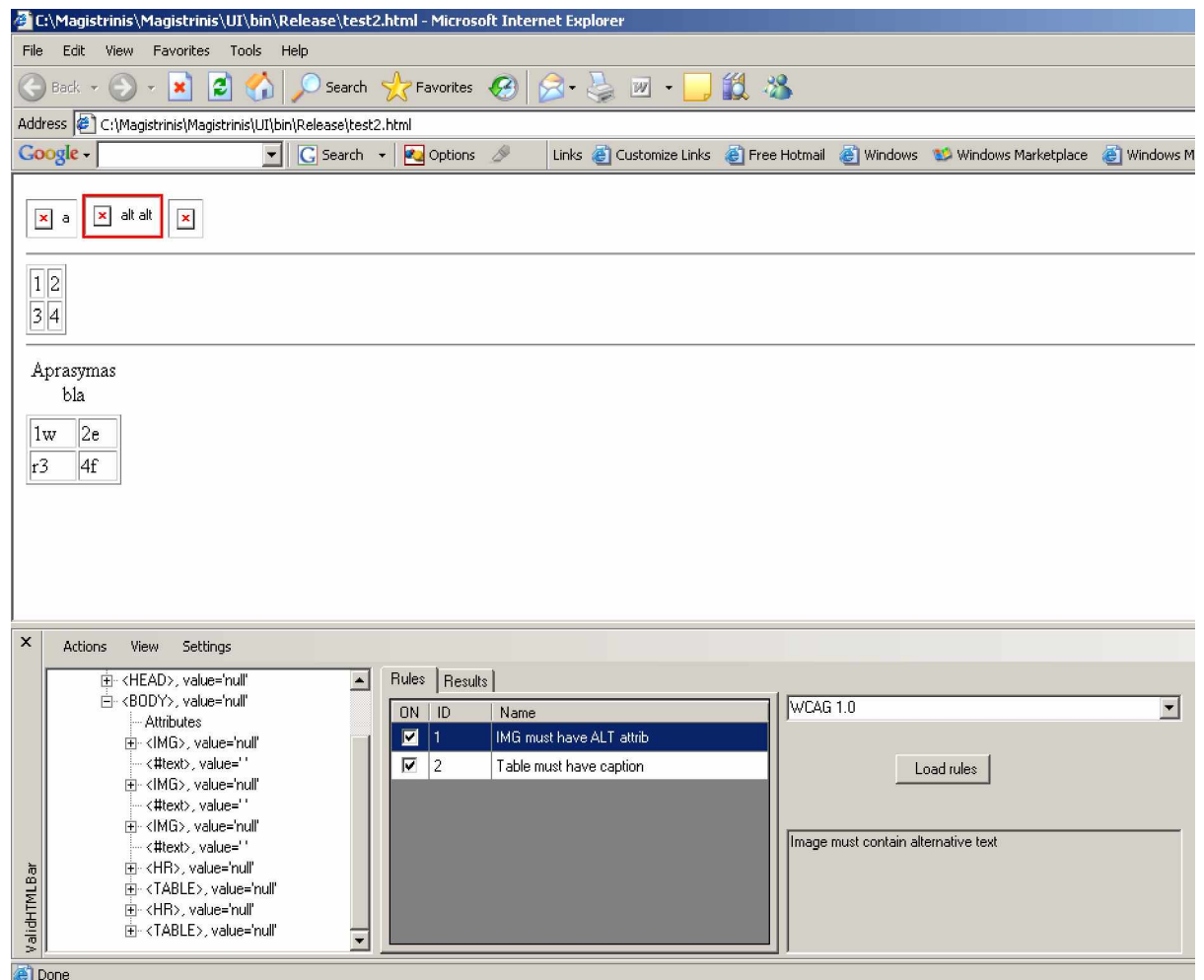
Norėdami priskirti korekciją sąlygai pagrindiniame lange pažymime sąlygą ir spaudžiame dešinį klavišą. Atsiradusiame meniu pasirenkame *Assign fixes*. Atsidaro korekcijų pasirinkimo langas.



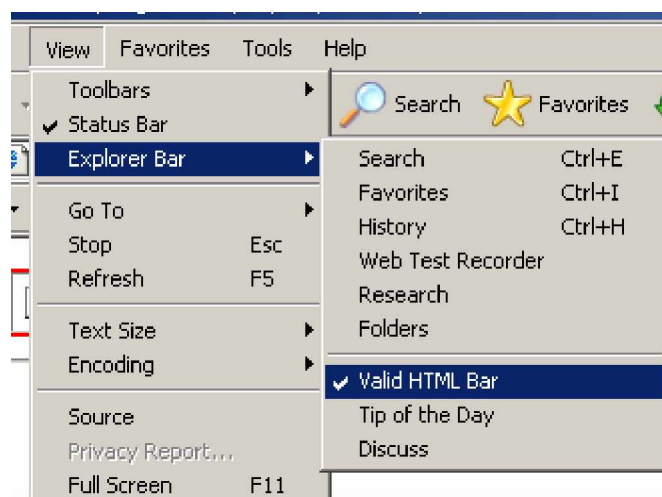
Pav. korekcijų priskyrimas

Pažymime norimas priskirti korekcijas ir spaudžiame *Add*.

Įrankio integracija į Internet Explorer programą



Programos funkcionalumas analogiškas aukščiau aprašytam. Analizuojamas tuo metu aktyvus HTML dokumentas. Programos langas išskviečiamas:



Išvados ir rezultatai

Magistrinio darbo tikslas buvo išanalizuoti nuotoliniu būdu studijuojančių žmonių su negalia problemas, mokantis iš hipertekstu pateiktos mokymosi medžiagos ir pateikti sprendimą, leidžiantį palengvinti ruošiamos medžiagos pritaikymą neįgaliesiems. Kadangi nuotoliniame mokyme dauguma medžiagos pateikiama hipertekstu, o neįgaliesiems toks mokymosi būdas vienas priimtinausių ir patogiausių, tai ši problema yra labai aktuali. Analizės metu buvo išanalizuotos priemonės naudojamos neįgaliųjų dirbant su hipertekstu pateikta medžiaga. Nustatyta, kad pagrindinės priemonės - tai ekrano skaitymo ir šriftą bei spalvas keičiančios programos. Išnagrinėjus hiperteksto kalbą HTML, nustatyta, kad ji turi pakankamai priemonių, kurių pagalba hipertekstu pateiktą medžiagą galima pritaikyti neįgaliesiems. Tačiau viena iš pagrindinių problemų yra tai, kad žmonės, ruošiantys tokią medžiagą, prastai arba visai neišmano HTML kalbos. Daugiausia naudojamos automatizuotos hiperteksto kūrimo priemonės, kurios dažniausia neturi funkcijų dokumento pritaikymui neįgaliesiems. O šių programų sugeneruotą hiperteksto dokumentą redaguoti labai sudėtinga. Be to, neįgaliųjų naudojamos priemonės taisyklingai nesupranta tokių hiperteksto dokumentų.

Kadangi HTML hiperteksto aprašymo kalba formaliai aprašo dokumento pateikimo struktūrą, o visi naudojami objektai yra vienodos struktūros ir standartizuoti, galima sudaryti taisyklių rinkinį, kurio pagalba būtų nesudėtinga technologiškai įvertinti hiperteksto dokumentą. Tačiau HTML kalba aprašo tik dokumento pateikimą neliesdama turinio, todėl ne visas analizės metu suformuluotas taisyklės įmanoma formaliai aprašyti. Taigi analizės metu prieita išvados, kad reikia įrankio, kurio pagalba būtų galima jau sukurtą dokumentą pritaikyti neįgaliesiems. Pagrindiniai šio įrankio kriterijai: 1) paprastumas naudoti – nereikalingos HTML programavimo žinios, galimybė plėsti be įrankio kodo perrašymo; 2) lankstus konfigūravimas ir 3) WYSIWYG darbo principas. Kadangi nei vienas iš nagrinėtų sprendimų neatitiko visų reikalavimų, buvo nuspręsta tokį įrankį sukurti. Taigi darbo rezultatas - analizės metu sudarytas formalus taisyklių rinkinys, nurodantis kriterijus, kuriuos turi atitikti hiperteksto dokumentas ir įrankis – programa, leidžianti paprastai ir efektyviai pritaikyti hiperteksto dokumentus neįgaliesiems. Įrankį galima integruoti į interneto naršyklę. Įrankio taisyklių sąrašas bet kada gali būti papildomas ar sukuriamos naujos taisyklių grupės, leidžiančios jį pritaikyti ne tik hipertekstinių dokumentų pritaikymui neįgaliesiems, bet ir kitoms reikmėms, pvz.: hiperteksto dokumento pritaikymas mobiliųjų įrenginių vartotojams, dokumento išvalymas nuo tam tikrų dalių (reklamos, interneto šiukšlių ir kt.).

Santrumpų sąrašas

HTML (HyperText Markup Language) – hiperteksto kalba

SQL (Structured Query Language) – struktūrizuota užklausų kalba

NM – nuotolinis mokymas

UML – unifikuota modeliavimo kalba

OS – operacinė sistema

DB – duomenų bazė

HTTP (HyperText Transfer Protocol) – hipertekstinių dokumentų perdavimo protokolas

DTD (Data Types Definition) - duomenų tipus aprašanti rinkmena

WYSIWYG (What You See Is What You Get) – „ką matai, tą ir gauni“

IT – informacinės technologijos

DOM (Dynamic Object Model) – dinaminis objekto modelis

DBVS – duomenų bazių valdymo sistema.

MSDE (Microsoft SQL Desktop Edition) – duomenų bazių serverio versija

UI (User Interface) – vartotojo sąsaja

UIC (User Interface Controler) – vartotojo sąsajos valdiklis

DALC (Data Access Layer Component) – duomenų lygio valdiklis

WML (Wireless Markup Language) – mobilių svetainių metakalba

Literatūros sąrašas

Probleminės literatūros sąrašė daugiausia naudotasi kitų universitetų patirtimi kuriant ir pritaikant kursus neįgaliems žmonėms. Tai pat nemažas dėmesys skirtas pagrindinėms operacinių sistemų gamintojų siūlomoms priemonėms, palengvinančiomis neįgalių žmonių darbą. Daugumą šaltinių rasta naudojantis paieškos svetainėmis: www.google.lt, www.search.lt.

1. Akių ligos: Vadovėlis aukštųjų mokyklų studentams / E. Daktaravičienė, G. Juodkaitė, K. Sukarevičius. Vilnius: Mokslas, 1992
2. Darbas neįgaliems. Prieiga per internetą:
<http://www.darborinka.lt/neigaliesiems/?pid=42>
3. Garbus J., Pascuzzi D., Chang A. Database Design on SQL Server 7. Москва, 2000
4. Gunnerson E. A Programmer's Introduction to C#. New York, 2000
5. Nuotolinis mokymasis. Mokomoji knyga / D. Rutkauskienė ir kt. Kaunas: Technologija, 2003.
6. PHARE projektas. Prieiga per internetą:
http://www.vtvk.lt/PHARE_Pr/PHARE2001.htm
7. Slatin J. M., Rush Sh. Maximum Accessibility: Making Your Web Site More Usable for Everyone. Boston, 2003
8. The Access Board. Prieiga per internetą: www.access_board.gov
9. The ITTATC—Information Technology Technical Assistance and Training Center. Prieiga per internetą: www.ittatc.org
10. University of Maryland University College. Prieiga per internetą:
<http://www.umuc.edu/ade/>
11. W3C--World Wide Web Consortium. Prieiga per internetą: www.w3c.org
12. WAI WCAG--Web Accessibility Initiative Web Content Accessibilities Guidelines. Prieiga per internetą: <http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505/>
13. Велихов С. Справочник по HTML 4. Москва, 2001
14. Дубовцев А. Microsoft .NET в подлиннике. Санкт-петербург, 2004
15. Шмullер Д. Освой самостоятельно UML за 24 часа. Москва, 2002

Priedai

Kaip priedas pateikiamas kompaktinis diskas su sukurtu hiperteksto analizės įrankiu.

Kompaktinio disko turinys:

- Ø ValidHTMLDoc – įrankis realizuotas kaip atskira programa.
- Ø ValidHTMLDoc for IE - įrankis realizuotas kaip IE įskiepis
- Ø Configurator – taisyklių įvedimo programa
- Ø DB – duomenų bazės sukurimo skriptai

Kiekviename kataloge yra rinkmena *Readme.txt* kurioje yra diegimo instrukcija.