

**KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
INFORMACIJOS SISTEMŲ KATEDRA**

Andrėjus Bereščanskis

**PERIODINIO LEIDINIO INFORMACIJOS SURINKIMO IR  
VALDYMO SISTEMA**

Magistro darbas

**Darbo Vadovė:** docentė  
*Rita Butkienė*

**KAUNAS 2005**

# TURINYS

<b><u>1. IVADAS</u></b>	<b>3</b>
<b><u>2. ANALIZĖS DALIS</u></b>	<b>4</b>
<u>2.1. Tyrimo sritis, objektas, problema, darbo aktualumas</u>	4
<u>2.2 Reikalavimų informacijos surinkimo sistemai analizė</u>	7
<u>2.2.1 Organizacijos informacijos surinkimo padalinio veiklos analizė</u>	7
<u>2.2.2 Veiklos objektų modelis</u>	8
<u>2.2.3 Veiklos tikslų modelis</u>	9
<u>2.2.4 Veiklos panaudojimo atvejų modelis</u>	10
<u>2.2.5. Veiklos procesų modelis</u>	12
<u>2.2.6 Vartotojų ir jų poreikių analizė</u>	13
<u>2.2.7 Nefunkciniai reikalavimai sistemai</u>	14
<u>2.2.8 Informacijos surinkimo sistemos klasių diagrama</u>	15
<u>2.3 Literatūros šaltiniuose pateiktų sprendimų problemai spresti lyginamoji analizė</u>	17
<u>2.3.1 Panašių sistemų analizė</u>	17
<u>2.3.2 Duomenų talpinimo į duomenų bazės technologijų paliginimas</u>	17
<u>2.4. Projekto tikslas ir jo pagrindimas, kokybės kriterijų apibrėžimas</u>	19
<u>2.5 Kompiuterizuojamos sistemos varianto parinkimas</u>	22
<u>2.6 Analizės išvados</u>	24
<b><u>3. PROJEKTO DALIS</u></b>	<b>26</b>
<u>3.1 Techninė užduotis</u>	28
<u>3.2 Reikalavimų modelis</u>	29
<u>3.2.1 Vartotojų panaudojimo atvejų modelis</u>	29
<u>3.2.2 Vartotojo grafinės sąsajos modelis</u>	34
<u>3.3 Sistemos projektas</u>	38
<u>3.3.1 Realizacijos modelis</u>	38
<u>3.3.2 Duomenų bazės modelis</u>	41
<u>3.3.3 Sistemos elgsenos modelis</u>	44
<u>3.3.4 Sistemos naudojimo instrukcija</u>	46
<u>3.3.5 Informacinės Sistemos diegimo planas</u>	46
<u>3.3.6 Reikalavimai sistemos funkcionalumui</u>	48
<b><u>4. EKSPERIMENTINIS TYRIMAS</u></b>	<b>50</b>
<u>4.1 Sukurtos sistemos kokybės tyrimas</u>	50
<u>4.2 Tolimesnio sistemos tobulinimo, plėtojimo galimybės</u>	53
<b><u>5. IŠVADOS</u></b>	<b>54</b>
<b><u>6. LITERATŪROS SARAŠAS</u></b>	<b>56</b>
<b><u>7. TERMINŲ IR SANTRUMPŲ ŽODYNAS</u></b>	<b>57</b>

<b><u>8. SANTRAUKA ANGLŲ KALBA</u></b>	<b>58</b>
<u>Summary</u>	58
<b><u>9. PRIEDAI</u></b>	<b>59</b>

## 1. Įvadas

Informacinių technologijų plėtra pradėjo įtakoti vis daugiau verslo sričių. Pastoviai besiplečiančiame verslo pasaulyje labai svarbu yra gauti ir pateikti informaciją kuo greičiau ir patogiau vartotojui. Greitis ir patikimumas, greita reakcija į besikeičiančią aplinką – tai yra vieni iš pagrindinių pelningo ir stabilaus verslo reikalavimų. Augant vartotojų poreikiams, didėjant informacijos apdorojimo poreikiui dinamiškai kintančioje aplinkoje, tradiciniai medžiagos apdorojimo metodai jau nebėra efektyvūs. Todėl šiuo metu labai daug dėmesio skiriama verslo plėtimuisi ir jo perkėlimui į web technologijų terpę, siekiant sukurti vieningą sistemą. Verslas susiduria su visiškai nauja problema – informacinių srautų valdymu. Vis dažniau pastebima tendencija kelti verslą į elektroninę erdvę ir kurti taip vadinamą e-verslą. Informacijos srautai laikui bėgant pastoviai didėja, ir tai priverčia naudoti kuo greitesnes, patogesnes ir lankstesnes informacines sistemas. Į pagalbą pastoviai pasitelkiamos pažangiausios IT technologijos ir sprendimai. Žmogaus galimybės valdyti ir rinkti informaciją yra ribotos. Jas praplečia ir papildo internetas. Naudojant globalius tinklus išsprendžia problemas su informacijos pasiekiamumu. Duomenys tampa pasiekiami bet kada ir bet kur.

Pastaruosiu metu vis daugiau IS uždavinių yra sprendžiami naudojant Interneto technologijas. Labai svarbus uždavinys yra greitai persiorientuoti prie naudojimosi globaliais tinklais, pavyzdžiui Internetu. Modeliuojant leidinio formavimo sistemą, kuri veiktų Interneto terpėje, būtina atkreipti dėmesį į tai, kad taisyklės, pagal kurias sudaromi leidiniai, paprastai keičiasi greičiau, negu probleminės srities duomenų struktūros ar objektų elgsena. Dažnai reikia leidinio sudarymo taisyklių rinkinį papildyti, peržiūrėti, keisti. Leidinio turinys yra nemažiau svarbus negu jo sudarymo taisyklės. Bet šiame darbe bus atsižvelgiama į tai, kaip padaryti kuo efektyvesnę priemonę, kurios pagalba, naudojant internetą, bus galima realizuoti duomenų užkrovimą ir šios sistemos administravimą.

## 2. Analizės dalis

### 2.1. Tyrimo sritis, objektas, problema, darbo aktualumas

Šio darbo tyrimo objektas – informacijos surinkimas iš autorių straipsnių Interneto terpėje bei vartotojų administravimas. Taip pat, panaudojant internetą, interaktyvaus bendravimo tarp leidinio kūrėjo ir straipsnių autoriaus uždavinio sprendimas.

Pagrindinė tyrimo sritis – informacijos ir dokumentų surinkimo sistema.

Visą šiame darbe nagrinėjamą tyrimo sritį galima išskirti į kelias atskiras sritis (lentelė 1).

Sritis	Srities aprašymas
Leidyba užsiimančios organizacijos veikla ir poreikiai	<ul style="list-style-type: none"><li>• Kas yra leidyba užsiimanti organizacija ir kas būdinga jos veiklai.</li><li>• Organizacijos duomenų surinkimo srities poreikių charakteristika, jų įgyvendinimo metodai</li></ul>
Informacijos ir resursų valdymas organizacijoje	<ul style="list-style-type: none"><li>• Kas yra organizacijos informacija ir resursai</li><li>• Kokie informacijos srautai duomenų surinkimo srities viduje</li><li>• Informacijos srautų valdymo būdai</li></ul>
Procesų automatizavimas	<ul style="list-style-type: none"><li>• Kokį žmonių atliekamą darbą ir procesus virtualioje organizacijoje galima automatizuoti</li><li>• Žmogiškojo faktoriaus įtakos mažinimas</li></ul>
Saugumas	<ul style="list-style-type: none"><li>• Informacijos surinkimo saugumo politika</li><li>• Technologijos ir metodai saugumui užtikrinti</li></ul>

Sistema neturi leisti vartotojams neautorizuotai atlikti operacijas su dokumentais. Sistema neturi leisti neautorizuotiems vartotojams matyti dokumentus.

Organizacijos informacijos surinkimo sistemoje visos šios sritys yra susijusios. Analizuojant ir kuriant organizacijos informacijos surinkimo sistemą, negalima palikti nenagrinėjamą nei vieną iš šių sričių. Tai reikalinga organizacijos duomenų surinkimo sistemos išbaigtumui užtikrinti.

Darbo tikslas - ištyrus periodinio leidinio sudarymo veiklos taisykles, sukurti sistemą, kurios pagalba galima būtų rinkti medžiagą iš jos vartotojų periodiniam leidiniui sudaryti ir surinkta medžiaga būtų galima naudotis straipsnių redaktoriams. Reikia nustatyti ir išskirti tik

tas taisykles, kurias po to galime struktūriškai realizuoti duomenų bazės struktūromis ir interaktyvaus bendravimo sąsajos sudarymui bei sudaryti lengvai suprantamą sąsają vartotojams prieiti prie duomenų, taip pat sukurti aplinką duomenims užkrauti į duomenų bazes. Sukurti administravimo aplinką, kuri leis priskirti įvairias straipsnių skiltis sistemos vartotojams. Sistema turi rinkti duomenys iš autorių arba informacijos tiekėjų ir suskirstyti juos atitinkamai į duomenų bazę. Dauguma duomenų turėtų būti straipsniai. Reikia atsižvelgti į praplečiamumo galimybę ir į galimybę įdiegti sistemą, kuri leistų formuoti kompaktinio disko turinį, jei to reikalauja leidėjas, kad toliau galima būtų naudoti tą medžiagą, kuri buvo surinkta iš autorių bei platinti nemokamą programinę įrangą kartu su leidiniu. Bet dabar ši galimybė nėra apžvelgiama, galima tik sukurti jai sąsają tolimesniems programinės įrangos praplėtimams.

Darbo aktualumas – periodinio leidinio kūrėjų informacinė veikla tampa vis svarbesnė ir tokio pobūdžio darbas šiuo metų yra labai aktualus dėl paprastų priežasčių. Informacijos surinkimas yra svarbus leidinio kūrimui. Prieinamumas ir paprastumas vartotojo sąsajos yra reikšmingas darbo našumo padidinimo faktorius. Ši programa turėtų išspręsti problemas, kurios kyla su informacijos valdymu, duomenų saugojimu, duomenų surinkimu, bendravimu tarp leidėjo ir straipsnių autorių

**Padalinio, kuriam kuriama sistema modelis.**

Žemiau pateikta schema įmonės, kuri naudoja IT sprendimus informacijai iš vartotojų rinkti.

Padalinio leidinio turinio formavimo programinė įranga surenka reikiamą informaciją iš vartotojų su kuriais pasirašytos sutartys ir kuriems leista dirbti su sistema. Toliau informacija keliauja į kitus padalinius, kur ji yra apdorojama.

Kiekviena technologija turi save pateisinti t.y duoti ekonominę ar kokią kitą naudą ją naudojančiai organizacijai ar žmogui. Taip pat ir kiekvienas sprendimas - kurti ir naudoti naują vienokią ar kitokią technologiją - turi būti pagrįstas. Naujos technologijos kūrimas dažniausiai reikalauja daugiau laiko ir resursų negu esamų technologijų pritaikymas. Daugumoje kompiuterijos sričių esamų technologijų pakanka iškilusiems uždaviniams spręsti. Jei yra keletas ar keliolika technologijų, skirtų tam pačiam uždaviniui spręsti, iškyla tinkamos technologijos pasirinkimo problema. Šio darbo vienas iš tikslų – išnagrinėti ir charakterizuoti dokumentų surinkimo sistemos uždavinius ir poreikius ir parinkti tinkamiausią technologiją kiekvienam uždaviniui. Kuriant tokią sistema iškila kelios esminės problemos:

- Dokumentų saugojimas.

Kokia duomenų saugojimo technologija geriausiai tinka dokumentų surinkimo sistemai;

- Žmogiškojo faktoriaus įtakos mažinimas.

Galimybės žmogui padaryti klaidą mažinimas. Maksimalus procesų automatizavimas. Galimybės piktnaudžiauti sistema mažinimas;

- Sistemos vartojimo paprastumas.

Sistema turi „patikti“ vartotojams. Sistemos priežiūra turi būti kiek galima paprastesnė;

- Sistemos vartotojų administravimas.

Organizacija, kuri operuoja dideliais dokumentų kiekiais, turi operatyviai ir efektyviai valdyti esamus ir turimus dokumentus. Tai reiškia, kad prieigos prie dokumentų laikas turi būti kiek galima mažesnis. Taip pat reikia siekti, kad saugomų duomenų užimamas dydis duomenų saugojimo laikmenose turi būti kuo mažesnis. Saugojimo išlaidos turi būti minimizuotos. Taip pat turi būti užtikrintas duomenų saugumas. Į visus šiuos faktorius reikia atsižvelgti parenkant duomenų saugojimo technologiją.

Organizacijos tikslas – išsaugoti turimus ir naujai atsiųstus dokumentus, visada žinoti, kur ir kokie dokumentai yra saugomi, o svarbiausia – panaudoti turimus duomenis greitesniam ir efektyvesniam sprendimų priėmimui. Tam dokumentų valdymo sistemoje reikalingas paieškos mechanizmas. Norint surasti reikiamą dokumentą ar informaciją dokumentuose, reikia žinoti, apie ką yra dokumentas ir kas jame saugoma. Tekstinio pobūdžio dokumentuose reikiamo dokumento galima ieškoti pagal dokumento turinį. Jei dokumentas ne tekstinio pobūdžio, atpažinti, kas yra saugoma dokumente neįmanoma. Reikia turėti kažkokius duomenis, kurie aprašo dokumentą.

Žmogiškasis faktorius lemia tai, kad žmogus kaip sistemos vartotojas gali daryti klaidas. Jis gali įvesti klaidingą informaciją apie dokumentą, užmiršti įvesti į sistemą parametrus ir panašiai. Visus procesus, kuriuos galima tiksliai aprašyti, reikia automatizuoti. Žmogus gali ir tyčia neįvesti tam tikrų parametrų arba įvesti klaidingai. Visur, kur galima, reikia automatizuoti procesus.

Sistemos vartotojai – organizacijos darbuotojai. Tik nedidelė dirbančiųjų organizacijoje dalis turi galias žinias ir patirtį kompiuterijos srityje, todėl jiems reikalinga nesudėtinga ir jų galimybes atitinkanti sistema. Vartotojo sąsaja turi būti kuo paprastesnė sistemos vartotojams. Jei sistema sudėtinga ir apimanti daug uždavinių, iškyla problema, kaip suderinti sistemos efektyvumą ir paprastą vartotojo sąsają.



## **2.2 Reikalavimų informacijos surinkimo sistemai analizė**

### **2.2.1 Organizacijos informacijos surinkimo padalinio veiklos analizė**

Periodinių leidinių turinio sudarymas pagrįstas interaktyvaus bendravimo tarp užsakovo ir užsakymo vykdytojo interneto aplinkoje - yra nepakankamai išvystyta šaka informacijos platinimo ir surinkimo srityje . Šiuolaikinės informacinių sistemų technologijos pilnai užtikrina realizavimą tokio tipo uždavinių, naudojant naujausias programavimo kalbas ir duomenų saugojimo technologijas. Spartėjant veiklos procesams pasikeičia ir funkciniai reikalavimai esamų duomenų valdymo, surinkimo ir kontrolės sistemoms. Duomenų prieinamumas ir suderinamumas reikalauja vis naujesnių sprendimų bandant juos realizuoti. Perėjimas nuo įprasto bendravimo tarp užsakovo ir užsakymo vykdytojo prie bendravimo, naudojant interneto technologijas, žymiai paspartintų procesą, kurio metu yra apsprendžiami dalykiniai klausymai objekto atžvilgiu. Stebint nagrinėjamą sritį informacijos kaupimo požiūriu, reikia pastebėti, kad informacija išlieka ta pati, tik pateikimo būdai dinamiškai keičiasi. Naujų funkcinių reikalavimų įvedimas paprastai reikalauja papildomų duomenų struktūrų. Palikuonines duomenų bazes išsaugoti vertinga ir tuo požiūriu, kad galima palikti eksploatuoti nepakitusių uždavinius. Šiame darbe palikuoninės duomenų bazės nebus naudojamos dėl to, kad sistema bus visiškai nauja, o ne senos sistemos pratęsimas. Projektas turi atitikti dabartinius Web svetainių reikalavimus. Reikia realizuoti interaktyvų bendravimą ir mainus duomenimis tarp užsakovo ir užsakymo vykdytojo, nepaisant operacinių sistemų, kuriomis naudojasi pastarieji. Preliminarus supaprastintas panaudojimo atvejų modelis iliustruoja užsakymo pateikimą straipsniui parašyti bei užsakymo pristatymą.

Pagal organizacijos poreikius bus keliami tikslūs funkciniai reikalavimai dokumentų valdymo sistemai. Organizacijos analizei naudojama UML modeliavimo kalba. Naudojami šie UML modeliai:

- Veiklos tikslų modelis;
- Veiklos panaudojimo atvejų modelis;
- Veiklos objektų modelis;

Veiklos tikslų modelio paskirtis – detalai apibrėžti informacijos surinkimo padalinio tikslus ir poreikius dokumentų valdymui. Veiklos panaudojimo atvejų modelio paskirtis – aprašyti analizuojamos veiklos srities modelį. Jame nurodomos svarbiausios veiklos dalyvių sąveikos (materialios ir informacinės) su veiklos uždaviniais. Veiklos objektų modelis naudojamas apibrėžti detalią veiklos objektų informaciją (savybes). Veiklos objektų modelis

sudaro pagrindinį veiklos sferos objektų komponentų aprašą. Veiklos procesų modelio paskirtis – visos sistemos, kaip tarpusavyje susijusių veiklų ir funkcijų rinkinio analizė. Šis modelis geriausiai tinka analizei ir loginiam projektavimui. Jis dažniausiai atliekamas ankstyvoje projektavimo fazėje.

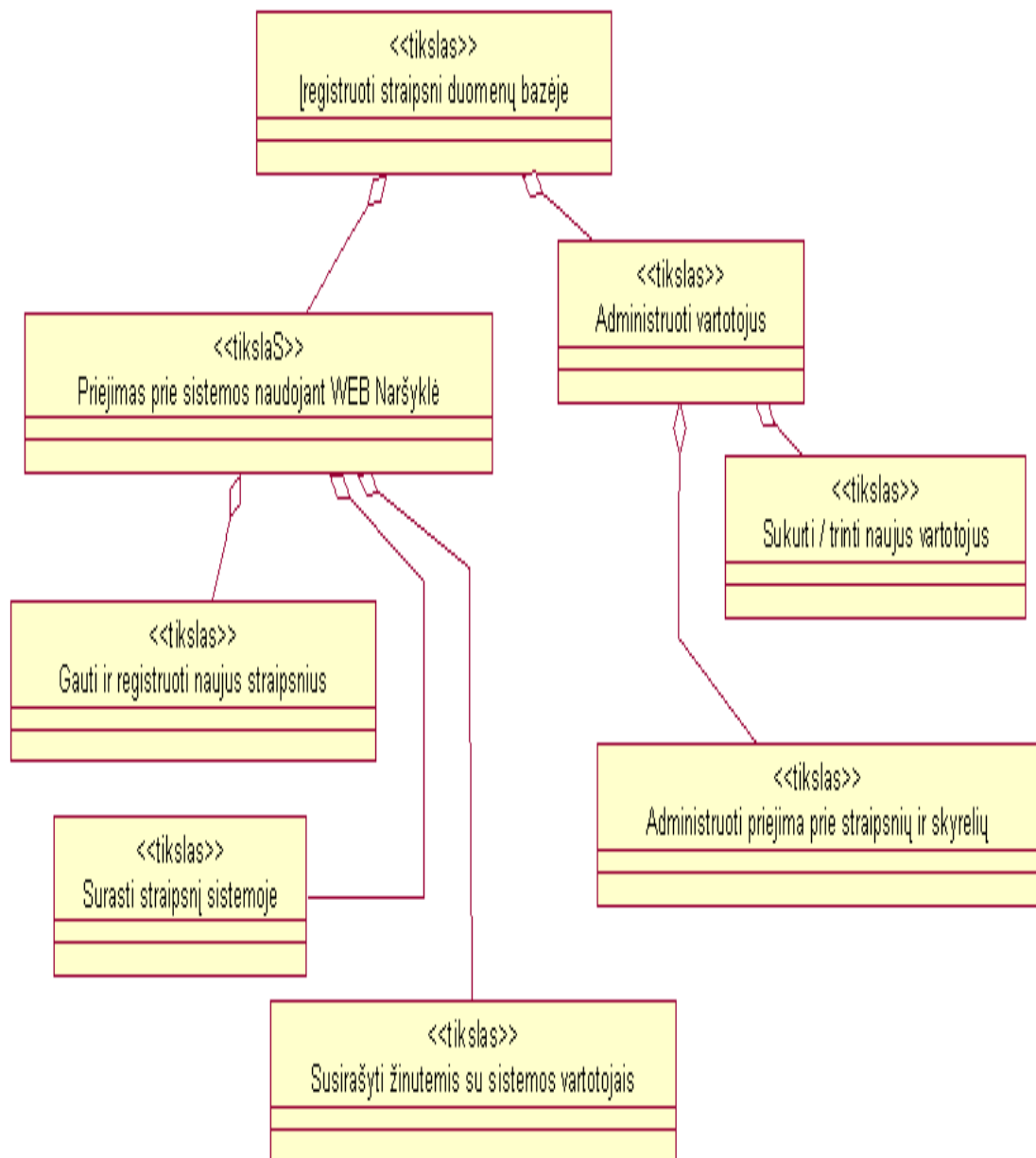
### **2.2.2 Veiklos objektų modelis**

Veiklos objektų modelis parodo kaip atliekami veiklos procesai. Tai abstrakcija, parodanti kaip veiklos darbuotojai ir veiklos esybės yra susiję ir bendradarbiauja tam, kad vykdytų veiklą

Autorius  
Sk.  
redaktorius  
Sistema  
Vartotojo programa

Pav. 2 Veiklos Objektų modelis

### **2.2.3 Veiklos tikslų modelis**

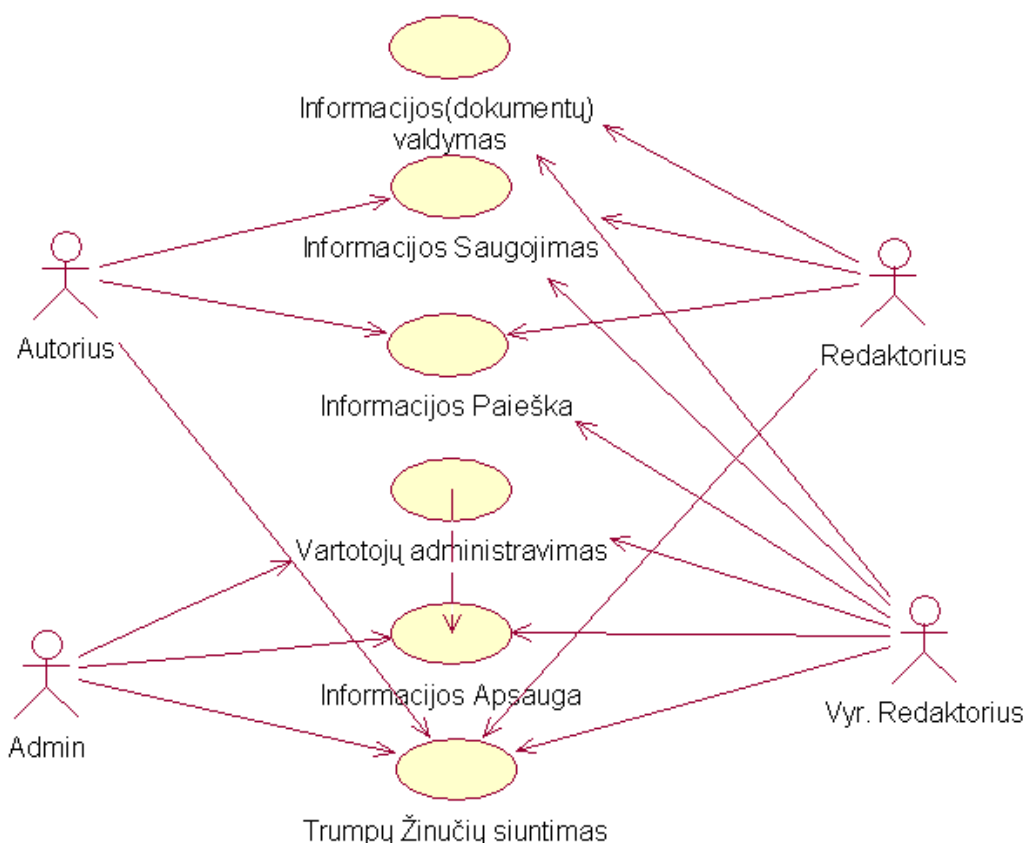


### Pav.3 Veiklos tikslų modelis

Veiklos tikslų modelyje pagrindinis tikslas yra įregistruoti straipsnių duomenų bazėje. Šiame modelyje pagrindinis veiklos tikslas yra aukščiausiam hierarchiniame lygyje, kuris yra detalizuojamas detalesniais tikslais, kurie priklauso žemesniems hierarchiniams lygiams. Taip visi tikslai vis einant į žemesnius lygius detalizuojami, kol gaunami elementarūs tikslai, kuriuos reikia pasiekti, norint pasiekti visus aukštesnius tikslus bei aukščiausio lygio tikslą. Pagrindinis tikslas yra detalizuojamas į du svarbius tikslus: priėjimas prie sistemos per web naršyklę ir vartotojų administravimas. Visa sistemos esmė - yra leisti vartotojams naudotis ja per web sąsają. Tai palengvintų sistemos įdiegimą vartotojui. Tik prisijungus prie sistemos per WEB naršyklę, vartotojas turės galimybę nusiųsti arba gauti informaciją iš sistemos, padaryti paiešką sistemoje arba susirašinėti su kitais sistemos vartotojais. Iš kitos puses, vartotojas turi būti autorizotas, kad atliktų kokius nors veiksmus sistemoje. Tam jis turi turėti atitinkamas teises sistemoje. Tikslas yra administruoti vartotojus, priskyrimą jiems tam tikras teises naudotis sistema. Priskyrimą jiems straipsnius ir skyrelius, kuriuos jie galės matyti. Taip pat vartotojus reikės priskirti tam tikroms grupėms, nuo kurių priklauso vartotojo sistemoje galimybės.

#### **2.2.4 Veiklos panaudojimo atvejų modelis**

## Vartotojų



Pav 4. Veiklos panaudojimo atvejų modelis

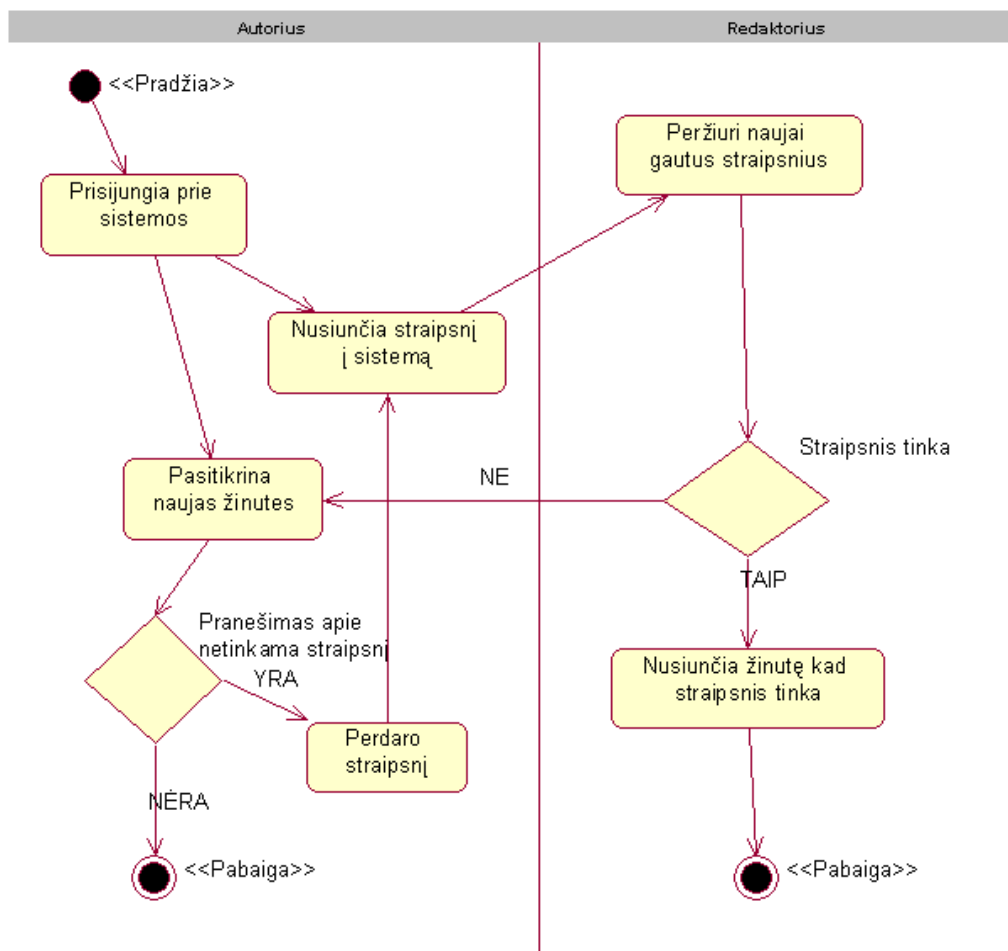
Modelis aprašo panaudojimą kuriamos informacijos surinkimo sistemos įmonėje arba padalinyje. Pagal vartotojo reikalavimus aprašyta sistema turi keturis pagrindinius panaudojimo atvejus. Kiekvienas atvejis įtraukia daugybę smulkių operacijų, kurių čia nebus aprašinėjama. Žemiau pateikti visi pagrindiniai panaudojimų atvejai, kurie pavaizduoti 4 pavyzdyje. Prie panaudojimo atvejų yra trumpi jų aprašymai.

Informacijos saugojimas	Vartotojai bet kada gali prisijungti prie sistemos ir paimti reikalingą dokumentą; taip pat išsaugoti naują ar pakeistą egzistuojantį dokumentą. Saugant dokumentą, kartu išsaugoma ir papildoma informacija apie dokumentą (metaduomenys).
Informacijos paieška	Vartotojas gali vykdyti dokumentų paiešką sistemoje. Žinodamas dokumento pavadinimą vartotojas gali surasti jį ir panašius į jį dokumentus.
	Sistemos administratorius kuria naujus vartotojus sistemoje, šalina egzistuojančius vartotojus iš sistemos, keičia jų teises. Administratorius

administravimas	gali priskirti vartotoją bet kuriai iš 4 grupių. Administratorius keičia ir atskiro vartotojo teises sistemoje. Administratorius gali nustatyti kiekvienam vartotojui atskirai teisių į straipsnius matomumą.
Informacijos apsauga	Kiekvienas vartotojas sistemoje yra identifikuojamas ir autentifikuojamas. Visi dokumentai ir vartotojai sistemoje turi tam tikras saugumo žymes, kurios nurodo, kokias teises turi kiekvienas vartotojas ir kokias saugumo žymes turi kiekvienas dokumentas.
Trumpų žinučių persiuntimas	Kiekvienas sistemos vartotojas turi teisę parašyti ir gauti trumpą žinutę kitam sistemos vartotojui. Visos žinutės yra siunčiamos naudojant sistemos duomenų bazę.
Informacijos (dokumentų) valdymas	Redaktoriai gali kelti visus arba atskirus dokumentus iš vieno skyriaus į kitą, taip pat jie gali trinti dokumentus iš duomenų bazės. Paprastiems vartotojams šią galimybę geriausia apriboti per administravimo meniu.

## 2.2.5. Veiklos procesų modelis

Paveikslėlyje 5 pavaizduotas procesą pavaizduoją naujo straipsnio įvedimo į sistema procesas. Šiame procese dalyvauja du aktoriai. Procesas prasideda kai Autorius prisijungia prie sistemos. Prisijungus autorius perskaito žinutes arba nusiunčia straipsnį į sistemą. Pirmuoju atveju autorius pasižiūri ar nėra žinučių apie blogai arba nekokybiškai atliktus darbus iš redaktoriaus. Antru atveju autorius talpina savo darbus į duomenų bazę. Vėliau redaktorius gali pasižiūrėti juos ir parašyti savo atiliepimus. Jei straipsniai yra netinkami redaktorius parašo savo pastabas trumpojoje žinutėje autoriui ir naikina straipsnį iš sistemos. Jei straipsnis tinkama redaktorius praneša apie tai autoriui ir procesas užsibaigia.



Pav 5. Veiklos procesų modelis Straipsnio siuntimas į sistemą

## 2.2.6 Vartotojų ir jų poreikių analizė

Informacijos surinkimo sistemos vartotojai:

- Sistemos administratorius;
- Sistemos vartotojas;

Sistemos administratorius – asmuo, kuris kontroliuoja visos sistemos darbą, administruoja vartotojus, jų teises, kuria bei realizuoja sistemos saugumo politiką.

Sistemos vartotojas – asmuo, kuris naudoja sistemą dokumentų saugojimui, jų paieškai bei jų išgavimui iš duomenų bazės.

Vartotojų poreikiai informacijos surinkimo sistemai – tai reikalavimai ir norai sistemos funkcionalumui. Vartotojas nori, kad sistema atliktų bei išspręstų jam aktualius uždavinius ir procesus. Vartotojų poreikiai formuluojami pasinaudojant analizės etapo modeliais – tikslų, procesų, veiklos panaudojimo atvejų.

### **Sistemos administratoriaus poreikiai:**

- Vartotojų sukūrimas ir modifikavimas;
- Vartotojų teisių suteikimas ir keitimas;
- Egzistuojančių dokumentų įtraukimas į sistemą;
- Vartotojų autentifikacijos mechanizmo užtikrinimas;
- Vartotojų identifikacijos mechanizmo užtikrinimas;
- Sistemos administravimas per web tinklą;

### **Sistemos vartotojo poreikiai:**

- Naujo ar pakeisto dokumento saugojimas sistemoje;
- Metaduomenų apie dokumentą įvedimas;
- Dokumento paieška sistemoje;
- Kitų vartotojų priėjimo teisių prie dokumentų kontroliavimas;
- Priėjimas prie dokumentų per web tinklą;
- Dokumentų klasifikavimas pagal įvairius skyrius;
- Automatinė dokumentų registracija;
- Žinutės vartotojams;
- Bendra ar detali dokumento proceso būsenos informacija;
- Procesų dalyvių rolių suteikimas: administratorius, vyr. Redaktorius, Autorius, Redaktorius

Šie išvardinti poreikiai reikalingi pilnos ir išbaigtos informacijos surinkimo sistemos



sukūrimui. Visi šie poreikiai yra svarbūs, nes vartotojų veiksmai bei uždaviniai yra gausūs, ir jiems visiems realizuoti reikia įgyvendinti šiuos uždavinius.

### **2.2.7 Nefunkciniai reikalavimai sistemai**

Sistemos pagrindinė paskirtis – palengvinti informacijos surinkimo padalinio uždavinį tvarkyti ir valdyti straipsnių srautą, greitai ir nesudėtingai rasti reikiamus straipsnius. Labiausiai tikėtinas sistemos vartotojas – neturintis profesionalių žinių informacinių technologijų srityje. Atsižvelgiant į tai, sudaromi nefunkciniai reikalavimai sistemai. Nefunkcinius reikalavimus galima suskaidyti į detalesnius. Sistemai keliami tokie nefunkciniai reikalavimai:

- Paprastumas – darbas su sistema turi būti nesudėtingas, nereikalaujantis iš vartotojo gilių žinių informacinių technologijų srityje.
- Patikimumas – sistema turi būti patikima, t.y. korektiškai atlikti jai pavestas užduotis ir funkcijas.
- Efektyvumas – sistema turi atlikti jos vykdomas funkcijas greičiau, negu tai būtų atliekama be sistemos, ir kad jos atliekamos funkcijos ir jų atlikimo sparta būtų realiai naudingi sistemos vartotojams.
- Nesudėtingas eksploatavimas – sistemos naudojimas neturi reikalauti didesnių papildomų žmogiškųjų išteklių bei materialinių resursų.
- Sistemos atnaujinamumas – turi būti galimybė keisti sistemos funkcionalumą, nepertraukiant sistemos darbo.
- Saugumas – vartotojai, naudodami šią sistemą, turi jaustis saugiai dėl savo dokumentų.
- Grafinės vartotojo sąsajos intuityvumas ir aiškumas – vartotojo sąsaja turi būti paprasta, patogi, logiška.



## 2.2.8 Informacijos surinkimo sistemos klasių diagrama

Klasė *AdminPage* apibrėžia vartotojo ar sistemos pateiktą užklausą sistemai administruoti vartotoją pagal nurodytus kriterijus. Kiekviena užklausa turi savo unikalų identifikatorių (atributas *userListID*), užklaustos turinį. Gali būti sukuriama nauja užklausa (metodas *CmdNew\_Click*) vartotojui sukurti, ištrinamas jau sukurtas vartotojas per užklausą (metodas *Delete\_Click*), koreguojami duomenys apie vartotojus, naudojant *Save\_Click*, *dgSkyr\_Update\_Command* metodus.

Klasė *Straipsn* apibrėžia vartotojui matomus straipsnius. Klasės pagalba galima atnaujinti straipsnių matomumo duomenys vartotojui. Metodas *update\_click* suformuoja užklausą, kurioje nusiunčiami duomenys apie matomus vartotojui straipsnius. Prieš straipsnių atnaujinimą straipsnių sąrašas išvedamas į ekraną metodo *straipsniai* pagalba

Klasė *login* apibrėžia užklausą, leidžiančią patikrinti vartotojo prisijungimo duomenis ir jo teises. Prisijungimo lange vartotojui surinkus savo vartotojo vardą ir slaptažodį, vartotojo duomenys siunčiami į serverį *MainService* klasės pagalba, iškviečiant *btnLogin\_Click* metodą. Toliau vartotojo ekrane pasirodo pagrindinis sistemos langas, kurį valdo klasė *MainPage* arba klaidos aprašymas.

Klasė *MainPage* yra sistemos kliento dalies klasė. Ji valdo pagrindinį sistemos puslapį. Klase formuoja užklausas perėjimui prie kitų sistemos funkcinių dalių. Taip pat klasė formuoja užklausą pagrindinio lango duomenim parodyti. Suformuoti duomenims ir atvaizduoti juos vartotojo sąsajos lange - naudojami metodai *GetSkyrList*, *BindSkyrList*. Formuojamos užklaustos perėjimo prie skyrelių *lstSarasas\_SelectedIndexChanged* metodo pagalba. Perėjimui prie kitų sistemos teikiamų funkcijų, naudojami metodai *btnReloadMessages\_Click*, *btnLogOff\_Click*, *btnAdmin\_Click*, *btnNewMessage\_Click*, *btnNewStraip\_Click*.

Klasė *StraipDetail* naudojama turimų metaduomenų apie dokumentą atvaizdavimui. Klasėje *StraipDetail* numatoma saugoti dokumento komentarus (atributas *txtApras*), Kiekvieni metaduomenys apie dokumentą turi unikalų identifikatorių *ID*. Metaduomenys gali būti atnaujinti (metodas *refreshDS*).

Klasė *Laiškas* naudojama trumpoms žinutėms, kurias galima siųsti naudojant sistemą, atvaizduoti arba į jas atsakyti arba ištrinti. Atitinkamai šios funkcijos realizuotos metodais *btnReply\_Click* ir *btnDelete\_Click*.

## 2.3 Literatūros šaltiniuose pateiktų sprendimų problemai spręsti lyginamoji analizė

### 2.3.1 Panašių sistemų analizė

Apžvelgiamos kelios dokumentų valdymo sistemos: SAURUS™,. Šios sistemos yra vienos iš pasaulinių lyderių, kuriant dokumentų valdymo sistemas. Šios sistemos nėra vien tik dokumentų surinkimo ir aprašymo sistemos.

SAURUS CMS ([www.saurus.info](http://www.saurus.info)) paketas.

Saurus CMS dokumentų valdymo sistema skirta įvairaus dydžio sistemoms: nuo asmeninių kompiuterių, serverių iki internetinių sistemų. Sistema gali tiek valdyti duomenis, tiek ir kurti WEB Svetaines duomenims atvaizduoti. Sistema vartoja vartotojo autentifikaciją dokumentų apsaugai. Dokumentus saugoja duomenų bazėse. Naudoja Microsoft SQL Server, tačiau gali prisijungti ir naudoti kitas, jau egzistuojančias organizacijoje duomenų bazes. Taip pat sistema turi galimybes talpinti paveikslėlius ir juos vaizduoti galerijoje. Sistema įdiegiama labai paprastai per WEB naršyklę. Konfigūracija ir derinimas irgi nereikalauja ypatingų žinių kompiuterių srityje.

PS BOMBAY ([www.pindersoft.com](http://www.pindersoft.com)) paketas.

PS Bombay yra duomenų bazė kuri talpina žurnalus, straipsnius ir autorius. Idealiai pritaikyta autoriams, rašytojams, laisvai pasamdomiems rašytojams ir reporteriams kurie norėtu talpinti ir saugoti savo informacija sudėtingoje bet lengvai naudojamoje duomenų bazėje. Su PS Bombay pagalba Galima išsaugoti ir išgauti autorių informacija, žurnalo leidėjus ir straipsnius, taip pat ir kontaktine informacija.

Autoriams , leidėjams arba šiaip skaitantiems vartotojams ši galinga maža duomenų bazė leidžia iš saugoti informaciją kurios jums nereikia o vėliau ją labai greit surasti. Talpinant duomenys į tokia duomenų bazę pradingsta poreikis talpinti informacija skirtinguose vietose, o paieškos sistema duoda galimybe greitai surasti reikiama dokumentą. Deja ši sistema neturi priėjimo prie duomenų bazės per WEB naršyklę. Sistema pritaikyta dirbti kaip aplikacija vartotojo darbinėje stotyje.

### **2.3.2 Duomenų talpinimo į duomenų bazės technologijų paliginimas**

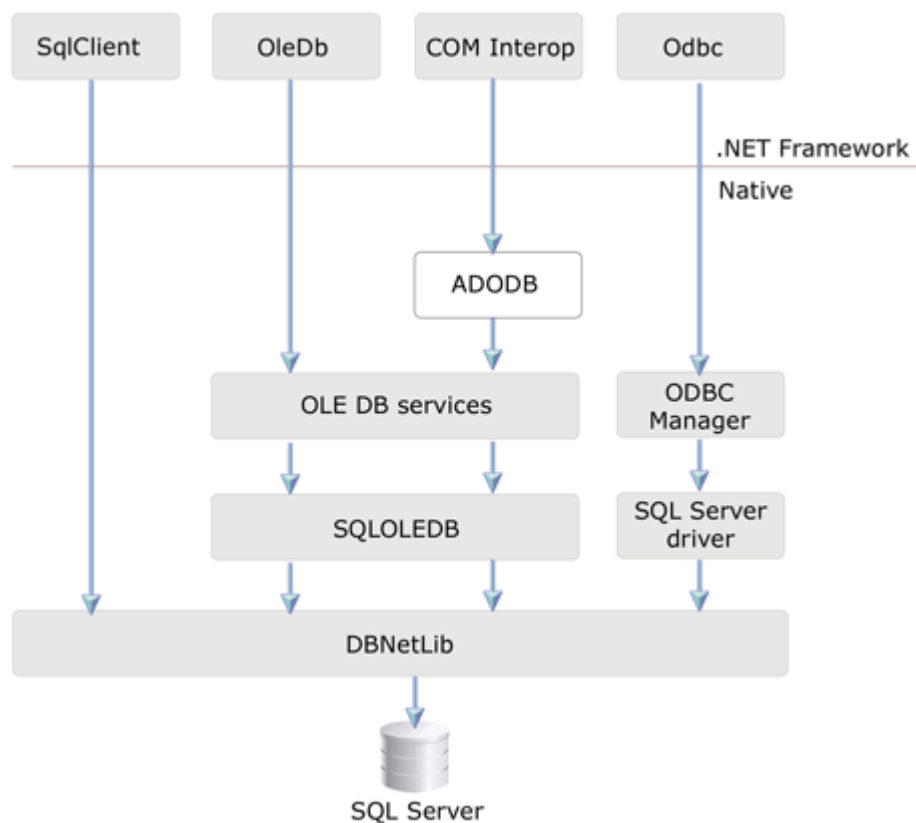
Sistemoms kurios naudoja internetą kaip duomenų nešėja, svarbu pasirinkti technologija užtikrinančia kuo geresnį duomenų perdavimo ir reakcijos laiką. Šiam projektui iš anksto buvo planuojama rinktis iš dviejų technologijų JAVA ir .NET. Kadangi sistemos branduolys – WEB Servis'as veiktu Microsoft operacinėje sistemoje , tai Java ir .NET pilnai tiktų projektui realizuoti. Bet padarius tyrimą tos sistemos dalies kuri kreipiasi į duomenų bazę sudariau

paliginimų lentelė iš kurios priėjau išvados kad tinkamiausia technologija būtų .NET.

Palyginimas Ado.Net ir JDO technologijų:

	Ado.Net	JDO
Duomenų modelis	Reliacinis	Objektinis
Duomenų apibrėžimo kalba ( <i>Data Definition Language</i> )	XML	Java ir XML
Užklausų kalba	SQL	JDOQL
Duomenų manipuliavimo kalba	SQL ir (C# arba C++ arba Visual Basic)	Java
Paveldėjimo ir polimorfizmo realizacija	Ne	Taip
Unikalus identifikatorius	Pirminis raktas	Pirminis raktas arba duomenų šaltinio identifikatorius
Transakcijos	Taip	Taip
Konkurentiškumas	Taip	Taip
Objektinio duomenų modelio palaikymas užklausose	Ne	Taip

Kuriant Java klases, naudojamas sąsajai su duomenų šaltiniu, naudojamas Java išeities tekstas ir XML byloje užrašyti metaduomenys. JDO technologija neturi MS SQL Server programinio palaikymo, todėl konvertuoti MS SQL Server RDB į ODB (Objektinę duomenų bazę) ji negali, todėl pasirinkta gimininga Ado.NET technologija, kuri yra sukurta kompanijos Microsoft ir nereikalauja įdiegti papildomų priemonių manipuliuojant duomenimis su MS SQL Server. (6.Pav)



6.Pav ADO.Net bendradarbiavimo būdai su MS SQL Server

Iš 6 pav. galime padaryti išvadą, kad SQL darbinė stotis turi labai lankstų priėjimą prie jos ir įdiegimo atžvilgiu yra geriausiai tinkančia tiriamam darbui.

#### 2.4. Projekto tikslas ir jo pagrindimas, kokybės kriterijų apibrėžimas

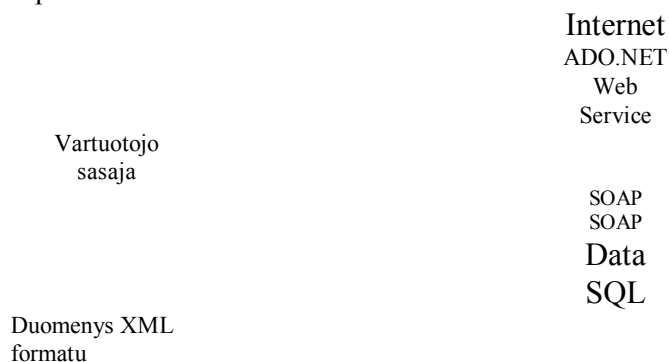
Šio projekto tikslas: sukurti programinę įrangą, kuri leistų sudarinėti periodinių leidinių turinį, teiktų tokias paslaugas kaip užsakovo ir vykdytojo bendravimas tarpusavyje internet aplinkoje, duomenų užkrovimas, parsisiuntimas naudojant interneto tinklą. Programinė įranga turi naudoti web sąsają, tai leistų lengvai prieiti prie sistemos iš bet kurios vietos, kur yra naršyklė ir interneto ryšis. Vartotojų hierarchiją taip pat yra būtina ir griežtai apibrėžta. Tik

vyriausias redaktorius gali priskyrinėti skiltis skilčių redaktoriams. Vartotojai netūrėtų matyti straipsnių, jeigu jie nėra autorizuoti sistemos. Taip pat reikia numatyti duomenų saugyklas, kuriose bus sudėti straipsniai ir laikmenos bei kurias norėtų įrašyti į kompaktinį diską ir pridėti prie leidinio. Šiomis duomenų saugyklomis galės naudotis tik užregistruoti vartotojai ir leidėjai. Naudojimosi teisės yra apribojamos vartotojų privilegijomis.

Tikslo pagrindimas būtų ASP.NET , ADO.NET, MS SQL ir Webservice panaudojimas konkrečioje dalykinėje srityje. Webservisas turėtų apdoroti visas sistemos vartuotoju užklausas. Kitaip tariant - tai bus pagrindinis sistemos veikėjas. ASP.NET pagalba bus sukurtos sąsajos per kurias vartotojai valdys ir naudosis duotojo serviso paslaugomis

- Tikslui pasiekti turi būti išspręsti tokie uždaviniai:
- išanalizuoti tyrimų sritį (reikalavimų specifikacijos struktūrą bei semantiką),
- išsiaiškinti reikalavimus kuriamai sistemai,
- paruošti sistemos projektą,
- programą realizuoti ir išbandyti.
- duomenys tarp vartotojo ir darbinės stoties, neatžvelgiant į tai kokios operacijos bus atliekamos turės būti perduodami XML arba SQL pavidalu. ( 7.Pav. )

Sql Užklausa



Vartotojo Pusė  
XML Webservice  
Darbinės stoties Pusė

7. Pav Duomenų gražinimas XML pavidalu iš MS SQL serverio

Pagrindinis projekto kokybės kriterijus – sukurtos programinės įrangos funkcionalumas. Pagrindinė sukurtoji programinės įrangos funkcija turės – leisti straipsnių tiekėjams ir tų straipsnių redaktoriams formuoti periodinio leidinio turinį iš esamų duomenų bazėje bei



užtikrinti bendravimą tarpusavyje. Duomenų bazėje bus saugoma informacija gauta iš įvairių informacijos šaltinių. Turės būti galimybė redaguotą informaciją pateikti atgal užsakymo informacijos teikėjui tam, kad jis ją peržiūrėtų ir pataisytų klaidas, įneštų pakeitimus.

Taip pat turi būti patikrinta, ar sukurtoji programinė įranga:

- Lengvai įdiegiama į darbinės stotis,
- yra išbaigta, atitinka jai visus iškeltus funkcinius reikalavimus,
- yra tolerantiška klaidoms,
- yra perkeliama, ar ją kartu su reikalavimų duomenų baze galima perkelti į bet kurią MS aplinką atitinkančia programinės įrangos reikalavimus.

## 2.5 Kompiuterizuojamos sistemos varianto parinkimas

Uždaviniai, kurie kyla kuriant sistemą yra griežtai apibrėžti. Sistema turi užtikrinti bendravimą tarp vartotojų. Sistema turi užtikrinti duomenų persiuntimą tarp vartotojo ir leidyklos darbinių stočių.

Betarpiskam bendravimui tarp vartotojų pasirinktas variantas, kai vartotojai susirašinėja tarpusavyje trumpom žinutėm. Sistema apdoroja šias žinutes ir nusiunčia jas adresatui. Toks bendravimo metodas realizuojamas naudojant standartines web naršyklės. Jokių papildomų priemonių nereikės diegti į vartotojo darbinės stotis. Toks pigiausias variantas užtikrins produkto prieinamumą nedidelėms įmonėms.

Duomenų apsikeitimams naudojamos tik web naršyklės ir sistema. Sistema realizuoja duomenų perdavimą naudojant binarinį duomenų siuntimo mechanizmą. Bandymai naudoti XML kalbą duomenim perduoti nedavė pakankamai gerų rezultatų, sistema nepalaikė didelių bylų bei perdavimo greitis neužtikrino reikalavimų.

Svarbiausi uždaviniai, kuriuos bus siekiama kompiuterizuoti, yra šie:

- Dokumentų saugojimas.
- Informacijos paieška.
- Vartotojų administravimas.
- Informacijos apsauga.

Numatoma informacijos valdymo ir surinkimo sistemą realizuoti web tinklo technologijų pagalba, pasitelkiant Microsoft .NET technologiją. Microsoft .NET technologija pasirinkta dėl to, kad projekto realizavimui bus naudojama C# programavimo kalba, o duomenų išgavimui ir perkėlimui į duomenų bazių darbinę stotį bus naudojama ADO.NET technologija, kuri taip pat įeina į .NET Framework<sup>®</sup> paketą. Įmanomas variantas naudoti savo galimybėmis panašią programavimo kalbą – JAVA (**Sun Microsystems**), tačiau JAVA programavimo kalba parašyti produktai naudoja žymiai daugiau kompiuterio resursų negu tokia pat programinė įranga, kuri realizuota naudojant C#. Technologijos panaudojamos sukurti panašaus tipo WEB programinę įranga yra ASP.NET technologija, kodavimui naudojant C# programavimo kalbą. Naudojama duomenų bazė – MSSQL, kurią pasiekiame per ADO.NET, naudojant SQL užklausų kalbą, Microsoft SQL Server 2000. Taip pat IIS (Internet Information Server) 6.0 versija.

Sistema turėtų komponuotis iš dviejų dalių: duomenų bazės ir priėjimo prie jos. DB realizavimui parinkta MSSQL valdymo sistema, o priėjimą galima realizuoti naudojant

kliento-darbinės stoties architektūrą MSSQL Server 2000 Enterprise. Kliento ir darbinės stoties dalims buvo parinktos ASP.NET, ir C# realizavimo kalbos.

Vartotojui reikalingas kompiuteris su interneto naršykle ir ryšys su centriniu kompiuteriu. Apribojimų vartotojų operacinei sistemai nėra.

Web technologija pasirinkta todėl, kad ja naudojantis yra paprasčiau realizuoti sistemą ir vartotojo interfeisą, negu kurti Windows™ operacinės sistemos taikomas programas. Naudojant web technologiją, nereikia programuoti taikomųjų programų, skirtų sistemos vartotojams prisijungti ir naudotis sistemos resursais. Tam bus panaudota interneto naršyklė.

## 2.6 Analizės išvados

Atlikus analizę, paaiškėjo, kad visas sistemos funkcijas ir jai keliamus uždavinius galima išskaidyti į 3 pagrindines grupes:

- Dokumentų saugojimas;
- Dokumentų paieška;
- Saugaus darbo užtikrinimas;

Dokumentų saugojimas iškelia tokias problemas kaip dokumentų saugojimo technologija, taip pat metaduomenų įvedimo problema, saugumo žymių realizavimas. Organizacija gali turėti labai daug duomenų, taigi reikia parinkti efektyvią ir tinkamą dokumentų saugojimo technologiją. Tai galėtų būti duomenų bazėmis paremta technologija, failinio saugojimo technologija ar šių metodų apjungimas. Tinkamiausias būdas saugoti dokumentus – naudoti duomenų bazes ir failų serverius. Jos užtikrina didesnę saugumą, galima vartoti sudėtingesnį informacijos struktūrizavimą, paprasčiau išsaugoti papildomus duomenis, paprastesnę ir efektyvesnę paiešką, saugumo žymių priskyrimas. Analizė parodė, kad metaduomenų įvedimas negali būti pilnai automatizuotas. Kai kurie duomenys turi būti įvesti vartotojo, nes neįmanoma automatizuoti kai kurių duomenų apie dokumento įvedimą (pvz.: komentarų) . Tuo būdu atsiranda žmogiškasis faktorius, kuris lemia tai, kad žmogus gali suklysti arba sąmojingai įvesti klaidingą ar ne visą informaciją. Sistema turi užtikrinti dokumentų apsaugą. Tai pakankamai sudėtingas uždavinys. Užtikrinant saugumą, taip pat dalyvauja žmogus, kas apsunkina saugumo užtikrinimą.

Dokumentų paieškai efektyviai atlikti reikalingi metaduomenys apie kiekvieną dokumentą sistemoje. Metaduomenyse saugoma visa reikalinga paieškai informacija (pvz.: dokumento pavadinimas, redagavimo data ir pan.). Kuo daugiau turima metaduomenų apie dokumentą, tuo efektyvesnė paieška. Organizacija prieš sistemos įdiegimą jau turi sukaupusi didelius dokumentų kiekius. Dažniausiai šie dokumentai yra svarbūs, tad jų prarasti negalima. Įtraukus į sistemą šiuos dokumentus, galima jais pilnai operuoti: saugoti, atlikti paiešką. Šių dokumentų įtraukimas į sistemą yra pakankamai sudėtingas. Kaip jau paminėta šiame skyriuje, ne visus metaduomenis galima įvesti automatiškai. Iš to seka, kad ir esamų duomenų įtraukimas į sistemą negali būti automatizuotas. Dokumentų paieška turi būti vykdoma pagal vartotojo užklausoje nurodytus kriterijus. Efektyviai dokumentui paieškai reikia naudoti kiek galima daugiau užklausoje parametrų, tad vartotojui turi būti suteikta galimybė įvesti šiuos parametrus.

Norint saugiai dirbti su organizacijos dokumentais, reikia užtikrinti saugų jų perdavimą kompiuteriniais tinklais. Internetu keliaujantys dokumentai gali būti pagrobti, jų turinys gali būti pakeistas, metaduomenys sugadinti. Tokiu atveju organizacija gali prarasti svarbius dokumentus, dokumentuose saugoma informacija gali būti klaidinga, informacija gali pakliūti tiems asmenims, kurie neturi teisės naudotis ta informacija. To reikia išvengti, todėl privaloma naudoti kriptografiją, kuri užtikrintų saugų dokumentų perdavimą kompiuteriniais tinklais. Tai galėtų būti SSL (Secure Sockets Layer), TSL (Transport Layer Security) kodavimo protokolai arba tiesiog DES3 duomenų kodavimas prieš perduodant į kitą darbinę stotį.

Kiekviena organizacija turi saugumo politiką. Tai reiškia, kad pagal organizacijos saugumo politiką tam tikriems dokumentams reikia nustatyti atitinkamas saugumo žymes. Saugi virtualios organizacijos dokumentų valdymo sistema turėtų palikti kuo mažiau erdvės vartotojų klaidoms realizuojant saugumo politiką. Reali saugumo politika turi kuo daugiau atitikti teorinę saugumo politiką, dokumentų saugumo žymės ir vartotojų teisės turi būti tokios, kaip numatyta organizacijos saugumo politikoje. Kuo daugiau žmonių turi teises nustatyti ir keisti saugumo žymes, tuo sunkiau valdyti ir kontroliuoti teisingą saugumo politikos vykdymą. Reikia paskirti vartotoją (-us), kurie kontroliuotų informacijos, procesų, vartotojų saugumo žymes.

Duomenų saugančios sistemos privatumas – tai svarbios informacijos laikymas, kuri negali būti atskleista kitoms sistemos dalyviams.

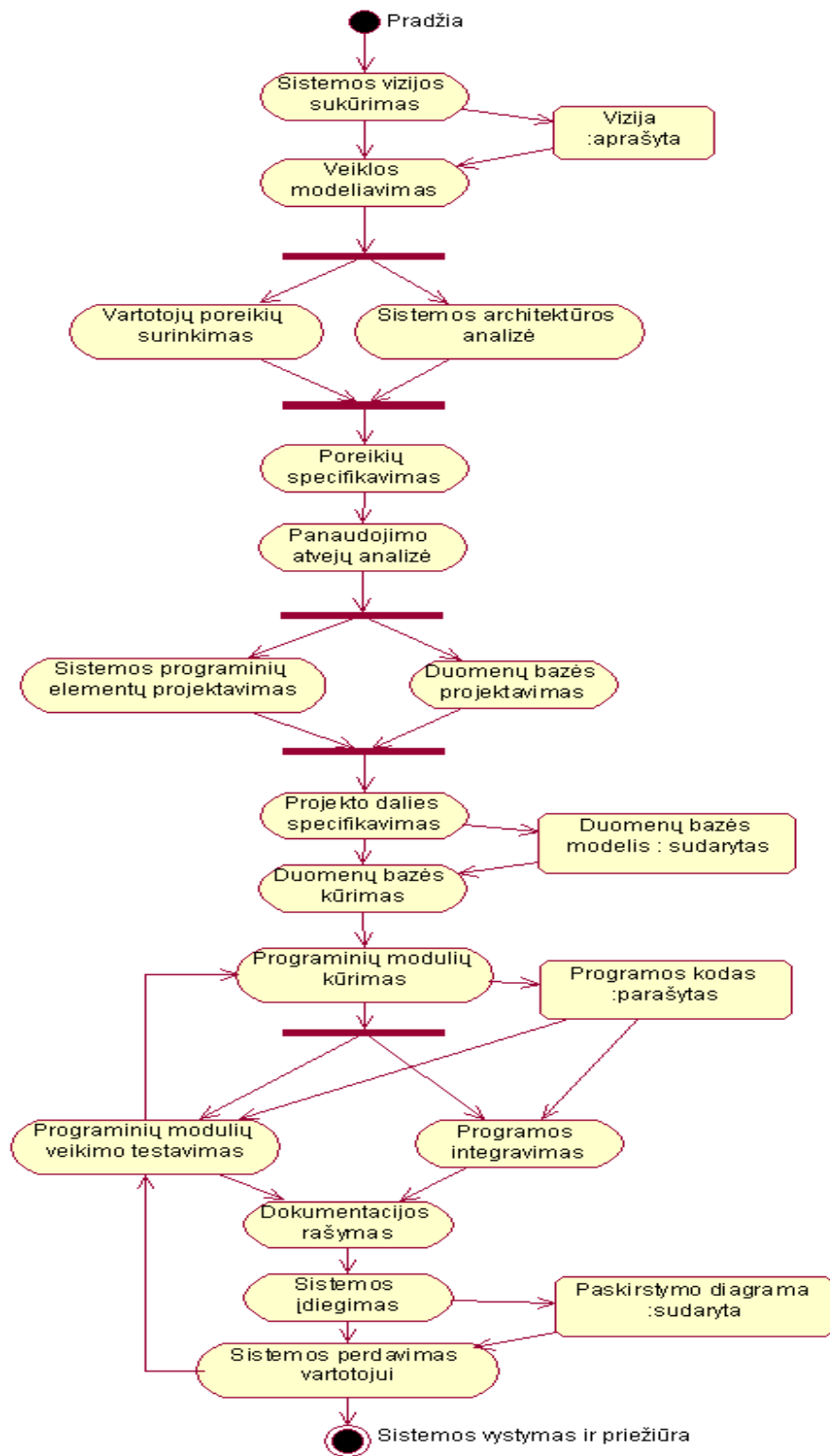
### 3. Projekto dalis

Projektui realizuoti panaudojamas RUP projektavimo metodas. RUP projektavimo metodo blok schema pavaizduota 8 pvz. Kaip ir visi projektai iš pradžių yra sukuriama projekto vizija ir ja sukūrus pereinama prie veiklos modeliavimo.

Vizijos kūrimo etape būtina kuo konkrečiau aprašyti norima rezultatą. Detaliau parašius viziją bus žymiai lengviau pereiti prie veiklos modeliavimo etapo. Taigi šiam projektui vizijos apibrėžimas būtų toks – „Sukurti sistemą, kuri leistų vartotojams prisijungti, nusiųsti arba parsisiųsti duomenų bylas, taip pat sistema privalo leisti administruoti ją besinaudojančių vartotojų teises ir užtikrinti minimalų bendravimo tarpusavyje mechanizmą trumpų žinučių pavidalu. Sistema turi palaikyti tolimesnio praplečiamumo galimybes“.

Apibrėžus savo viziją - tai yra ką mes norim matyti galutiniam variante - galime pereiti prie sistemos veiklos modeliavimo. Veiklos modeliavimas išsiskaido į dvi dalis. Vartotojo poreikių susirinkimą ir sistemos architektūros analizę. Vartotojo poreikių surinkimas yra svarbiausias etapas sistemos projektavime. Praktika parodo, kad nekompetentingai surinkti vartotojo poreikiai vėlesniuose etapuose gali privesti prie sistemos netinkamumo ir neatitikimo kokybės bei kriterijų. Lygiagrečiai vartotojo poreikių surinkimams reikia vesti sistemos architektūros analizę. Būtina išanalizuoti kuriamą sistemą ir įsitikinti, kad vartotojo poreikiai būtų pilnai užtikrinti bei sistemą būtų galima plėsti toliau vartotojui panorėjus. Kadangi kuriama sistema bus naudojama interneto terpėje reikia atsižvelgti ir į techninius sistemos reikalavimus. Būtinai peržiūrėti visas įmanomas prisijungimo prie sistemos alternatyvas. Pasirinkimas darbinės stoties ir duomenų saugykloms naudojamos programinės įrangos yra gana svarbus etapas. Sistemos architektūros analizės metu bus parinktas MSSQL Server ir .NET technologija tam kada realizuoti vartotojo poreikius. Bus naudojama IIS 5.0 arba aukštesnės versijos programinė įranga, kuri turės saugoti ir apdoroti sistemos programines bylas. Iš programinės pusės sistema turės galimybę veikti ant kelių darbinių stočių. Sistema turės architektūrą klientas - serveris. Tam realizuoti bus naudojami WEB Services, kurie yra dalys .NET architektūros.

Preliminariai sudarius sistemos architektūrą ir surinkus vartotojo poreikius pereiname prie poreikių specifikavimo etapo. Neformaliai aprašyti poreikiai šiame etape formalizuojami ir specifikuojami. Poreikiai surašomi ir griežtai aprašomi. Toliau numatomas panaudojimo atvejų analizės etapas. Poreikiai transformuojami į panaudojimo atvejus.



### 8.pav projektavimo metodas (pagal Rational Unified process)

Panaudojimo atvejų analizė - tai veiksmų, kurie turi būti atlikti sistemai bendraujant su aktoriumi seka, užsibaigianti konkrečiu rezultatu. Veiksmų seka turi būti tokia, kad ją įvykdžius sistema pereitų į būseną, kurioje vartotojui vėl būtų leista pakartoti tą veiksmų seką. Panaudojimo atvejai turi būti tokie, kad padėtų vartotojui pasiekti jo tikslus. Kiekvienam vartotojo tikslui turi būti bent vienas panaudojimo atvejis. Panaudojimo atvejai aprašo sistemos elgesį, bet ne jos realizaciją. Tuo jie įgalina vartotojus, projektuotojus, ekspertus suprasti sistemą. Po šio etapo pereiname prie sistemos programinių elementų projektavimo ir lygiagrečiai duomenų bazės projektavimo.

Po to, kai duomenų bazė ir sistemos programiniai elementai jau yra suprojektuoti, seka Duomenų bazės kūrimo ir programinių modulių kūrimo etapai. Šiuose etapuose jau praktiškai realizuojama tai, kas buvo suplanuota ankstesniuose, sistemos kūrimo, etapuose. Rašomas programinis kodas ir kuriama reali duomenų bazė.

Programinių modelių testavimo metu testuojamas parašytas programinis kodas. Testai gali kartotis tol, kol nebus įsitikinta, kad programa arba sistema atitinka saugumo ir kitus kriterijų. Ištestavus sistemą galima pereiti prie programos integravimo etapo. O vėliau prie dokumentacijos rašymo.

Galiausiai prieinama prie sistemos įdiegimo. Šiame etape sistema diegiama į darbinę stotį ir vartotojai jau gali realiai bandyti tai, kaip buvo realizuoti ir suprasti jų poreikiai.

### **3.1 Techninė užduotis**

#### DUOMENŲ SURINKIMO IR VALDYMO SISTEMA PEREODINIO LEIDINIO TURINIUI FORMUOTI

Realizuoti WEB aplikaciją, kuri dirbtų principu Klientas-Serveris. Realizuoti kliento dalį kaip WEB Aplikacija (web application), o darbinės stoties dalį kaip WEB Servisą (web service) . Padaryti priėjimą prie duomenų bazės per web servisą. Suprogramuoti WEB servisą



ir WEB aplikacija, naudojant ASP .NET ir C# .NET. Sukurti duomenų bazę, naudojant Microsoft SQL Server 2000. Sistema turi būti suprojektuota ir suprogramuota atitinkant reikalavimus sistemos funkcionalumui .

## 3.2 Reikalavimų modelis

### 3.2.1 Vartotojų panaudojimo atvejų modelis

Vartotojų panaudojimo atvejų diagrama ir panaudojimo atvejų specifikacijos

Zinutės išsiuntimas vartotojui  
 Vartotojas  
 Sk. Redaktorius  
 <<include>>  
 <<include>>  
 <<include>>  
 SISTEMA  
 <<include>>  
 Vyr.Redaktorius  
 Arba admin  
 Naujų vartotojų kurimas  
 <<include>>  
 Straipsniu  
 persiuntimas  
 <<include>>  
     Esamų vartotojų  
         redagavimas  
 Vartotojų pašalinimas  
     Skyrelių straipsnių  
     matomumo priskyrimas  
         vartotojams  
 Vartotojų priskyrimas prie  
     skyrelių

Pav 9. Vartotojo panaudojimo atvejų modelis

Panaudojimo atvejis	Žinutės siuntimas vartotojui
<b>Aktorius</b>	Redaktorius/Vartotojas
<b>Prieš sąlyga</b>	Turi būti prisijungus prie sistemos
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija ir sprendimai</b>
1. Prisijungiama prie sistemos 2. Pasižiūrima ar nėra naujų žinučių 3. Pereinama prie naujos žinutės rašymo. 4. Parašyta žinutė nusiunčiama	1.1 Sistema patikrina slaptažodį ir vykdo autorizaciją; 2.1. Sistema tikrina, su kokiomis privilegijomis buvo prisijungta; 2.2. Išduoda žinučių ir straipsnių sąrašą pagal vartotojo vardą 3.1 Sistema paruošia žmonių, kuriems galima siusti žinutę, sąrašą. 4.1 Siunčiama žinutė
<b>Po sąlyga</b>	Duomenų bazė papildyta, nauja žinutė priskirta gavėjui
<b>Alternatyvos (nesėkmės atvejai)</b>	2.1.a Neužpildyti visi būtinieji laukai: Sistema parodo pranešimą apie neužpildytus būtinuosius laukus ir grįžta į 1 žingsnį; 2.1.b Laukai užpildyti neteisingai: Sistema parodo pranešimą apie užpildytus neteisingai laukus ir grįžta į 1 žingsnį.
<b>Vykdyimo variantai</b>	

<b>Veiklos taisyklēs</b>	Vartotojas turi įvesti būtinuosius duomenis ir būtinai teisingai įvesti reikalingus duomenis
<b>Specialūs reikalavimai(nefunk.)</b>	Duomenys turi saugiai pasiekti duomenų bazę

<b>Panaudojimo atvejis</b>	<b>Naujų vartotojų sukūrimas</b>
<b>Aktorius</b>	Vyr. Redaktorius / Admin
<b>Prieš sąlyga</b>	Turi būti prisijungus prie sistemos
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija ir sprendimai</b>
1. Prisijungiama prie sistemos	1.1 Sistema patikrina slaptažodį ir vykdo autorizaciją ;
2. Pereinam į vartotojų administravimo puslapį	2.1. Sistema tikrina su kokiomis privilegijomis buvo prisijungta; 2.2. Perkeliama į vartotojų teisių administravimo puslapį;
3. Nuspaudžiamas mygtukas leidžiantis sukurti naują sistemos vartotoją	3.1 Sistema paruošia naujo vartotojo įvedimo formą ir išvalo laukus.
4. Užpildomi duomenys apie vartotoją ir nuspaudžiamas mygtukas leidžiantis išsaugoti naujo vartotojo duomenys į DB	4.1 Sistema patikrina duomenų teisingumą ir suformuoja duomenis vartotojų bazės papildymui.
5. Naujas vartotojas įrašomas į duomenų bazę	5.1 Išsiunčiami duomenys apie vartotoją į duomenų bazę
<b>Po sąlyga</b>	Duomenų bazė papildyta, naujas vartotojas atsiranda sąrašė
<b>Alternatyvos (nesėkmės atvejai)</b>	4.1.a Neužpildyti visi būtinieji laukai: Sistema parodo pranešimą apie neužpildytus būtinuosius laukus ir grįžta į 1 žingsnį; 4.1.b Laukai užpildyti neteisingai: Sistema parodo pranešimą apie užpildytus neteisingai laukus ir grįžta į 1 žingsnį.
<b>Vykdyimo variantai</b>	
<b>Veiklos taisyklės</b>	Vartotojas turi įvesti būtinuosius duomenis ir būtinais teisingai įvesti reikalingus duomenis
<b>Specialūs reikalavimai</b>	Duomenys turi saugiai pasiekti duomenų bazę, vartotojas turi būti autorizotas atlikti operaciją

<b>Panaudojimo atvejis</b>	Vartotojų pašalinimas
<b>Aktorius</b>	Vyr. Redaktorius / Admin
<b>Prieš sąlyga</b>	Turi būti prisijungus prie sistemos
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija ir sprendimai</b>
1. Prisijungiama prie sistemos	1.1 Sistema patikrina slaptažodį ir vykdo autorizaciją
2. Pereinam į vartotojų administravimo puslapį	2.1. Sistema tikrina, su kokiom privilegijomis buvo prisijungta; 2.2. Perkeliama į vartotojų teisių administravimo puslapį.
3. Nuspaudžiamas mygtukas leidžiantis šalinti sistemos vartotoją	3.1 Sistema pašalina vartotoją.
4. Naujas vartotojas pašalintas iš duomenų bazės	4.1 Atsinaujina vartotojų sąrašas kuriame jau nesimato vartotojo, kuris buvo pašalintas.
<b>Po sąlyga</b>	Duomenų bazė atnaujinta, vartotojas ištrintais
<b>Alternatyvos (nesėkmės atvejai)</b>	3.1.a Nenurodytas trinamas vartotojas: Sistema parodo pranešimą apie neužpildytus būtinuosius laukus ir grįžta į 1 žingsnį
<b>Vykdomo variantai</b>	
<b>Veiklos taisyklės</b>	Vartotojas turi įvesti būtinuosius duomenis
<b>Specialūs reikalavimai</b>	Duomenys turi saugiai pasiekti duomenų bazę, vartotojas turi būti autorizotas atlikti operaciją

<b>Panaudojimo atvejis</b>	Esamų vartotojų redagavimas
<b>Aktorius</b>	Vyr. Redaktorius / Admin
<b>Prieš sąlyga</b>	Turi būti prisijungus prie sistemos
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija ir sprendimai</b>
1. Prisijungiama prie sistemos	1.1 Sistema patikrina slaptažodį ir vykdo autorizaciją.
2. Pereinam į vartotojų administravimo puslapį	2.1. Sistema tikrina, su kokiom privilegijomis buvo prisijungta; 2.2. Perkeliama į vartotojų teisių administravimo puslapį.
3. Nuspaudžiamas mygtukas leidžiantis redaguoti pasirinkto sistemos vartotojo duomenis	3.1 Sistema išveda pasirinkto vartotojo duomenys į ekraną.
4. Užpildomi nauji duomenys apie vartotoją ir nuspaudžiamas mygtukas, leidžiantis išsaugoti naujo vartotojo duomenys į DB	4.1 Sistema patikrina duomenų teisingumą ir suformuoja duomenis vartotojų bazės papildymui.
5. Naujas vartotojas įrašomas į duomenų bazę	5.1 Išsiunčiami duomenys apie vartotoją į duomenų bazę

<b>Po sąlyga</b>	Duomenų bazė papildyta, vartotojo duomenys pasikeičia
<b>Alternatyvos (nesėkmės atvejai)</b>	4.1.a Neužpildyti visi būtinieji laukai: Sistema parodo pranešimą apie neužpildytus būtinuosius laukus ir grįžta į 1 žingsnį; 4.1.b Laukai užpildyti neteisingai: Sistema parodo pranešimą apie užpildytus neteisingai laukus ir grįžta į 1 žingsnį.
<b>Vykdomo variantai</b>	
<b>Veiklos taisyklės</b>	Vartotojas turi įvesti būtinuosius duomenis ir būtina teisingai įvesti reikalingus duomenis
<b>Specialūs reikalavimai</b>	Duomenys turi saugiai pasiekti duomenų bazę, vartotojas turi būti autorizotas atlikti operaciją

<b>Panaudojimo atvejis</b>	Vartotojų priskyrimas prie skyrelių
<b>Aktorius</b>	Vyr. Redaktorius / Admin
<b>Prieš sąlyga</b>	Turi būti prisijungus prie sistemos
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija ir sprendimai</b>
1. Prisijungiama prie sistemos	1.1 Sistema patikrina slaptažodį ir vykdo autorizaciją
2. Pereinam į vartotojų administravimo puslapį	2.1. Sistema tikrina, su kokiomis privilegijomis buvo prisijungta 2.2. Perkeliama į vartotojų teisių administravimo puslapį
3. Pasirenkamas vartotojas iš vartotojų sąrašo	3.1 Sistema išveda pasirinkto vartotojo duomenys į ekraną. Taip pat parodomas sąrašas skyrelių ir teisių į juos.
4. Pasirenkamas naujo skyrelio suteikimas arba skyrelio teisių redagavimo režimas	4.1 Sistema priskiria naują skyrelį vartotojui arba atitinkamai leidžia koreguoti skyrelio matomų teises.
5. Duomenys apie teises į skyrelius atnaujinami	5.1 Išsiunčiami duomenys apie vartotoją į duomenų bazę.
<b>Po sąlyga</b>	Duomenų bazė papildyta, vartotojo skyrelių matomumas pasikeičia
<b>Alternatyvos (nesėkmės atvejai)</b>	4.1.a Neužpildyti visi būtinieji laukai: Sistema parodo pranešimą apie neužpildytus būtinuosius laukus ir grįžta į 1 žingsnį; 4.1.b Laukai užpildyti neteisingai: Sistema parodo pranešimą apie užpildytus neteisingai laukus ir grįžta į 1 žingsnį.
<b>Vykdomo variantai</b>	
<b>Veiklos taisyklės</b>	Vartotojas turi įvesti būtinuosius duomenis ir būtina teisingai įvesti reikalingus duomenis
<b>Specialūs reikalavimai</b>	Duomenys turi saugiai pasiekti duomenų bazę, vartotojas turi būti autorizotas atlikti operaciją



<b>Panaudojimo atvejis</b>	Skyrelių straipsnių matomumo priskyrimas vartotojams
<b>Aktorius</b>	Vyr. Redaktorius / Admin
<b>Prieš sąlyga</b>	Turi būti prisijungus prie sistemos
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija ir sprendimai</b>
1. Prisijungiama prie sistemos	1.1 Sistema patikrina slaptažodį ir vykdo autorizaciją
2. Pereinam į vartotojų administravimo puslapį	2.1. Sistema tikrina, su kokiom privilegijomis buvo prisijungta 2.2. Perkeliama į vartotojų teisių administravimo puslapį
3. Pasirenkamas vartotojas iš vartotojų sąrašo	3.1 Sistema išveda pasirinkto vartotojo duomenis į ekraną. Taip pat parodomas sąrašas skyrelių ir teisių į juos.
4. Nuspaudžiamas mygtukas, leidžiantis matyti straipsnių sąrašą, kurį gali matyti vartotojas	4.1 Sistema pagal vartotojo teises išveda sąrašą straipsnių, kuriuos gali matyti vartotojas.
5. Duomenys apie teises į straipsnius atnaujinami	5.1 Išsiunčiami duomenys apie vartotoją į duomenų bazę
<b>Po sąlyga</b>	Duomenų bazė papildyta, vartotojo straipsnių matomumas pasikeičia
<b>Alternatyvos (nesėkmės atvejai)</b>	4.1.a Neužpildyti visi būtinieji laukai: Sistema parodo pranešimą apie neužpildytus būtinuosius laukus ir grįžta į 1 žingsnį; 4.1.b Laukai užpildyti neteisingai: Sistema parodo pranešimą apie užpildytus neteisingai laukus ir grįžta į 1 žingsnį.
<b>Vykdomo variantai</b>	
<b>Veiklos taisyklės</b>	Vartotojas turi įvesti būtinuosius duomenis ir būtinai teisingai įvesti reikalingus duomenis
<b>Specialūs reikalavimai</b>	Duomenys turi saugiai pasiekti duomenų bazę, vartotojas turi būti autorizotas atlikti operaciją

<b>Panaudojimo atvejis</b>	Straipsnių persiuntimas
<b>Aktorius</b>	Vyr. Redaktorius / Admin / Autorius / Straipsnių redaktorius
<b>Prieš sąlyga</b>	Turi būti prisijungus prie sistemos
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija ir sprendimai</b>
1. Prisijungiama prie sistemos	1.1 Sistema patikrina slaptažodį ir vykdo autorizaciją
2. Pereinam į naujo straipsnio patalpinimo į darbinę stotį puslapį	2.1. Sistema tikrina, su kokiom privilegijomis buvo prisijungta; 2.2. Jei leistina siusti naujus straipsnius perkeliama į naujo straipsnio siuntimo puslapį
3. Užpildomas straipsnio pavadinimo laukas , straipsnio aprašymo laukas , nurodomas	3.1 Sistema išveda laukus užpildymui į ekraną . Prikabinama byla, kurioje kaip manoma, yra straipsnis.

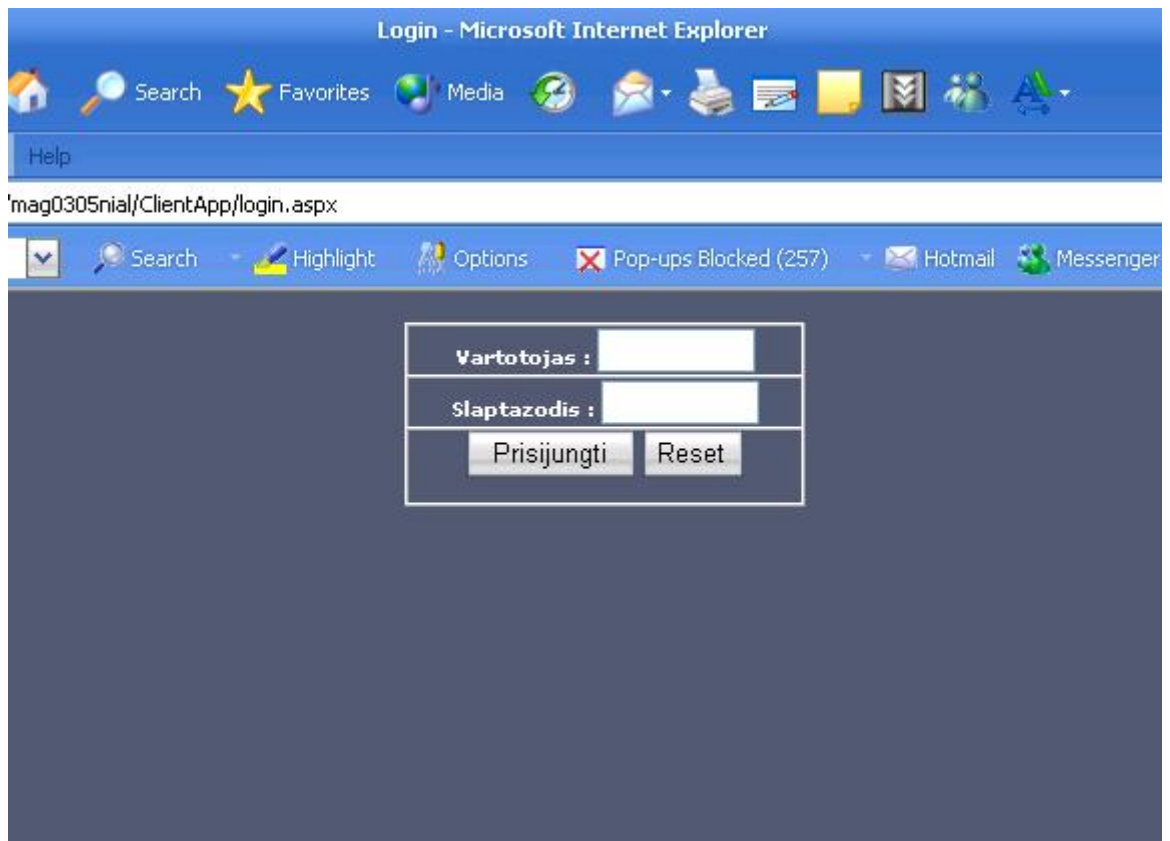


skyrius, į kuri bus talpinamas straipsnis ir nurodoma byla, kuri bus siunčiama į darbinę stotį	
4. Nuspaudžiamas mygtukas, leidžiantis nusiųsti straipsnį į darbinę stotį.	4.1 Sistema pradeda siuntimą , iš pradžių siunčiama byla, po to informacija apie straipsnį
5. Naujas straipsnis pridedamas prie straipsnio sąrašo.	5.1 Išsiųsti duomenys patalpinami į duomenų bazę
<b>Po sąlyga</b>	Duomenų bazė papildyta, vartotojo straipsniu
<b>Alternatyvos (nesėkmės atvejai)</b>	4.1. a Neužpildyti visi būtinieji laukai: Sistema parodo pranešimą apie neužpildytus būtinuosius laukus ir grįžta į 1 žingsnį; 4.1. b Laukai užpildyti neteisingai: Sistema parodo pranešimą apie užpildytus neteisingai laukus ir grįžta į 1 žingsnį. 4.1. c Byla nepatalpinta į darbinę stotį: Sistema praneša apie talpinimo klaidą
<b>Vykdyimo variantai</b>	
<b>Veiklos taisyklės</b>	Vartotojas turi įvesti būtinuosius duomenis ir būtinai teisingai įvesti reikalingus duomenis
<b>Specialūs reikalavimai</b>	Duomenys turi saugiai pasiekti duomenų bazę, vartotojas turi būti autorizotas atlikti operaciją

### 3.2.2 Vartotojo grafinės sąsajos modelis

#### Navigavimo planas.

Prisijungimo langas  
 Pagrindinis puslapis  
 Administravimo /  
 vartotojo puslapis  
 Naujo straipsnio  
 siuntimo puslapis  
 Žinutės skaitymo /  
 atsakymo puslapis  
 Straipsnių matomumo administravimo puslapis  
**Prisijungimo langas**



Pav. 10 Prisijungimo langas

**Administratoriaus vartotojo pagrindinis langas**

Sveiki : Andrejus Berescanskis - Admin						
Naujas Straipsnis	Administravimas	Atsijungti	Rasyti Zinute	Atnaujinti Zinutės		
<b>Priemamu skyriu sarasas</b>	<b>Data</b>	<b>Pavadinimas</b>	<b>Panaudotas</b>	<b>Detaliai</b>	<b>Kelimas</b>	<b>Trinimas</b>
Naujienos	2004.01.10	Pirmas Pirmo Pirmo	False	Daugiau Info	Perkelti	Trinti
Mobilios Technologijos	2004.12.18	reikalavimai magistriniam	False	Daugiau Info	Perkelti	Trinti
Pasaulio ivykiai	1					
<hr/>						
<b>Zinutės</b>						
Sveikas						

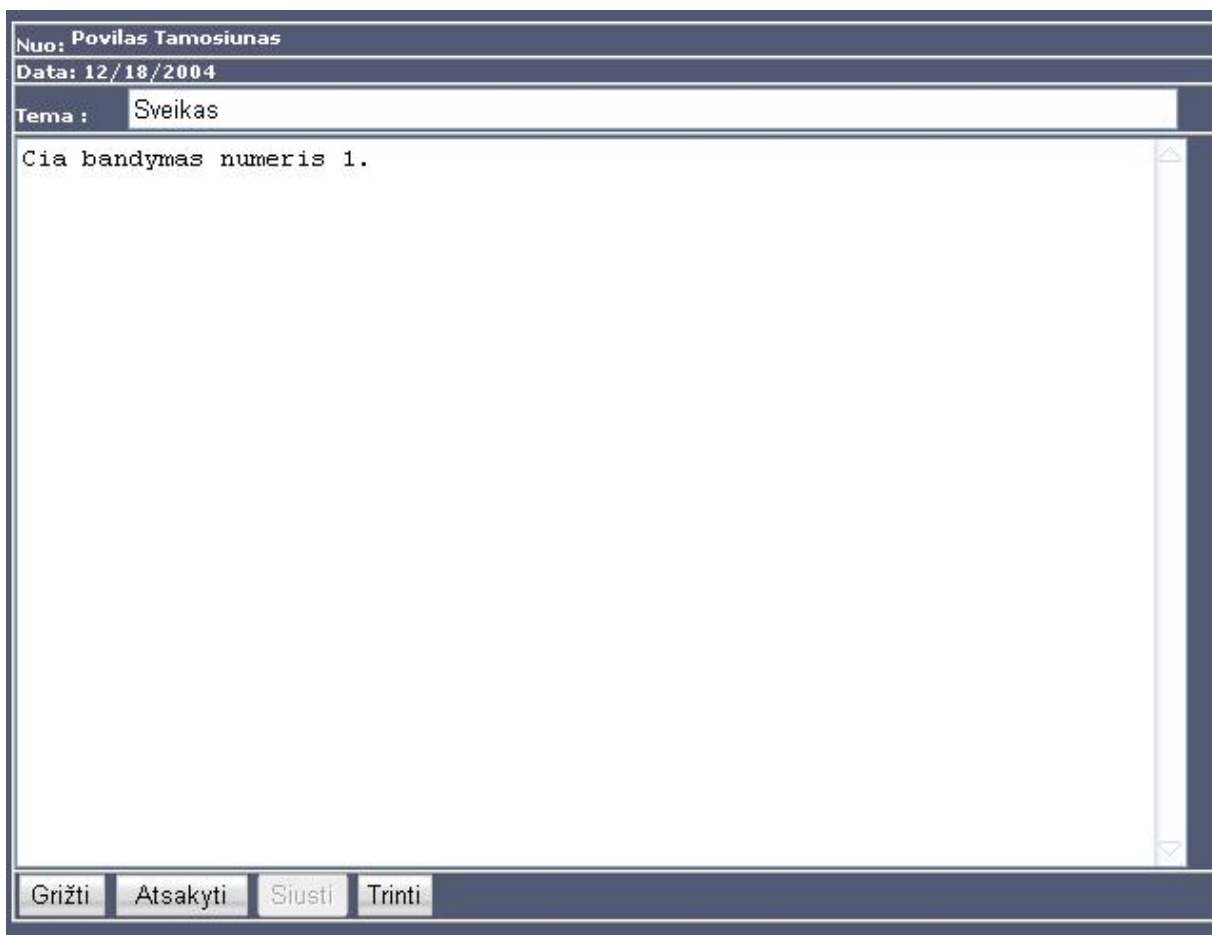
Pav. 11 Pagrindinis langas prisijungus administatoriaus/vyr. redaktoriaus teisem

## Redaktoriaus/Vartotojo pagrindinis langas

Sveiki : Povilas Tamosiunas - Autorius					
Naujas Straipsnis	Atsijungti	Rasyti Zinute	Atnaujinti Žinutės		
Priemamu skyriu sarasas	Data	Pavadinimas	Panaudotas	Detaliai	Kelimas
Naujienos:	2004.01.10	Pirmas Pirmo Pirmo	False	DaugiauInfo	Perkelti
	2004.12.18	reikalavimai magistriniam	False	DaugiauInfo	Perkelti
	1				
<b>Žinutės</b>					
ŽINUČIŲ NĖRA					

Pav. 12 Pagrindinis langas prisijungus vartotojo/skyr. redaktoriaus teisem

**Žinutės skaitymo / atsakymo puslapis**



Pav. 13 Žinutės skaitymo/atsakymo langas

### Naujo straipsnio siuntimo puslapis

Pavadinimas	<input type="text"/>
Įkrovimo skyrius	Naujienos <input type="button" value="v"/>
Byla	<input type="text"/> <input type="button" value="Browse..."/>
Komentaras	<input type="text"/>
<input type="button" value="Pakrauti"/> <input type="button" value="Grizti"/>	

Pav. 14 Naujo straipsnio siuntimo langas

### Straipsnių matomumo administravimo puslapis

<input type="button" value="Atnaujinti"/> <input type="button" value="Grizti"/>		
skyriaus pavadinimas	Straisnis	Matomumas
Naujienos	Pirmas Pirmo Pirmo	✓
Naujienos		✓
Pasaulio įvykiai	Pirmas Trecio Trecio	✓
Naujienos	reikalavimai magistriniam	✓
Mobilios Technologijos	Antras Antro Antro	✓
Mobilios Technologijos	Pirmas Antro Antro	✓
< >		

Pav. 15 Straipsnių matomumo langas

### Vartotojų administravimo puslapis

Naujas	Trinti	Atnaujinti	Grizti	Straipsnių matomumas											
<b>Vartotojų sąrašas</b>				<b>Paradūmimas</b>											
nial danioma divas povtam				Vardas	Andrejus	Skaitymas	<input checked="" type="checkbox"/>	Atnaujinimas	<input checked="" type="checkbox"/>	Rasoti	<input checked="" type="checkbox"/>	Išimimas	<input checked="" type="checkbox"/>	Redaguoti	Ištrinti
				Pavardė	Berescanskis	Mobilios Technologijos	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Redaguoti	Ištrinti			
				Slaptazodis	123	Pasaulio įvykiai	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Redaguoti	Ištrinti			
				E-pastas	nial@senukai.lt	Naujienos	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Atsaukt	Prideti			
				Telefono numeras		<>									
				Parolis	Admin										

Pav. 16 Pagrindinis administravimo langas

### 3.3 Sistemos projektas

#### 3.3.1 Realizacijos modelis



#### Sistemos naudojimo instrukcija.

Prie sistemos prisijungti galima WEB naršyklės pagalba. Suvedus suteiktą vartotojo vardą ir slaptažodį (pav. 10), ir nuspaudus mygtuką “prisijungti”, sistema atidaro pagrindinį puslapį priklausomai nuo jūsų teisių (pav. 11,12). Pagrindinio puslapio mygtukai:

	<ul style="list-style-type: none"><li>• Sistema jums leis nusiųsti naują straipsnį, jeigu jūs turite pakankamai teisių šiai operacijai. Sistema parodys naujo straipsnio sukūrimo langą (pav. 14)</li></ul>
	<ul style="list-style-type: none"><li>• Sistema atidarys administravimo langą. Administravimo lango aprašymas seka toliau.(pvz. 16)</li></ul>
	<ul style="list-style-type: none"><li>• Atsijungti nuo sistemos. Sistema atjungia esamą vartotoją ir grįžta prie prisijungimo lango (pvz. 14)</li></ul>
	<ul style="list-style-type: none"><li>• Sistema atidaro žinutės rašymo langą (pvz. 13)</li></ul>
	<ul style="list-style-type: none"><li>• Jeigu vartotojas pageidauja priverstinai atnaujinti jam skirtų žinučių sąrašą, jis gali pasinaudoti Žinučių atnaujinimo funkcija.</li></ul>

Perėjus prie naujo straipsnio lango suvedame visus reikalingus laukus: Straipsnio pavadinimą, Straipsnio aprašymą ir nurodome straipsnio skyrelį. Atlikus šiuos veiksmus galime spausti mygtuką “Pakrauti”. Straipsnio nusiuntimo mygtukai:

	Sistema pakrauna naują straipsnį į duomenų bazę
--	---



	Grižtama į pagrindinį langą
--	-----------------------------

Jeigu byla buvo nusiųsta sėkmingai, tai atsiranda pranešimas apie sėkmingą duomenų bazės papildymą.

Perėjus prie Administravimo lango (pav. 16) matome vartotojų sąrašą, ir pasirinkus vartotoją iš to sąrašo atsiranda vartotojo detali informacija ir skyrelių matomumo lentelė. Vartotojo valdymo mygtukai:

	Sistema papildomai atidaro lauką "vartotojo vardas". Ir leidžia pildyti vartotojo informacijos laukus.
	Sistema pašalina vartotoją.
	Norėdami tiesiog atnaujinti vartotojo duomenis – pasirenkame vartotoją iš sąrašo, duomenų laukuose pakeičiame duomenis ir spaudžiame "Atnaujinti" mygtuką. Sistema atnaujina vartotojo duomenis.
	Sistema gražina prie pagrindinio lango (pvz. 10,11)
	Pasirinkus vartotoją atsiranda šis mygtukas. Nuspaudus jį, sistema pereina prie Straipsnių matomumo administravimo puslapio (pvz. 14)

Svarbu pastebėti, kad "Straipsnių matomumo" mygtukas neatsiranda kol nėra pasirinktas vartotojas iš vartotojų sąrašo. Pasirinkus vartotoją, galima administruoti jo duomenis, arba skyrelių matomumą. Skyrelių matomumo keitimą pradėdame paspaudus mygtuką "redaguoti" ir užbaigiame nuspaudus "Atnaujinti" arba "Atšaukti". Taip pat matomą skyrelį galima ištrinti iš sąrašo nuspaudus mygtuką "Trinti". Skyrelių valdymo mygtukai:

	Sistema atnaujina duomenis apie skyrelių matomumą vartotojui.
	Sistema anuliuoja bet kokius pakeitimus ir grįžta į pradinę būseną
	Sistema ištrina skyrelį iš skyrelių matomumo sąrašo
	Sistema įjungia redagavimo režimą skyrelių matomumo sąrašo
	Sistema anuliuoja bet kokius pakeitimus ir grįžta į pradinę būseną

	Sistema prideda naują skyrelį į skyrelių matomumo langą
--	---

Nuspaudžius mygtuką “Straipsnių matomumas” pereiname prie straipsnių matomumo nustatymo lango. Atsidarius langui matome visus straipsnius, kurie yra sistemoje (pvz. 14), ir prie kurių parinktas vartotojas turi priėjimą. Vartotojas matys straipsnių sąrašą tik tų skyrelių, kurie yra jam priskirti. Stulpelyje “Matomumas” galime uždėti arba nuimti “varnelę”. Baigus visas korekcijas nuspaudžiame mygtuką “Atnaujinti” ir sėkmingai atnaujinus duomenis sistema gražins pranešimą apie operacijos rezultatą. Lango mygtukai:

	Sistema atnaujina duomenis apie straipsnių matomumą vartotojui.
	Sistema gražina prie administravimo lango (pvz. 15)

Norėdami rašyti naują žinutę, nuspaudžiame iš pagrindinio puslapio meniu mygtuką “Rašyti žinutę”. Sistema atidaro langą, kuriame galime sukurti naują žinutę ir nusiųsti ją kažkuriam iš sistemos vartotojų. Užpildžius žinutės temą ir žinutes teksto laukus spaudžiame mygtuką siusti. Sistema nusiųčia žinutę parinktam vartotojui. Naujos žinutės siuntimo mygtukai:

	Sistema išsiunčia žinutę vartotojui.
	Sistema gražina į pagrindinį langą

Sistemą gali priverstinai atnaujinti žinučių sąrašą jei nuspaudžiame mygtuką “Atnaujinti Žinutės”. Šiuo atveju sistema atnaujina langelį, pagrindiniame lange, kuriame yra žinučių sąrašas. Jei vartotojas žino, kad jis turėtų gauti naujų žinučių, bet kol kas jų nemato, tai nuspaudžius šį mygtuką, jis gali atnaujinti sąrašą priverstinai.

Norėdami perskaityti žinutę, pasirenkame ją iš žinučių sąrašo. Sistema atidaro žinutės atvaizdavimo langą. Iš šio lango galima iškart atsakyti žinutės autoriui nuspaudus mygtuką “Atsakyti”. Žinutę galima pašalinti iš sistemos nuspaudžius mygtuką “Trinti” arba galima grįžti į pradinį meniu, nuspaudus mygtuką “Grįžti”. Žinučių skaitymo lango mygtukai:

	Sistema gražina į pradinį puslapį
	Sistema atidaro puslapį, kuriame galima atsakyti į gautą žinutę ir suaktyvuoja mygtuką Siusti
	Sistema išsiunčia žinutę adresatui
	Sistemą ištrina žinutę

Iš pradinio puslapio (pvz. 10) nuspaudžius mygtuką “Atsijungti” sistema atjungia vartotoją. Ir panaikina jo sesiją. Sistema grįžta į pradinę būseną (pvz. 9) ir išveda prisijungimo langą. Baigus darbą vartotojas turi išsiregistruoti iš sistemos šio mygtuko pagalba. Kilus nesklandumams su programos veikla, galima pasinaudoti mygtuku, kad išsiregistruoti iš sistemos ir vėl prisijungti prie jos tam kad tęsti darbą.

### 3.3.2 Duomenų bazės modelis

Duomenų bazės modelis naudojamas aprašyti atributų tipus – tai aktualu realizuotojui, kuris turi žinoti kokio tipo kintamuosius naudoti manipuliuojant duomenimis su programa ir duomenų baze. Detaliai aprašysime kiekvienos esybės atributus.

#### **tblzines – žinučių saugojimo lentelė**

<b>Pavadinimas</b>	<b>Tipas</b>	<b>Dydis</b>	<b>Reikšmė</b>
userid	bigint	8	Vartotojo kodas
antraste	char	10	Žinutės antraštė
tekstas	Text	16	Žinutes tekstas
zinueid	bigint	8	Žinutės ID
writerid	bigint	8	Vartotojo kuris paraše žinutę kodas
msgread	Bit	1	Ar žinutė perskaityta
Msgdate	Datetime	8	Žinutės data

**tblusers – vartotojų lentelė**

Pavadinimas	Tipas	Dydis	Reikšmė
userid	bigint	8	Vartotojo kodas
username	char	10	Vartotojo prisijungimo vardas
userpass	char	16	Vartotojo slaptažodis
firstname	char	15	Vardas
secname	char	25	pavarde
position	char	50	Pareigos
address	char	50	Adresas
Tel	Char	50	telefonas
posid	int	4	Pareigu ID

**tblstraipsninfo – straipsnio informacijos lentelė**

Pavadinimas	Tipas	Dydis	Reikšmė
straipid	bigint	8	Straipsnio ID
stradata	smalldatetim e	4	Straipsnio patalpinimo data
strpanaudotas	bit	1	Straipsnio panaudojimo požymis
strfailas	char	50	Straipsnio failas(jei prikabintas)
strperkeltas	int	4	Straipsnio perkėlimo požymis
strperkeldata	smalldatetim e	4	Straipsnio kėlimo data
skyriusid	bigint	8	Skyriaus ID
userid	bigint	8	Vartotojo ID
strpavad	char	50	Straipsnio pavadinimas
strinfo	Char	500	Straipsnio detali informacija

**tbluserpositions – vartotojų pareigų lentele**

Pavadinimas	Tipas	Dydis	Reikšmė
posid	Char	20	Pareigos ID
position	int	4	Pareigos pavadinimas

**tblstraipsn – straipsnių lentelė**

Pavadinimas	Tipas	Dydis	Reikšmė
userid	bigint	8	Vartotojo ID
straipid	bigint	8	Straipsnio ID
skyrid	bigint	8	Skyriaus ID
stread	int	4	Skaitymo teisės
strwrite	int	4	Rašymo teisės
strdel	int	4	Trynimo teisės
strupdate	int	4	Atnaujinimo teisės
strmaster	int	4	Straipsnių savininkas

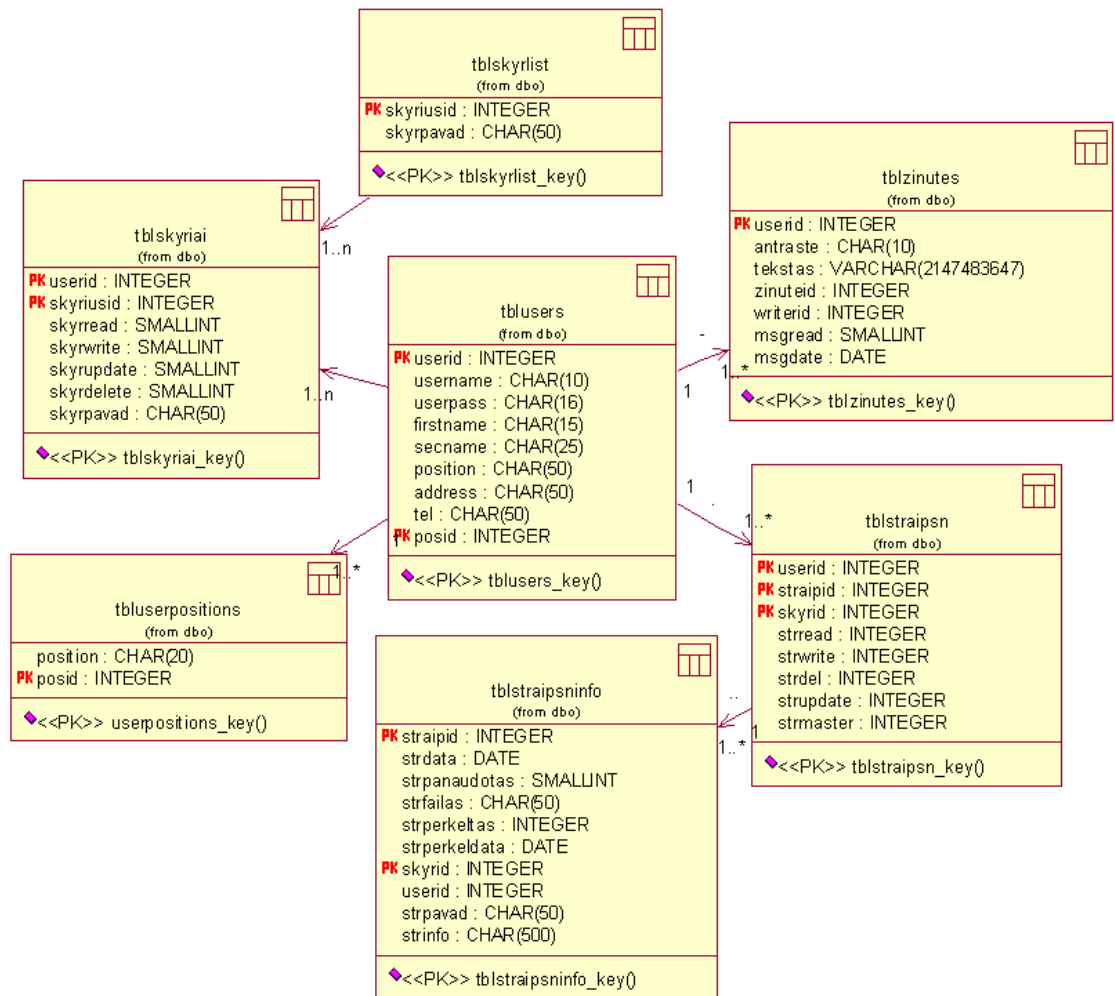
**tbluserpositions – vartotojų pareigų lentelė**

Pavadinimas	Tipas	Dydis	Reikšmė
skyriusid	bigint	8	Skyriaus ID
skyrpavad	Char	50	Skyriaus pavadinimas

**tblstraipsn – straipsnių lentelė**

Pavadinimas	Tipas	Dydis	Reikšmė
userid	bigint	8	Vartotojo ID
skyriusid	bigint	8	Skyriaus ID
skyrread	bit	1	Skaitymo teisės
skyrwrite	bit	1	Rašymo teisės
skyrdel	bit	1	Trynimo teisės
skyrupdate	bit	1	Atnaujinimo teisės
skyrpavad	char	50	Skyriaus pavadinimas

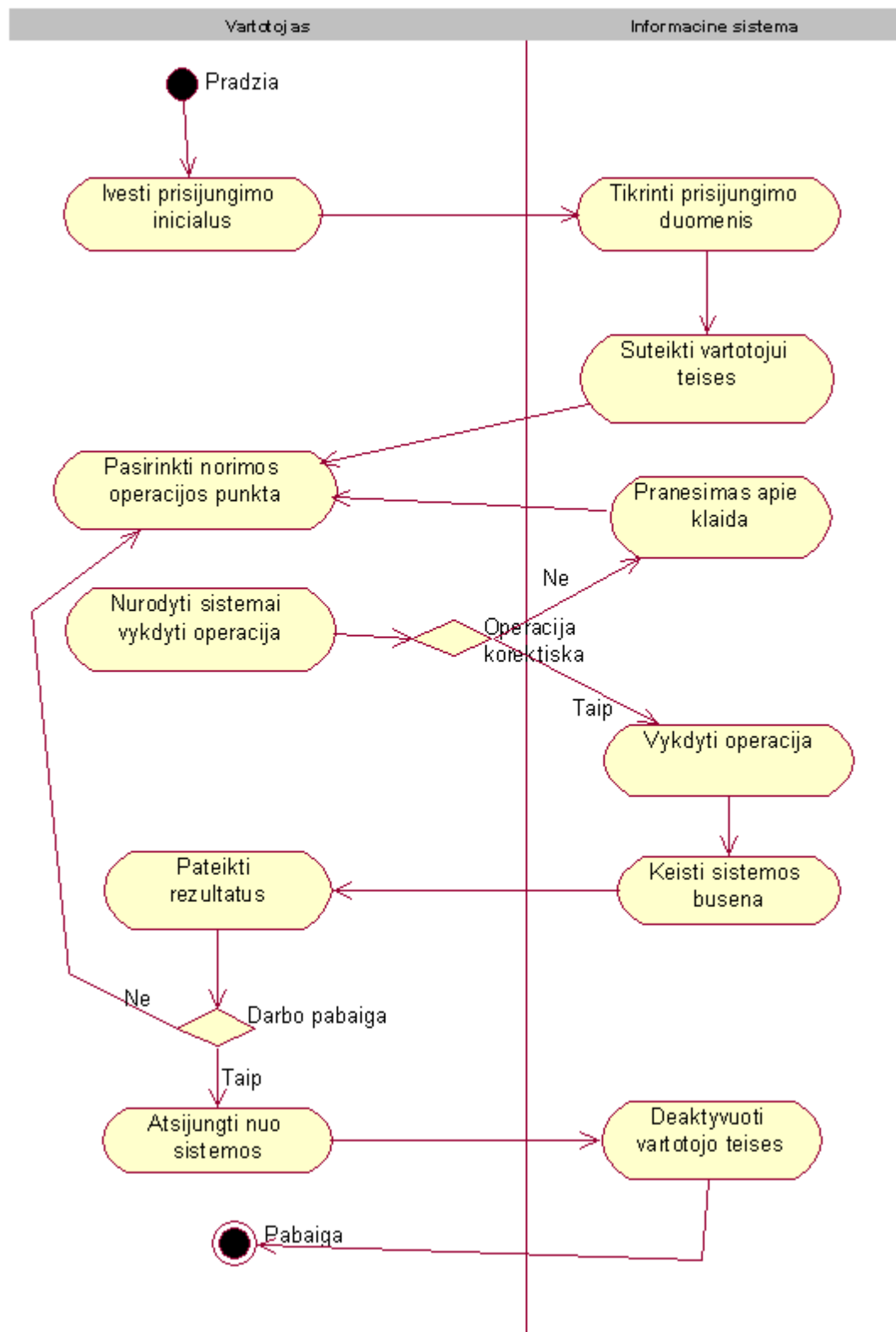
## Duomenų bazės schema



Pav. 17 Duomenų bazės schema

### 3.3.3 Sistemos elgsenos modelis

Sistemos elgsenos modelis naudojamas sistemos elgsenai nusakyti. Šis modelis vaizduoja, kaip sistema reaguos į tam tikrus veiksmus ir kokia bus veiksmų seka. Modelio kūrimui naudojamas IBM Rational programinis paketas Rational Rose Enterprise Edition. Sistemos elgsena nagrinėjama nuo aukščiausio lygmens komandų. Pagrindinis sistemos elgsenos modelis pavaizduotas 18 paveikslėlyje.





Pav. 18 Pagrindinis sistemos elgsenos modelis

### 3.3.4 Sistemos naudojimo instrukcija

Pradedant darbą su informacijos surinkimo sistema, vartotojui reikia interneto naršyklėje paleisti pagrindinę vykdomąją rinkmeną *login.aspx*. Sistema prašo įvesti vartotojo vardą ir slaptažodį. Įvedimas patvirtinamas spragtelėjus pele ties mygtuko *Prisijungimas*. Įvedus neteisingą vardą arba/ir slaptažodį, leidžiama pakartoti prisijungimą.

Prisijungus prie sistemos vartotojui pateikiami 4 meniu juostos punktai *Naujas Straipsnis, Atsijungti, Rašyti Žinutę, Atnaujinti Žinutes*. Prisijungus administratoriaus teisėmis matomi 5 punktai *Naujas Straipsnis, Atsijungti, Rašyti Žinutę, Atnaujinti Žinutes, Administravimas*. Meniu punktai pasirenkami pelės pagalba arba TAB bei ENTER klavišais. Pasirinkus meniu punktą pateikiama forma. Norint įvykdyti atitinkamą komandą, reikia nuspausti formos mygtuką su pele arba ENTER klavišu. Norint pasirinkti kitą pagrindinio meniu punktą, reikia uždaryti formą nuspaudžiant formos meniu mygtuką - *Atgal*. Vartotojas apie tai informuojamas pranešimu ekrane.

Darbas su sistema baigiamas uždarius interneto naršyklę arba nuspaudžius iš pagrindinės formos meniu *Atsijungti*.

### 3.3.5 Informacinės Sistemos diegimo planas

Aptarsime kokias priemones naudosime pačiai sistemos realizacijai. Kaip buvo minėta aukščiau, kad naudosime kaip duomenų bazę, savo pačių sukurtą reliacinę duomenų bazę. Tam kad sukurti jos struktūras panaudosime Microsoft Enterprise Manager.

Duomenų bazės programinę įrangą naudosime: darbinėje stotyje – MS SQL Server 2000. Darbinės stoties duomenų šaltinis gali būti naudojamas, esant reikalui, įdiegus programinę įrangą kitame kompiuteryje – galima bus naudotis duomenimis bet kuriame kompiuteryje kur yra įdiegta duomenų bazė

Realizuot vartotojo sąsają, bei realizuoti programinę dalį naudosime Microsoft Visual Studio .NET 2003 Profesional programinį. Šį įrankį pasirinkome todėl, kad yra reikalaujama, kad sistema būtų realizuota per WEB sąsają. Microsoft Visual Studio paketas turi visas galimybes jungtis prie reliacinių duomenų bazių šaltinių, prie XML šaltinių, taipogi naudoti Microsoft OLAP funkcijas. Todėl iškilus arba pasikeitus ateityje funkciniais reikalavimams, keisti sistemos architektūrai neturėtų iškilti jokių problemų.

## Grafiškai atvaizduotą sistemos diegimo planas

Sukurta sistema turi būti diegiama keliais etapais. 1-as etapas – reikiamų programinių priemonių įdiegimas; 2 – as etapas – sistemos konfigūravimas ir paruošimas darbui; 3 – ias etapas – personalo apmokymas.

Reikalinga įdiegti šias programines priemones:

- IIS web serveris
- .NET Framework
- Microsoft 2000 SP 3 duomenų bazių serveris
- Interneto naršyklė (vartotojų kompiuteriuose)

Įdiegus IIS web serverio ir .NET Framework programines priemones, reikia suderinti

IIS web serverį su ASP.NET puslapių interpretatoriumi. Suderinimo aprašymą galima rasti Microsoft MSDN tinklalapyje [8].

Microsoft Server 2000 duomenų bazių tarnybenyje stotyje reikia sukurti duomenų bazę *mag0305nial*. Joje sukurti lentelės (pav. 17). Lentelių ir duomenų bazės kurimo principus galima rasti MS SQL Server aprašyme [7].

Įdiegus programines priemones, sistemos administratoriui reikia sukurti sistemos vartotojus ir darbo grupes, bei priskirti jiems atitinkamas teises sistemoje.

Organizacijos personalas turi būti supažindintas su sistema, apmokytas naudotis sistemos funkcijomis, racionaliai ir pagal paskirtį naudotis sistemos teikiamomis galimybėmis. Visi vartotojai turi būti supažindinti su organizacijos saugumo politika, žinoti savo vykdomą rolę organizacijoje ir atsakomybę.

## **6 Reikalavimai sistemos funkcionalumui**

Šiame etape apibrėžiami reikalavimai sistemos atliekamoms funkcijoms ir savybėms.

- Tinkamumas sprendžiamai problemai

Sistema turi spręsti analizės dalyje įvardintas problemas ir uždavinius: dokumentų saugojimą, informacijos paiešką, žmogiškojo faktoriaus įtakos mažinimą, sistemos vartojimo paprastumą.

- Rezultatų korektiškumas

Sistemos darbo funkcijos ir rezultatai – informacijos paieška, dokumentų saugojimas, saugumo politika – turi būti vykdomi tiksliai, be klaidų. Procesų rezultatai turi būti pilni, išbaigti.

- Saugumas

Sistemoje turi būti realizuota tokios saugumą užtikrinančios priemonės ir technologijos: sistemos vartotojų identifikavimas ir autentifikavimas, sistemos vartotojų teisės, dokumentų saugumo žymės. Ryšio saugumui padidinti galima vartoti duomenų šifravimą. Duomenų šifravimą realizuoti 3DES algoritmu

- Užbaigtumas

Visos sistemos posistemės turi būti išbaigtos, t.y jos turi pilnai ir korektiškai atlikti savo funkcijas ir uždavinius. Jų rezultatai turi būti korektiški ir nedviprasmiški.

- Atstatomumas

Turi būti galimybė atstatyti paskutinę gerą sistemos būseną, duomenis (duomenis apie vartotojus, sistemą, dokumentus).

- Klaidų tolerancija

Sistema turi tikrinti vartotojo įvedamus duomenis. Ji turi neleisti įrašyti klaidingų duomenų. Vartotojui įvedus klaidingus duomenis, sistema turi pranešti apie klaidą ir leisti ją ištaisyti.

- Patogumas

Sistemos atliekamos funkcijos turi palengvinti vartotojo darbą (automatizuoti tam tikrus procesus, mažinti žmogišką faktorių). Vartotojo grafinė sąsaja turi būti paprasta, logiška, intuityvi, lengvai suprantama.

- Efektyvumas

Sistema turi atlikti jos vykdomas funkcijas greičiau, negu tai būtų atliekama be sistemos, ir kad jos atliekamos funkcijos ir jų atlikimo sparta būtų realiai naudingi sistemos vartotojams. Sistemos reikalaujami techniniai ir žmogiškieji išteklių turi būti kuo mažesni. Sistemos teikiama ekonominė nauda turi būti didesnė už reikalingus techninius ir žmogiškuosius išteklius

- Sistemos priežiūra

Sistema turi leisti analizuoti jos darbą ir naudojamus resursus. Sistema turi turėti galimybę atnaujinti programą neprarandant duomenų ir sistemos funkcionalumo ilgesniam laikui.

- Perkeliamumas

Sistema turi leisti perkelti duomenis ir pačią programą iš vienos palaikomos techninės ir programinės platformos į kitą. Perkėlimas neturi reikalauti didelių ekonominių, techninių, žmogiškųjų ir laiko resursų.

--

## 4. Eksperimentinis tyrimas

### 4.1 Sukurtos sistemos kokybės tyrimas

Sukurta sistema turi atitikti reikalavimus apibrėžtus 2 punkte.

Sistema turi veikti per web sąsają, kas leistų lengvai prieiti prie sistemos iš bet kurios vietos, kur yra naršyklė ir interneto ryšis. Projektas realizuotas naudojant APS . NET , ADO.NET technologijas ir naudoja MSSQL 2000 darbinę stotį. Tai atitinka reikalavimo dėl web sąsajos. Naudojant IIS(Internet Information Server) ir .NET framework prie sistemos galima prisijungti naudojant internetą ir standartinę WEB naršyklę (pvz. 17).

Pvz.19 Prisijungimas prie sistemos naudojant naršyklę

Prisijungus prie sistemos iškart patikrinamos vartotojo teisės, ir pagal jas sudaromas vartotojo meniu. Viena iš realizuotų užduočių yra vartotojų administravimo sistema. Naudojant ją galima priskirti vartotojams teises į skyrius, jų matomumą. Taip pat gali priskirti vartotojams jų matomiems sistemoje skyriaus straipsnių matomumą. Pavyzdžiui galima priskirti vartotojui kažkokį skyrių, bet nuimti teises ką nors jame koreguoti, arba matyti tam skyriui priklausančius straipsnius. Ši sistema atitinka reikalavimuose nurodytą vartotojų hierarchiją. Teisių administravimo sistemai sukurta sąsaja, kuri užtikrina vartotojų administravimą žiūrint į jį iš įvairių aspektų. (pvz. 18)

Pvz.20 Vartotojų ir jų teisių administravimo sistema naudojant WEB naršyklę.

Tik vyriausias redaktorius gali priskyrimėti skiltys skilčių redaktoriams. Vartotojai nemato straipsnių jeigu jie nėra autorizuoti sistemos. Taip pat galima uždrausti matyti visą skyrių. Sistema leidžia kurti ir trinti vartotojus. Vartotojai su Administratoriaus arba Vyr. Redaktoriaus teisėmis galės sukurti, panaikinti arba atnaujinti vartotojo duomenis. Paprastiems tipo vartotojams Autorius arba Redaktorius savo duomenų ir straipsnių, arba skyrių matomumą keist negalima - jie net nemato tokio meniu savo pradiniam puslapyje.

Sistema turi funkcionalumą užtikrinanti duomenų apsikeitimą tarp vartotojų. Sistema leidžia atsisiusi duomenų bylą iš darbinės stoties arba nusiųsti ją į darbinę stotį. Pirmuoju atveju vartotojas, kuris turi teisę matyti straipsnį, gali parsisiųsti sau prie to straipsnio antraštes, priskirtą bylą. (pvz.20). Jeigu byla nebuvo prisegta, tai failo parsisiuntimo galimybė bus neaktyvi.

Taip pat sistema leidžia vartotojui nusiųsti bylą iki 40 mb. į darbinę stotį. (pvz. 21) Kaip pavyzdį reikia pateikti situaciją, kai Autorius parašius straipsnį, turi jį nusiųst į sistemą. Šiuo atveju jis pasirenka skiltį, kuriai tas straipsnis bus skirtas iš sąrašo skilčių, kurios jam leistos pagal teises ir įvedus trumpą parašymą bei straipsnio antraštę, išsiunčia šį straipsnį į darbinę stotį. Po straipsnio išsiuntimo įrašas apie straipsnį įsirašo i DB darbinę stoti, o pati byla patalpinama į darbinėje stotyje skirtą straipsniams vietą. Nusiųntus straipsnį vartotojui, pranešama apie siuntimo būseną ir leidžiama grįžti atgal į pagrindinį meniu. Naujas straipsnis atsiranda vartotojo matomų straipsnių sąrašė. Bet niekas kitas be jo ir skyriaus bei vyr. redaktoriaus neturi teises jo matyti. Norint priskirti straipsnio matomumą kitiems autoriams reikia suteikti jiems skaitymo teises ant to straipsnio.

Pvz.21 Peržiūra straipsnio antraštės ir failo parsiumimas.

Pvz. 22 Naujo straipsnio įvedimas.

Išvados:

- Sistema užtikrina duomenų apsikeitimą tarp jos vartotojų.
- Sistema užtikrina vartotojų administravimą ir duomenų apsaugą.
- Prisijungimas prie sistemos vykdomas per WEB sąsają

#### **4.2 Tolimesnio sistemos tobulinimo, plėtojimo galimybės**

Sistema bus plėtojama ir toliau. Bus pridėtos papildomos funkcijos tokios kaip detalaus užsakyto straipsnio aprašymas. Naujų straipsnių įvedimas ir užsakymas pagal poreikį, kuris bus realizuojamas automatiškai tokiam poreikiui atsiradus . Sistema turės galimybes užsiųsti laikmeną su programą arba kokia nors papildoma medžiaga ir patalpinti ją į laikmenų darbinę stotį, taip pat surišti naują laikmeną su jau esančių straipsnių. Tai bus realizuota naudojant .NET technologijas leidžiančias lengvai naudotis tokiomis galimybėmis. Planuojama įvesti ir vartotojo sąsajos derinimo priemones, kur kiekvienas vartotojas galėtų susiderint savo virtualiąją darbinę aplinką taip, kaip jam yra patogiau. Tam bus sukurtos taip vadinamos valdymo lentelės, kuriuose bus saugomi parametrai kiekvieno individualaus vartotojo ir programos formos naudosis tais parametrais, kad keistų savo išvaizdą. Tai nebūtinai praplėtimas, bet šį modifikacija galės žymiai palengvinti darbą. Tolimesnius praplėtimus galima bus atlikti, kai užsakovas juos suformuluos.





## 5. Išvados

Sistema realizuota naudojant .NET ir MS SQL Server 2000. Šios technologijos užtikrina pilną sistemos funkcionalumą, praplečiamumą ir stabilumą. Visas išeities kodas suprogramuotas naudojant C# programavimo kalbą. Vartotojo sąsaja suprogramuota naudojant ASP programavimo kalbą. Duomenys talpinami MS SQL 2000 darbinėje stotyje. Sistema sudaryta iš 3 dalių. Vartotojo dalys – vartotojo sąsaja. Aptarnaujanti dalis – web service. Duomenų saugykla – MS SQL 2000 Server darbinė stotis. Kiekvieną iš šių dalių galima naudoti skirtinguose darbinėse stotyse, nuo to sistemos funkcionalumas nepasikeičia, tačiau tai gali neigiamai įtakoti visos sistemos veikimo spartą.

Sistema atitinka kaip funkcinį taip ir nefunkcinį bei saugumo reikalavimų, pateiktų 3.12 reikalavimų modelio skyrelyje. Sistema užtikrina saugų duomenų perdavimą, dėl to, kad naudojama 3DES KEY kriptavimo technologija duomenų srautams siūsti ir gauti tarp vartotojo dalies ir web servise dalies. Duomenų perdavimas tarp Webservice ir MS SQL Server darbinė stoties nėra apsaugotas. Dėl to idealus sistemos funkcionavimo variantas būtų toks, kad Webservice ir MS SQL Server veiktų vienoje darbinėje stotyje saugumo sumetimais.

Suformuluoti kriterijai, kuriuos turėtų tenkinti efektyvi ir naudinga periodinio leidinio sudarymo web aplinkoje sistema. Ją projektuojant panaudoti kitų sistemų privalumai bei sukurta šiuos privalumus panaudojanti, bei praplečianti architektūra.

Praktinių bandymų metu paaiškėjo, jog pasiūlyta sistema efektyviai susidoroja su informacijos valdymo uždaviniais bei tenkina visus realizacijai ir veikimui numatytus reikalavimus. Kaip ir buvo tikėtasi, automatizavus informacijos valdymo procesą, apdorojamos informacijos kiekis bei jos atnaujinimo operatyvumas smarkiai išauga, bei sumažėja vartotojo sugaištas laikas tam tikriems uždaviniams atlikti.

Nebuvo pastebėta jokių trūkumų dėl naudojimo .NET platformos. Visi uždaviniai greitai ir efektyviai realizavosi naudojant Microsoft kompanijos žinių bazę. Kiekvienam uždaviniui spręsti buvo randami daugiau negu vienas sprendimai

Net ir dabartinio sistemos varianto panaudojimas, leidiniais užsiimančios įmonės veiklos procese, žymiai padidina jos veiklos efektyvumą bei suteikia ekonominę naudą. Sistema pritaikyta pateikti informaciją vartotojui internete, todėl gali būti pliečiama ir pildoma

tolimesniu funkcionalumu.

## 6. Literatūros sąrašas

Buvo pasinaudota laboratorinių darbų aprašymu „Darbas su informacija duomenų bazėse ASP skriptų pagalba“.

[1] Scott Short, Building XML Web Services for the Microsoft .NET Framework, 2002.

[2] Microsoft Publishing, Implementing Microsoft® Windows® 2000 Professional and Server.2002.

[3] Kornis Olly, VB.NET ir ASP.NET Pagrindai, Lory, 2003.

[4] ASP.NET Scripts [http://www.stickysauce.com/scriptdirectory/html/ASP\\_Scripts/](http://www.stickysauce.com/scriptdirectory/html/ASP_Scripts/)

[5] ASP.NET QuickStart tutorial, <http://www.dotnetjunkies.com/quickstart/aspplus/>

[6] Simon Robinson, K. Scott Allen, Ollie Cornes, Jay Glynn, Zach Greenvoss, Burton Harvey, Christian Nagel, Morgan Skinner, Karli Watson, Professional C# Second edition, 2003

[7] MS SQL Server Specification

[8] Configuring an ASP.NET Application for an ASP.NET Version

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpguide/html/cpconconfiguringaspnetapplicationforaspnetversion.asp>

## 7. Terminų ir santrumpų žodynas

**DB** (Database) – duomenų bazė.

**SQL** (Structured Query Language) – struktūrizuota užklausų kalba, skirta manipuluoti reliaciniais duomenimis.

**UML** (Unified Modelling language) – notacija, skirta projektuoti ir aprašyti programinę įrangą.

**XML** (eXtensible Markup Language) – duomenų aprašymo standartas, pripažintas 1998 m.

**HTML** (Hypertext Markup Language) - hiperteksto paruošimo kalba. Tai standartinė tinklalapių turinio ir išvaizdos aprašymo kalba. Ja apibrėžiama tinklalapio struktūra ir ryšiai.

**RUP** (Rational Unified Process) – sistemos kūrimo metodas.

**Eng** – vertimas iš anglų kalbos

**HTTP** (Hyper Text Transfer Protocol) – protokolas, kuriuo perduodama informacija tarp serverio ir interneto naršyklės

**OS** – operacinė sistema

**IS** – informacinė sistema

**WEB** – interneto terė

## 8. Santrauka Anglų kalba

### Summary

In our days every part of business goes on-line. Using old methods of information gathering means that your business might be left behind or maybe even be at risk of extinction. Information gathering in publishing business is a critical task. Information and media gathering and storing is a task of highest importance. Computerizing this task means that you can have more data in less time. Data must be gathered anytime, anywhere. To accomplish this critical goal online data management systems are created. Every data management system must have data gathering system connected to it.

In this paper document gathering system conception, problems and realization methods are analyzed. Document uploading system is an "entry point" for data management systems. Document gathering system lets us accomplish document uploading, storing and management tasks. Such system consists of three parts.

Document or media uploading task:

- uploading or downloading information
- grouping and classifying information that is going to be uploaded
- managing information that is uploaded to server

System user management

- creating/deleting/modifying user information
- assigning user permissions for various operations

User communication

- Writing and sending short messages to other system users
- reading and replying to messages received from other users
- deleting unwanted messages from system

All of these parts have to be realized in order for system to work properly and to provide maximum use of it.

## **9. Priedai**

Kompaktinis diskas su:

- Programo išeities kodu
- Programos duomenų baze
- Sistemos dokumentacija