



KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

INFORMACIJOS SISTEMŲ KATEDRA

Kęstutis Vanagas

IT žinių portalo reitingavimo modelis

Magistro darbas

Darbo vadovas

prof. Rimantas Butleris

Kaunas, 2007



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
INFORMACIJOS SISTEMŲ KATEDRA

Kęstutis Vanagas

IT žinių portalo reitingavimo modelis

Magistro darbas

Recenzentas

2007-05-28

doc. dr. Aleksas Riškus

Vadovas

2007-05-28

prof. Rimantas Butleris

Atliko

2007-05-28

IFM-1/2 gr. stud.
Kęstutis Vanagas

Kaunas, 2007

Turinys

1	Įvadas	5
2	Egzistuojančių reitingavimo technologijų analizė	9
2.1	Tikslai, tyrimo sritis, objektas ir problema	9
2.1.1	Darbo tikslas	9
2.1.2	Analizės tikslas	9
2.1.3	Tyrimo sritis	9
2.1.4	Sprendžiama problema	10
2.2	Reitingavimo technologijų aplinkos analizė	10
2.2.1	Reitingavimo posistemio kontekstas	10
2.2.2	Reitingavimo posistemio naudotojai	11
2.3	Egzistuojantys sprendimai reitingų skaičiavimo problemai spręsti	12
2.3.1	Tiesinio reitingavimo sistemos	12
2.3.2	Vidurkio principu grindžiamas reitingavimas	14
2.3.3	Konsensuso(pastovumo) prioritetų reitingavimo schemas	16
2.3.4	ELO reitingavimo sistema	17
2.3.5	Glicko reitingavimo sistema	20
2.3.6	Glicko-2 reitingavimo sistema	21
2.4	Analizės išvados	22
3	Reitingavimo metodo formali koncepcija	24
3.1	Reitingų skaičiavimo prototipas	24
3.1.1	Bendras reitingų skaičiavimo algoritmas	24
3.1.2	Reitinguojamų objektų aibės išskyrimas	28
3.1.3	Autorių reitingavimo savybės	29
3.1.4	Straipsnių reitingavimo savybės	30
3.2	Duomenų saugojimo struktūra	31
4	Eksperimentinis reitingavimo modelio tyrimas	36
4.1	Panaudos atvejai	36
4.2	Architektūrinio išdėstymo vaizdas	37
4.3	Administratoriaus sąsaja	38
5	Pasiūlytojo reitingavimo modelio galimybių palyginimas su kitais modeliais	47
6	Išvados	48
7	Naudojamos literatūros sąrašas	49
8	Priedai	50
	Literatūros sąrašas	55

Internet news portal ranking model

Summary

Almost all internet portals has it's own user ranking systems. In this way all users which participate in this portal life gather meaningful units or points. This document's investigative work is to create computer-assisted and easily expandable internet news portal ranking control process and it's possible optimal implementation algorithm. This algorithm should not load system with difficult computation. In this document we examine available ranking methods and algorithms and ways how they work. Those methods involve linear, mean value, Elo, Glicko and other ranking models. In these models we examine all advantages and disadvantages they have. In this document we describe ranking model that has wide adaptability and that involves consumer in active embarrassment. Architectural concept part describes ranking calculation process, formally define algorithms and actions. For acceptable flexibility we use weight coefficients for rating units. This proposed method is provided with experimental prototype.

1 Įvadas

Pastaruoju dešimtmečiu labai sparčiai pradėjo vystytis internetinės technologijos. Pradžioje internetas buvo daugumai tik prabanga, o jame pateikta informacija tai tik konkretizuotos duomenų struktūros. Dabar interneto panaudojamumui apibrėžti neužtektų paprasto išsireiškimo. Šiuo metu jame galima atlikti bene visus į galvą šaunančius veiksmus ir netgi rasti visai neįsivaizduojamos pakraipos informacijos. Kaip bebūtų, visą informaciją galima dalinti ir klasifikuoti didesniais ar mažesniais idėjiniais vienetais. Mažiausias konkretus idėjinis internetinis vienetas yra viena internetinė svetainė, portalas. Čia dažniausiai būna išreikštos bei patalpintos vieno žmogaus, grupelės ar organizacijos surinktas konkrečios srities turinys, kuris atitinkamai turi savo paskirtį. Taip pat kiekviena svetainių turi savo lankytojų aibes, kurių pritraukimas yra vienas iš pagrindinių internetinių puslapių poreikių. Viliojimo veiksmus organizatoriai atlieka taip kaip jiems atrodo geriausia, tačiau pagrindiniai būdai vyrauja visur – tai ir vartotojo sąsajos patogumas, ir pateikiamos informacijos patrauklumas, pilnumas, įdomumas, ir veiksmų atlikimo spartumas, ir reklamų bei pasiūlymų protingas dozavimas ir daug kitų bendrai naudingų paslaugų teikimas.

Psichologiškai žmogus dažniausiai gerai ir saugiai jaučiasi, kai su juo yra bendraujama ir rodomas atgalinis ryšys jo atliktiems veiksmams. Daugelis internetinių svetainių šį vartotojų pritraukimo variantą bando atlikti įvairiais būdais rodydami gyvą ar automatizuotą dėmesį. Gyvam dėmesiui pateikti reikalingi brangūs resursai – žmonės, kurie ne visada gali budėti ir teikti reikiamas paslaugas. Kita vertus, internetinės svetainės yra globaliniame tinkle, kuriame vartotojai gali būti aktyvūs ištisą parą, priklausomai nuo poreikio ir geografinės buvimo vietos.

Labiausiai dalyvavimui vartotojus skatinantis veiksnys yra sąmoningas ar nesąmoningas varžymasis tarpusavyje – lenktyniavimas. Kiekvienas žmogus save nori pateikti labiau geresniais už kitus. Dėl šių priežasčių dauguma internetinių svetainių įdiegia savo vartotojų vertinimo reitingavimo sistemas. Tokiu būdu kiekvienas dalyvis savo sąskaitoje kaupia vienokius ar kitokius svertinius vienetus ar taškus. Išpopuliarėjus šiai vartotojų pritraukimo technologijai, svetainių kūrėjai pradėjo ieškoti vis tobulesnių ir geresnių reitingavimo algoritmų. Beje, kiekvienai probleminei sričiai negali būti pritaikomas vienas ir tas pats reitingavimo algoritmas.

Šio darbo tyrimo objektas yra automatizuotas ir nesudėtingai išplečiamas internetinių žinių svetainių reitinguojamų elementų valdymo procesas bei galimas optimalus jo realizavimo algoritmas.

Nagrinėjant termino **žinios** prasmę, galima sakyti, jog tai yra organizuotos informacijos visuma bei sąryšiai tarp jos, kurių pagrindu gali veikti koks nors asmuo arba sistema. Žinios reikalauja tvirto pagrindimo ir paaiškinimo. Viena svarbiausių kriterijų – galimybė jas patikrinti bei pakartotinai surasti [16]. Mūsų nagrinėjamoje probleminėje srityje žinios labiau koncentruojasi į recenzuotų straipsnių aibę. Recenzentas yra kompetentingas asmuo, todėl pro jo rankas praėjusio straipsnio turinys gali būti laikomas pagrįstu ir informatyviu, taip kaip reikalauja žinių sąvoka.

Atitinkamai, **reitingavimas** yra individų ar grupių pozicionavimo procesas, nurodantis jų vietą eilėje ir ryšius su kitais objektais [10]. Internetinio portalo reitingavimas naudojamas pozicionuojant jame dalyvaujančius objektus, tokius kaip straipsniai, rašytojai, skaitytojai.

Programų inžinerijoje **išplečiamumas** (ang. extensibility) yra sistemos kūrimo principas, kurio įgyvendinime numatomas ateities savybių augimas. Tai yra sistemos matas nusakantis galimybę sistemos išplėtimui bei nusakantis pastangų lygį, kurio reikės praplėtimui realizuoti [5]. Internetiniame žinių portale reikalingas išplečiamumas atsiradus poreikiui sukurti naujus reitinguojamus vienetus.

Darbo tikslas yra suformuoti ir plėtoti reitingavimo sistemos modelio galimybių ribas, jas maksimaliai automatizuojant ir priartinant prie žmogaus kaip vartotojo pritarimo ribos.

Aprašomame darbe bus sprendžiami šie pagrindiniai klausimai:

- Esamų reitingavimo metodų bei algoritmų realizavimo būdai bei analizė;
- Reitingavimo modelio, grindžiamo plačiu pritaikomumu bei vartotojo įtraukimu į aktyvų dalyvavimą, formavimo metodo kūrimas;
- Pasiūlytojo metodo patikrinimas kuriant reitingavimo sistemos prototipą.

Mokslinės problemos esmė yra optimalaus sistemai ir vartotojui reitingavimo apskaičiavimo bei valdymo metodo sukūrimas.

Šiuo metu daugelyje internetinių svetainių naudojami reitingavimo metodai yra tiesiški, ko pasekoje vartotojams tampa visiškai aiškūs jų veikimo principai. Pagrindė visuose portaluose reitinguojami vienetai turi kaupiamąjį taškų algoritmą, kuriuo kiekvienas vartotojas lenktyniauja tarpusavyje vienas su kitu. Tačiau šioje vietoje neižvelgiama daug iškylančių problemų, tokių kaip vartotojų veiksmų skaidrumas bei teisingumas, istoriškai senų vartotojų „taškų kraities“ sukaupimas bei monopolizacija, sistemos išplečiamumas, esminių vienetų panaudojamumo reguliavimas bei kiti iš anksto nesuplanuoti veiksniai.

Paleidžiant į panaudą sistemą visada turi būti galimybė įsikišti ekspertui, kuris galėtų reguliuoti visą bendrą darbą nesugadindamas informacijos ir „nenukriausdamas“ nei vieno

objekto. Ši vieta yra labai jautri ir manipuliuojanti pagrindine vertybe – vartotojo pasitikėjimu sistema. Veikiančios sistemos pagrindinis manipuliavimo vienetas tampa vartotojo ar reitinguojamo vieneto uždirbtas kiekvienas taškas ar įvertinimas, kuris sulyginamas su žmogaus indėliu į darbą ir savo reikšme gali būti laikomas materialia vertybe. Šiuo atveju kiekvienas veiksmas sukėles praradimus gali inicijuoti grandininę vartotojų nepasitikėjimo reakciją ir taip sužlugdyti net ir labiausiai prižiūrimą bei puoselejamą produktą.

Darbų rezultatų mokslinis naujumas siejamas su siūlomu metodu, kuris leidžia kurti nuo probleminės srities nedaug priklausančią loginę reitingavimo struktūrą su galimybe pilnareikšmiškai įgyvendinti vartotojo lūkesčius, leidžiant jam suprasti pagrindines tobulėjimo ir savęs išaukštinimo nišas, bei sukuriant patogią galimybę į visus vykstančius procesus įsikišti probleminės srities ekspertui su apgalvota sąsaja bei logika, kurios variacijos galimybės palieka tas pačias tarpusavio pozicijas visiems reitinguojamiems objektams.

Veikloje yra aprašomas reitingų apskaičiavimo procesas, formaliai apibrėžiami algoritmai bei vykdomi veiksmai. Pats reitinguojamas vienetas išskiriamas kaip atskiras neįpareigojantis sukonkretinimui objektas, kurio naudojimas padeda plėtoti ir suprasti reitingavimo modelio praplečiamumo ir lankstumo galimybes.

Taip pat svarbus akcentas yra tai, jog kiekvienas atskiras naudojamas objektas yra grindžiamas svertinių koeficientų panaudojamumu. Šios metodo ypatybės įgalina lengvai operuoti turimais duomenimis, privedant prie realiai tikėtinų skaitinių reikšmių rezultatų. Siūlomas metodas organizuojamas egzistuojančiais naujais technologiniais ir projektavimo sprendimais, pasitelkiant naujas galimybes. Kaip ir daugelyje didesnės apimties sistemų, duomenys, konfigūracinė informacija, bei kaupiamoji informacija saugoma duomenų bazėse, taip įgalinant greitai, kokybiškai ir operatyviai manipuluoti objektais.

Praktinė darbo svarba. Darbo eigoje suprojektuota pasiūlytąjį reitingavimo modelį realizuojanti programinė įranga. Ji remiasi automatiniu reitingavimo palaikymu, leidžia būti patogiai reguliuojama išorės klientui – reitingavimo sistemos ekspertui. Automatizavimas sumažina laiko sąnaudas atsirandančias dėl galimų klaidingų vartotojo veiksmų.

Reitingavimo sistema nėra kietai įsiūta į visos sistemos programinį kodą, dėl to paliekama galimybė lanksčiam metodikos keitimui. Tai yra dalis reuse (pakartotinio panaudojimo) technologijos.

Darbo struktūra. Analitinėje dalyje aptariama reitingavimo samprata, analizuojami galimi reitingų taškų rinkimo būdai bei egzistuojantys ir internetinėje aplinkoje taikomi reitingavimo metodai.

Koncepcinėje dalyje aprašomas sukurtas metodas, skirtas internetinės svetainės reitingavimo posistemiiui realizuoti. Detaliai specifikuojama reitingų formavimo bei interpretavimo algoritmai.

Ekspertinėje dalyje pristatomas sukurtas reitingų sistemos prototipas.

Darbo pabaigoje pateikiamos galutinės išvados, literatūros sąrašas bei terminų ir santrumpų žodynėlis.

2 Egzistuojančių reitingavimo technologijų analizė

Skyriuje nagrinėsime egzistuojančius reitingavimo sistemų metodus išryškindami jų privalumus bei trūkumus. Pagrindė sutelkiamas dėmesys į plačiausiai naudojamus reitingavimo algoritmus: tiesinis reitingavimas, vidurkio principo skaičiavimai, Elo, Glicko šeimos reitingavimo metodai, konsensuso reitingavimas. Analizuosime kuo šie metodai tinkami mūsų nagrinėjamai sričiai bei ką reiktų iš jų eliminuoti ar sukurti naujo. Bus siūlomi galimi patobulinimai bei sudaromos pradinės specifikacijos.

2.1 Tikslai, tyrimo sritis, objektas ir problema

Šiame poskyryje nustatomi analizės tikslai bei apibrėžiama tyrimo sritis. Taip pat pateikiamas tyrimo objektas bei sprendžiama problema. Tyrimas vykdomas su tikslu patikrinti egzistuojančias sistemas, sužinoti jų panaudojimo galimybes žinių portalo reitingavimo modelyje.

2.1.1 Darbo tikslas

Darbo tikslas yra sukurti reitingavimo [11] sistemos modelį, kuris būtų pritaikytas publikuojamų žinių svetainės struktūrai ir kuris galėtų pagrindinę funkcinę darbo dalį atlikti be žmogaus – eksperto įsikišimo.

Analizuosime egzistuojančius objektų reitingavimo bei vertinimo metodus, tam kad būtų galima rasti optimalų naudojamą įgyvendinimo sprendimą.

2.1.2 Analizės tikslas

Analizės tikslą apibrėšime šiais punktais:

- Žinių portalo reitingavimo apibrėžimas ir palyginimas su įprastine reitingavimo sistema;
- Reitingavimo modelio pritaikymo tam tikrai tiriamai sričiai (žinių portalas) analizė;
- Egzistuojančių reitingavimo sistemų bei modelių analizė.

2.1.3 Tyrimo sritis

Darbe tiriamos galimos reitingavimo sistemos, jų modeliai, panaudojimas viešame tinkle bei kokybiškumo savybės. Ieškoma tokių reitingavimo sistemų, kurios savo idėjine prasme yra pritaikomos internetinių žinių svetainių objektų vertinimo panaudojamumui.

Darbo tyrimo objektas bus:

- Rreitingavimo modelio atitinkančio patogaus valdymo ir panaudojamumo sudarymas;
- Modelio panaudojamumas internetinio žinių portalo aplinkoje.

2.1.4 Sprendžiama problema

Beveik visuose portaluose randama galimybė už kažką balsuoti ir vertinti. Vėliau tie balsai vertintam objektui suteikia paskatinimą ar išimtinumą visų objektų aibėje. Darbe ieškoma metodų kokybiškam bei kiek galima labiau intelektualiam internetinės svetainės reitingavimo algoritmui realizuoti. Daugumoje internetinių svetainių, kuriose yra įdiegta kokios nors rūšies reitingavimo sistema, pastaroji būna labai paprastai realizuota. Paprasta reitingavimo sistema yra nesudėtinga įgyvendinimui, saugojimui bei vartotojo pateikimui, tačiau nesuteikia lankstumo, naudojimo įvairumo bei profesionalumo laisvės.

Kitos svetainės naudoja galingas reitingavimo sistemas, tačiau jose naudojami metodai tinka tik specializuotų objektų varžyboms, kas netinka internetinio žinių portalo reitinguojamiems objektams.

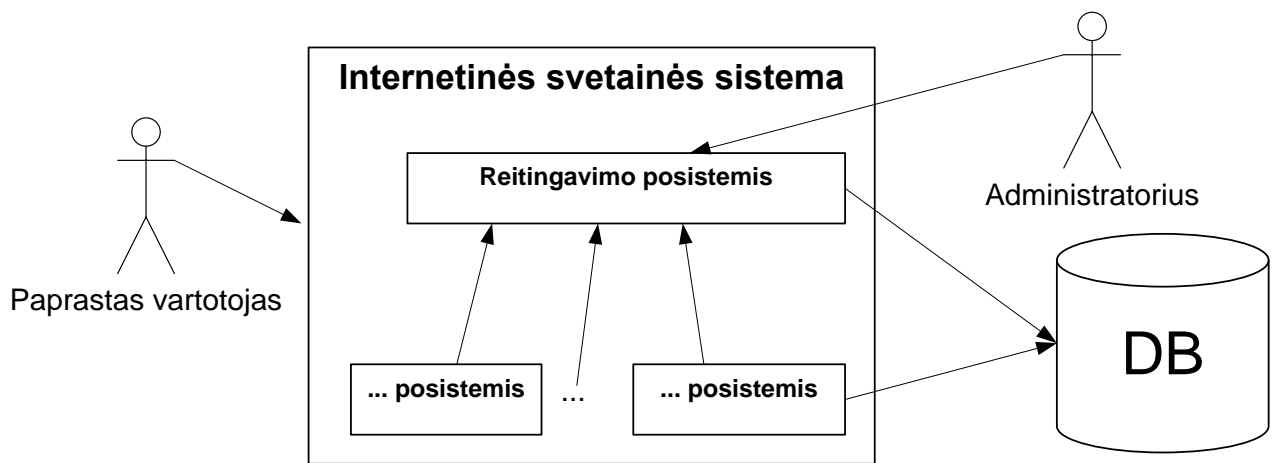
Naudojant sudėtingesnes duomenų bazės struktūras, galima pasiekti žymiai platesnių panaudojimo galimybių. Galima objektizuoti visus reitinguojamus vienetus, taip padidinant panaudojamumo galimybes. Tokiu principu sukurta sistema gali apdoroti nesvarbu kokio tipo reitinguojamus vienetus: vartotojus, straipsnius, reklamas, komentarų rašymus ir t.t.

2.2 Reitingavimo technologijų aplinkos analizė

Darbo eigoje gautas rezultatas bus internetinės sistemos viena iš darbinių dalių, kurios veiksmingumas vartotojo lygyje matomas bus tik minimaliai – jam bus pateikiama tik iš reitingavimo posistemio suskaičiuota skaitinė informacija. Darbinę sąsają su reitingavimo posistemiū turės tik sistemos administratorius – ekspertas, kurio pagrindinė funkcija bus nustatyti pagrindinius pradinius posistemio darbinius parametrus bei konfigūraciją. Tokią aplinką geriausiai bus galima analizuoti į ją žiūrint kaip į „juodąją dėžę“ [4] – operuojant su įėjimo bei išėjimo informacija.

2.2.1 Reitingavimo posistemio kontekstas

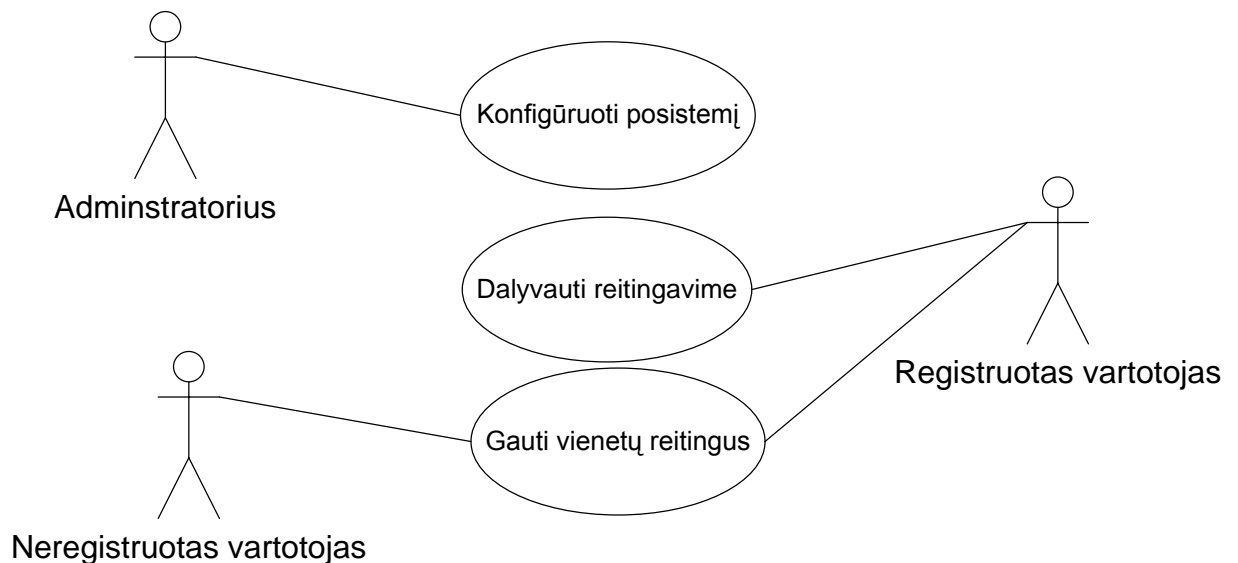
Reitingavimo posistemis savo veiksmingumui užtikrinti nenaudos kitų sistemos posistemų išskyrus bendrąją duomenų bazę. Posistemio veiksmingumas apibrėžiamas reitingo taškų skaičiavimu bei rezultatų gražinimu. Dėl šių savybių pagrindiniai iniciavimo ryšiai ateina iš kitų posistemų **1 pav.**



1 pav. Reitingavimo posistemio kontekstas internetinio portalo sistemoje

2.2.2 Reitingavimo posistemio naudotojai

Visi internetinės svetainės vartotojai parodyti 2 pav. Matoma, kad ne visi iš jų gaus priėjimą prie reitingavimo. Administratorius turi daugiausiai teisių į duomenų keitimą ir sistemos konfigūracijos tvarkymą. Registruotas vartotojas atlieka du pagrindinius veiksmus: reitinguoja ir stebi rezultatus. Pašalinis vartotojas gali tik stebėti rezultatus.



2 pav. Reitinguojamos internetinės svetainės aktoriai

Padetalizuosime atskirus vartotojus ir jų galimybes:

Administratorius – tai sistemos vartotojas/ekspertas, kurio paskirtis nustatyti pradinis sistemos konfigūracijos duomenis taip, kad internetinės svetainės reitingavimo posistemis atliktu visas savo funkcijas. Išskirtinai nei kitiems vartotojams, šiam bus suteikiama galimybė priėti ir keisti sisteminius duomenis.

Registruotas vartotojas – tai internetinio portalo išteklių vartotojas, kuris galės daryti įtaką reitingavimui ir pats būti reitinguojamas.

Neregistruotas vartotojas – tai internetinio portalo vartotojas, kuris neturės teisės daryti įtakos reitinguojamiems vienetams bei pats negalės būti reitinguojamas. Jis galės tik stebėti ir gauti reitinguotų vienetų bendrinius galutinius reitingus.

2.3 Egzistuojantys sprendimai reitingų skaičiavimo problemai spręsti

Šiame skyriuje aptariama internetinių svetainių reitingavimo metodų privalumai bei trūkumai bei bus ieškomas optimalus sprendimas reitingavimo sistemos specifikai nusakyti. Reitinguojamų internetinių svetainių yra daug, tačiau visos jos iš principo naudoja panašius skaičiavimo principus.

2.3.1 Tiesinio reitingavimo sistemos

Didelis šio reitingavimo tipo pranašumas yra tas, jog čia nereikalaujama didelių darbo resursų. Šio metodo metu kiekvienam reitinguojamam vienetui priskiriamas skaitliukas, kuris įtakojamas išorinių veiksnių didinamas atitinkamai pagal atlikto veiksmo stiprumą. Tokia reitingų apskaita neapkrauna procesoriaus bei duomenų bazės darbu.

Veiksmų aibė nusakoma tokiai formule:

$$R_j = R_j + v_{ij} \quad (1)$$

Čia:

i Reitinguojamo elemento identifikatorius

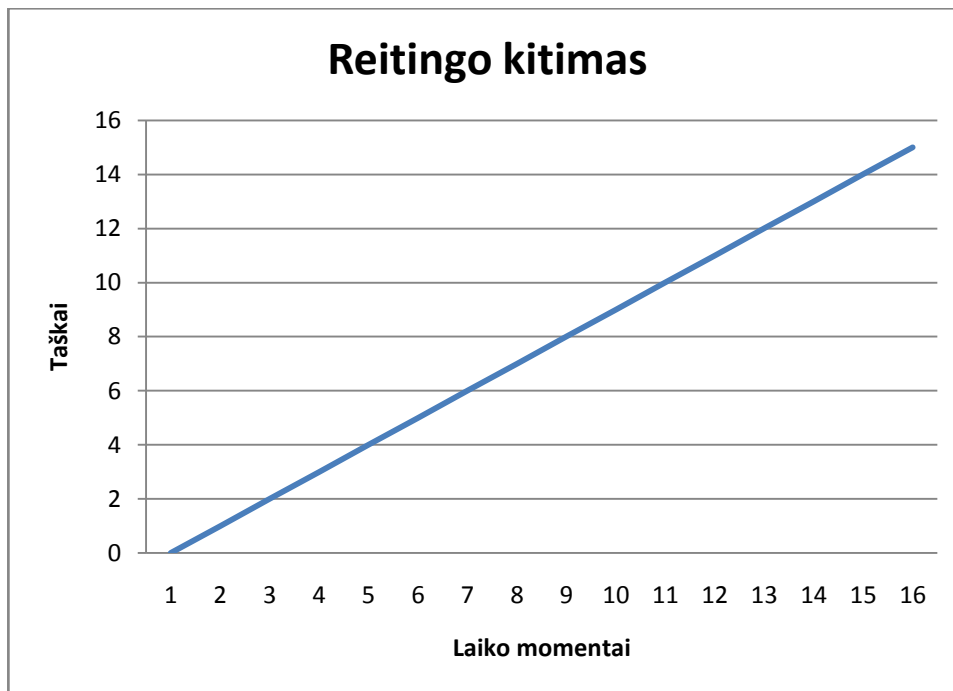
j Reitinguojamo vieneto identifikatorius

R_j Reitinguojamo j -tojo vieneto esamas skaitinis reitingas

v_{ij} Gauta reitingo skaitinė vertė j -tajam reitinguojamam vienetui už i -tajį įvykį

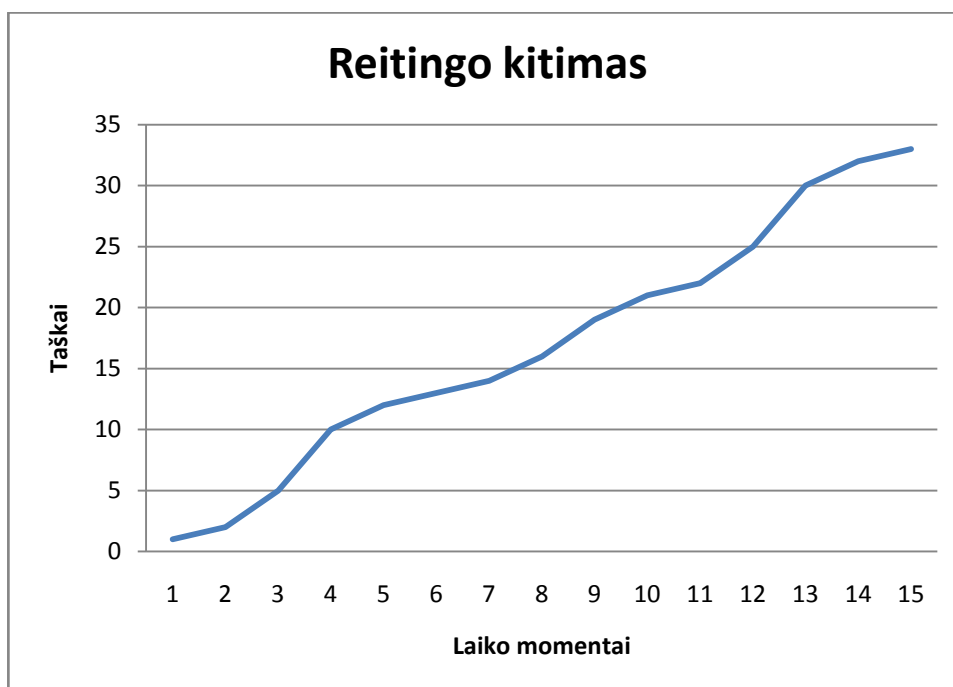
Gauta reitingo reikšmė visiems yra aiški ir visi supranta esminį tikslą – daryti kuo daugiau žinomų veiksmų (atlikus esminį judesį matomas taškų pasikeitimas) tam, kad užkaupiti kuo didesnę taškų kraitį ir išsiveržti į priekį kitų reitinguojamų vienetų akivaizdoje.

Jei tokios reitingavimo sistemos skaitliuko didinimo reikšmė $v = 1$ (taip būna aktyvumo matavimo reitinguose), tuo atveju visa laiko funkcija tampa tiesine **3 pav.** Iš grafiko matoma, jog už kiekvieną atliktą įvykį reitingo balas pakyla vienetu.



3 pav. Tiesinis reitingo kitimo grafikas

Lankstesnei tiesinei sistemai už įvykius skiriami papildomi natūralieji svoriai $v \in \mathbb{N}$ (pvz.: „už nuotraukos patikima galima skirti nuo 1 iki 5 taškų“), tada reitinguojamo vieneto reitingo augimo grafikas yra intervalais tiesiškas (žiūrėkite **4 pav.**).



4 pav. Tiesinis reitingo kitimo grafikas įvertinus koeficientus

Tiek kiek ši sistema yra paprasta, tiek pat ji yra ir nelanksti. Čia galime įžiūrėti aibę neigiamų savybių. Viena iš tokių yra internetinės svetainės senbuviai – pastarieji jau turi prisirinkę pakankamai didelius kiekius taškų, ir naujai pasirodžiusiam svetainės dalyviui

dažniausiai patekti į viršūnes būna ganėtinai sunku arba net visai neįmanoma. Iš to galima daryti išvadas, jog šis reitingavimo būdas labiau tinka vartotojų lojalumui ar senbuviškumui nusakyti, tačiau netinka reitinguojamo vieneto (vartotojo, straipsnio, ...) kokybei nusakyti.

Atitinkamu reitingavimo metodo principu grindžiamas vartomo puslapio populiarumas [7]. Objekto reitingavimui, kuris pagrįstas žiūrimumu (reitinguojame straipsnį, ir jam skaičiuojame reitingą pagal tai kiek kartų pastarasis buvo perskaitytas), tinkamas metodas, kai skaičiuojame internetinio puslapio, kuriame tas objektas publikuotas, apšaukimo skaičių. Lankymo populiarumui išskaičiuoti naudojami tokie kintamieji:

- Puslapio gylis d (kaip giliai reikia nueiti iki puslapio nuo šakninio katalogo);
- Puslapių skaičius tame pačiame gylyje n_d ;
- Nuorodų skaičius į tiriamą puslapį r .

Populiarumo koeficientui a apskaičiuoti naudojama funkcinė priklausomybė:

$$a_i = F\left(d_i, n_i, \frac{1}{r_i}\right), i = 1, \dots, K \quad (2)$$

Priklausomybėje K nurodo internetinių puslapių skaičių.

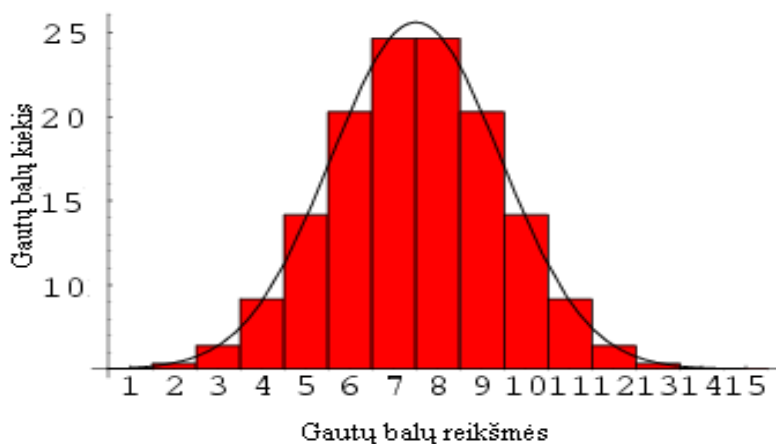
a_i apskaičiuojamas naudojantis formule:

$$a_i = c_1^* d_i + \frac{c_2^* n_i}{r_i} \quad (3)$$

Formulėje c_1^* , c_2^* - konstantos pasirenkamos reitingo svoriui koreguoti.

2.3.2 Vidurkio principu grindžiamas reitingavimas

Naudojant taškų vidurkiu pagrįstus reitingavimo modelius, randame kitus privalumus bei trūkumus. Šiuo metodu galime pasiekti normaliojo skirstinio arba kitaip vadinamo Gauso skirstinio dėsnį [3]. Dėsni pasiskirstymas parodytas **5 pav.**



5 pav. Pavyzdinė Gauso pasiskirstymo kreivė

Šio metodo būdu reitinguojami vienetai dažniausiai vertinami kitų to paties internetinio portalo dalyvių. Reitinguojant atitinkamus vienetus, reitinguotojui būna suteikiama galimybė išsirinkti vieną iš kelių reitingo įverčių. Pradžioje reitinguojamo vieneto galutinė reikšmė labai svyruoja nuo gauto įverčio, nes vidurkis [13] išskaičiuojamas tik iš jau surinktų reikšmių. Vėliau vidurkis nusistoja ties daugmaž bendra visų balsuojančiųjų nuomone.

Vidurkio principo reitingavimo privalumas yra tas, jog jau po keleto pirmųjų reitingavimo įvykių, galima nustatyti reitinguojamo vieneto tikrąją vertinę reikšmę, ir reitinguojamo vieneto ilgas buvimas sistemoje nedaro jo monopolistu dėl surinktų taškų skaičiaus. Priešingai – kuo ilgiau reitinguojamas vienetas būna sistemoj, tuo jo vidutinė reikšmė darosi stabilesnė, ir kiekvienas naujas didelis arba mažas balas galutiniam rezultatui ypatingos įtakos nedaro. Kitas, naudingas finansine prasme, šio reitingavimo metodo aspektas yra tas, jog internetinė svetainė gali nesudėtingais būdais pelnytis iš aukštų balų pardavinėjimo. Gali būti apibrėžti didelių įverčių taškai, kurie yra atitinkamai apmokestinami. Žinoma šią savybę galima vertinti ir neigiamai, nes tokiu atveju balsavime dalyvauja ne reitinguojamo vieneto kokybė, o pinigų laisvumo forma.

Šiame metode yra dar ir kita neigiama naudojimo pusė. Atsiranda galimybės nesažiningam reitingavimui. Jei kiti vartotojai norės, kad reitinguojamas vienetas neiškiltų, jie gali tyčia skirti mažus balus. To pasekoje reitinguojamo vieneto galutinis rezultatas mažės nuo savo vidurkio normos. Taip pat ši reitingavimo sistema yra labai lengvai perprantama ir nesuteikia jokio intelektualumo pojūčio reitinguojamiems vienetais bei patiems reitinguotojams.

2.3.3 Konsensuso(pastovumo) prioritetų reitingavimo schemas

Naudojantis konsensuso prioritetų reitingavimo sistema galima patogiai apskaičiuoti objekto arba objektų grupės vertę iš anksto apibrėžtomis kokybės linijoms. Sistema pagrįsta egzistuojančia aibe objektų bei reitinguojamų aspektų, pagal kuriuos vertinami vartotojai. Reitingo uždirbtų taškų kiekis tiesiogiai priklauso nuo atlikos užduoties ir gauto rezultato skirtumo nuo tikrojo užduoties rezultato [14]. Kiekvienos užduoties teisingas rezultatas yra tam tikras sveikasis skaičius, o besivaržančio objekto gautas rezultatas gali pakliūti į trejas sritis:

- Gautas rezultatas yra didesnis nei „tikrasis“;
- Gautas rezultatas yra lygus „tikrajam“;
- Gautas rezultatas yra mažesnis nei „tikrasis“.

Žinoma, geriausias pasiekimas reitinguojamam objektui yra tas, kai jis gauna rezultatą lygų „tikrajam“ užduoties rezultatui. Cook ir Seiford metodas [15] intuityviai demonstruoja galimybę sujungti nukrypimų nuo esminio taško skirtumus tam, kad apibrėžti pastovumo reitingus. Metodas pagrįstas reitinguojamų objektų gautų taškų skirtumo iki realiosios užduoties reikšmės mažinimu.

Probleminėje srityje turime n reitinguojamų objektų – vartotojų ir m užduočių. Tuo atveju r_{ij} bus atskiro vartotojo atskiros užduoties reitingas, kai:

$$i = 1, \dots, n; \quad (4)$$

$$j = 1, \dots, m; \quad (5)$$

r_j^c yra tikrasis j -tosios užduoties rezultato įvertis. Tada i -tojo vartotojo absoliutus pastovumo reitingas apibrėžiamas formule:

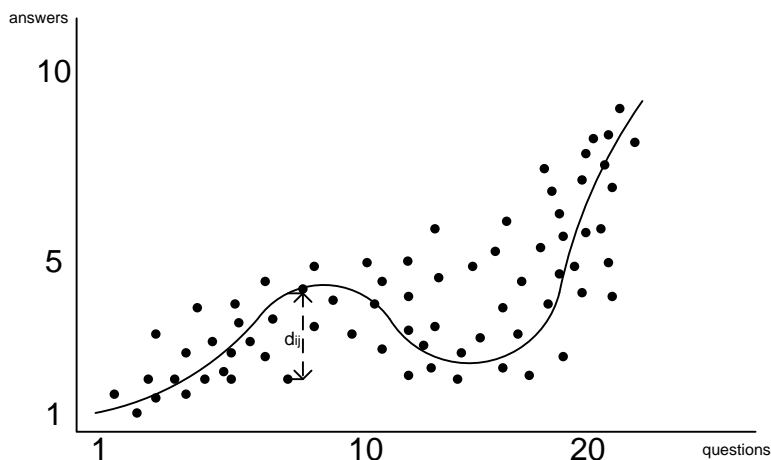
$$d_i = \sum_{j=1}^m |r_{ij} - r_j^c|, i = 1, \dots, n. \quad (6)$$

Tuo atveju, visų vartotojų arba vartotojų grupės, komandos absoliutus pastovumo reitingas apibrėžiamas formule:

$$d = \sum_{i=1}^n d_i = \sum_{i=1}^n \sum_{j=1}^m |r_{ij} - r_j^c| \quad (7)$$

Kiekvienoje užduotyje triumfuoja tie vartotojai, kurių absoliutus nuokrypis nuo realios užduoties reikšmės yra pats mažiausias arba lygus nuliui. Skaičiuojant bendrą visų užduočių

įvertinimų skirtumą d_i kiekvienam vartotojui atskirai, taip pat galime nusakyti bendrą viso uždavinyno išsprendimo kokybiškumą. **6 pav.** galime matyti galimą atsakymų išsimėtymą apie optimumo kreivę.



6 pav. Atsakymų išsidėstymas aplink teisingumo kreivę

Šis metodas tinka gauti reitinguojamų objektų – vartotojų reitingus, leidžiančius nusakyti kuris iš besirungusių dalyvių buvo arčiausiai teisingo rezultato. Metodas labiau tinkamas vartotojų skatinimui tobulinti savo žinias ir vis labiau gerinti atsakymų kokybę. Taip pat atsirandanti privalomoji problema yra ta, jog kiekvienam uždaviniui reikia pateikti teisingo įverčio reikšmę. Taip gaunasi, kad vartotojų aktyvumą naudojantis šiuo reitingavimo algoritmu skatina ir kuria internetinio portalo ekspertas, ruošdamas ir publikuodamas galimus reitingų elementus su jų atitinkamais įverčiais. Vartotojo veiksmai tokioje aplinkoje neatlieka kūrybinio-statybinio darbo ir naujos reikalingos žinių portalo turiniui informacijos neįneša.

2.3.4 ELO reitingavimo sistema

Naudojantis ELO reitingavimo modeliu [1][12], reitinguojamo vieneto kokybė gali būti išskaičiuojama tik iš pergalių arba pralaimėjimų tarp dviejų varžovų. Tokiu atveju skirtingų reitinguojamų vienetų reitingai tampa giminingais. Varžydamiesi tarpusavyje reitinguojami vienetai, iš karto gali pasiskaičiuoti kaip pasikeis jo ir jo priešininko galutiniai taškų rezultatai įvykus rungtynėms su atitinkamomis pergalių pabaigomis. Šis metodas yra plačiai taikomas šachmatų žaidimo žaidėjų įvertinimui bei reitingavimui ir dėl teikiamos informacijos tikslumo metodas yra plačiai taikomas pasaulinėje šachmatų žaidimo lygoje. Metodą bandoma pritaikyti visur, kur reitingavimo specifika atitinka šio modelio koncepciją.

Matematinės modelio detalės.

ELO reitinguojamo tipo varžytuvių pabaiga gali baigtis trejopai:

- Lygiosiomis;

- Pergalę;
- Pralaimėjimu.

Vertinant taškais:

- Už pralaimėjimą aktualus išloštas taškų skaičius yra lygus 0;
- Už sužaidimą lygiosiomis išloštas taškų skaičius yra lygus 0,5;
- Už pergalę išloštas taškų skaičius lygus 1.

Žaidėjo A esamas reitingas žymimas R_A , o naujasis žaidėjo reitingas apskaičiuojamas pagal formulę:

$$R'_A = R_A + K(S_A - E_A) \quad (8)$$

Čia:

R_A esamas žaidėjo A reitingas

R'_A naujas žaidėjo A reitingas

K svorio reguliavimas (stipriems žaidėjams $K=16$, silpnesniems žaidėjams $K=32$)

S_A galutinis surinktas aktualių taškų kiekis, žaidėjui A žaidžiant su vienu priešininku

E_A lauktas (ang. Expected) taškų skaičius išloštas žaidėjo A žaidžiant su atitinkamu priešininku

Lauktas taškų skaičius E_A apskaičiuojamas pagal formulę:

$$E_A = \frac{1}{1 + 10^{(R_B - R_A)/400}} \quad (9)$$

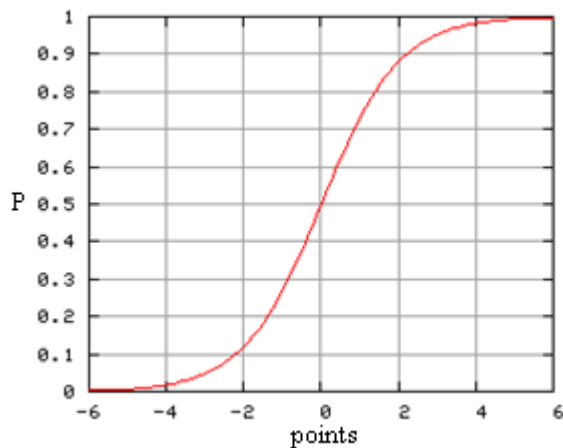
Analogiškai taškų skaičių priešininkui galima suskaičiuoti pagal formulę:

$$E_A = \frac{1}{1 + 10^{(R_A - R_B)/400}} \quad (10)$$

Pastebėtina, kad

$$E_A + E_B = 1 \quad (11)$$

Laukto taškų skaičiaus E_A grafinis išsidėstymas remiasi logistine kreive [2], kuri pavaizduota **7 pav.**



7 pav. Logistinė kreivė

Šis taškų pasiskirstymo dėsnis realizuoja gyvenimišką ribų metodą, kai priartėti prie maksimumo/minimumo reikšmės yra sunkiau nei būti vidurkyje. Ši savybė labai tinka vartotojų reitingavimo modeliui, kadangi galime apsibrėžti norimas ribas.

Imant pavyzdį, jei žaidėjas turintis pradinį reitingą 1613 žaidžia su priešininkais, ir atlieka tokias pergales/pralaimėjimus: pralošia prieš žaidėją, kurio reitingas 1609, sužaidžia lygiosiomis su žaidėju, kurio reitingas 1477, laimi prieš žaidėją, kurio reitingas 1388, laimi prieš žaidėją, kurio reitingas 1586 ir pralošia prieš žaidėją kurio reitingas 1720, tai aptariamo žaidėjo aktualus taškų skaičius yra $(0 + 0.5 + 1 + 1 + 0) = 2.5$, o laukiamas taškų skaičius pagal aukščiau minėtą formulę gaunamas $(0.506 + 0.686 + 0.785 + 0.539 + 0.351) = 2.867$. Dėl šių naujų įvykių aptariamo žaidėjo naujasis reitingas bus $(1613 + 32(2.5 - 2.867)) = 1601$.

ELO metodo reitingo privalumas labai aiškus – jis skirtas priešininkų poros žaidimo vertinimui, kurio pabaigą galiausiai nulemia pasirinkto priešininko stiprumas ir sužaisto žaidimo pabaigos rezultatas. Žaidėjams nepalikta galimybių sukčiauti. Žaidimas gali nulemti žaidėjo galutinį rezultatą ne tik teigiama bet ir neigiama linkme.

Pagrindinius metodo minusus galima būtų apibrėžti tik juos taikant kitoje reitingavimo srityje. Taškų kaupimas yra tik vienai siaurai sričiai, nėra galimybės detalizavimui (taškas gaunamas tik už žaidimą). Nėra galimybės išplėsti panaudojamumo, atsiradus kitiems reitingavimo elementams (reitingavimas už aktyvų dalyvavimą, reitingavimas už papildomo įvykio sukūrimą, ...). Šis reitingavimo metodas neįvertina parametro, kurio dėka galima mažinti dalyvio reitingą dėl įsisenėjimo. Taip pat šis metodas ir savo taikomajai sričiai turi neigiamų aspektų: formulėje (8) esanti konstanta K turi būti parenkama atitinkamai pagal žaidėjo stiprumą, kurį pradžioje turi nuspręsti sistemos konfigūravimo ekspertas. Jei parinkta reikšmė bus per maža, formulės reikšmių jautrumas bus labai mažas ir bus sudėtinga pasiekti

reikšminį kiekį taškų pergalei siekti. Kitu atveju, jei K reikšmė bus per didelė, atsiras labai didelis laimėjimų bei pralaimėjimų jautrumas, ko pasekoje žaidėjai nelabai orientuosis savo profesionalumo lygyje.

2.3.5 Glicko reitingavimo sistema

Glicko reitingavimo metodo praktinis pritaikomumas yra toks pats kaip ir ELO reitingavimo sistemos. Priešingai nei ELO reitingavimo sistema, Glicko įvertina vartotojo patikimumo faktorių. ELO sistemoje, jei rungtis du žaidėjai su tokiu pačiu reitingo dydžiu, tai pergalės atveju, vienam žaidėjui taškų skaičius sumažėja per n vienetų, o kitam žaidėjui atitinkamai padidėja per n vienetų. Glicko sistemos pranašumas yra tas, kad jei varžosi du varžovai su vienodu reitingu, tačiau vienas iš jų nedalyvavęs žaidimuose kelerius metus, o kitas yra aktyvus žaidėjas, tai įvykus seniai nežaidusio žaidėjo pergalei jo taškų skaičius padidėja kn taškų skaičiumi, o dažnai žaidžiančiojo sumažėja pn taškų skaičiumi, kur $k > p$. Taip gauname, jog dažnai žaidžiančio žaidėjo stabilumas yra didesnis ir labiau užtikrintas. Glicko sistema įveda dar vieną faktorių į reitingų skaičiavimo formulę – reitingo nuokrypį (ang. Ratings deviation (RD)), arba kitaip statistiškai vadinamą standartinį nuokrypį [6] (ang. Standard deviation). Naudojantis šiuo metodu didelis RD nusako nepatikimus reitingus, taip pat, kad dalyvis su šiuo RD yra retas žaidėjas. Atitinkamai priešingai, jei RD mažas, tai nurodo, jog žaidėjo reitingas yra labai patikimas ir jis yra dažnas žaidėjas.

Glicko sistemoje vartotojo reitingas keičiasi nuo sužaisto žaidimo kokybės: pergalė, pralaimėjimas ar lygiosios. Tačiau RD koeficientas keičiasi priklausomai nuo to, kada ir kiek dažnai žaidėjas žaidžia [8]. Kiekvienas sužaistas žaidimas sumažina vartotojo RD.

Kadangi žaidėjas Glicko sistemoje turi ir reitingą ir RD, tokiu atveju jis yra daug informatyvesnis nei ELO sistemoje. Mes iš to žinome žaidėjo stiprumą ir pasiruošimo formą tam tikrame laiko intervale. Naudojantis Glicko sistema, vartotojo galutinis reitingas nusakomas intervalu, kurio tikrumas apibrėžtas 95%. Intervalas nusakomas rėžiais:

$$[r - 2 * RD; r + 2 * RD] \quad (12)$$

Žaidėjas sistemoje gali būti dviejų tipų: naujas (neturintis jokio reitingo) ir senas (turintis sukauptą reitingą). Naujam žaidėjui suteikiamas pradinis reitingas pagal nutylėjimą 1500 taškų. Reitingo nuokrypis abiem žaidėjų tipams išskaičiuojamas pagal formulę:

$$RD = \min\left(\sqrt{RD_{old}^2 + c^2 t}, 350\right) \quad (13)$$

Čia:

RD_{old}	Senasis reitingo nuokrypis
RD	Naujasis reitingo nuokrypis
t	Nuo paskutinių varžybų praėjęs laikotarpio periodų skaičius (jei dalyvis nepraleidžia nei vieno periodo, tai $t = 1$)
c	Konstanta, kuria padidinamas vartotojo nepatikimumas kiekvienu periodu.

Iš formulės matome, jog jei dalyvis yra naujas ir iki šiol nėra nustatytas jo RD, tuo atveju $RD = 350$. Taip pat sistema neleidžia RD būti didesniai nei 350.

Reitingų ir nepatikimumo atnaujinimai vykdomi naudojantis šiomis formulėmis:

tarkime kad r' ir RD' yra reitingo ir reitingo nuokrypis po skaičiavimų

$$r' = r + \frac{q}{1/RD^2 + 1/d^2} \sum_{j=1}^m g(RD_j) (s_j - E(s|r, r_j, RD_j)) \quad (14)$$

$$RD' = \sqrt{\left(\frac{1}{RD^2} + \frac{1}{d^2}\right)^{-1}} \quad (15)$$

Čia:

$$q = \frac{\ln 10}{400} = 0.0057565; g(RD) = \frac{1}{\sqrt{1+3q^2(RD^2)/\pi^2}}; E(s|r, r_j, RD_j) = \frac{1}{1+10^{-g(RD_j)(r-r_j)/400}};$$

$$d^2 = \left(q^2 \sum_{j=1}^m (g(RD_j))^2 E(s|r, r_j, RD_j) (1 - E(s|r, r_j, RD_j)) \right)^{-1}$$

Glicko reitingavimo sistemos vartotojų aktyvumo įvertinimas yra teigiamai vertinamas parametras bet kokioje kuriamoje reitingavimo sistemoje ir gali būti sėkmingai pritaikytas vartojimui internetinių portalų bet kokių objektų reitingavime. Žinoma šiame Glicko reitingavimo modelyje išlieka reitinguojamų objektų aibės mažas išplečiamumas – čia reitinguojami tik tarpusavyje besivaržantys objektai, veiksmas vyksta poromis, o šis metodas nelabai tinkamas internetinio žinių portalo reitingavimo metodui. Be to iš pateiktų skaičiavimo formulių matoma, jog rezultatų apskaičiavimas gerokai labiau apsunkins sisteminius resursus.

2.3.6 Glicko-2 reitingavimo sistema

Glicko-2 reitingavimo modelis turi ne tik objekto reitingą r , reitingo nuokrypį RD , bet dar vieną papildomą parametą – reitingo nepastovumą σ [9]. Šis parametras apibrėžia reitingo reikšmės šuolių kitimą. Reitingo nepastovumas σ būna didelis, kai reitinguojamo

objekto reitingas turi permainingus kitimus (jei objekto reitingas ilgą laiką buvo pastovus ir vėliau netikėtai šoktelėjo) ir atvirkščiai, kai reitingo nepastovumas σ būna mažas, tai nusako, jog reitinguojamo objekto reitingai tolydžiai didėja, mažėja arba yra pastovaus dydžio.

Reitingavimo pradžioje reitinguojamas objektas turi turėti pradinį arba jau sukauptą reitingą, reitingo nuokrypį bei reitingo nepastovumą. Glicko-2 reitingavimo skalė skiriasi nuo Glicko reitingavimo skalės, dėl to pradžioje duomenis reikia perversti iš vienos skaičiavimo sistemos į kitą, o pabaigoje atverčiama atgal.

Skaičiuojant reitingus pradžioje apsibrėžiam reitingo nepastovumo kitimo parametras τ ribose nuo 0.3 iki 1.2. Atitinkamai kaip ir Glicko sistemoje, jei reitinguojamas objektas neturi pradinių reitingų, nustatome juos atitinkamai $r = 1500$, $RD = 350$, $\sigma = 0.06$.

Glicko-2 reitingų skaičiavimo būdas įgalina sekti reitinguojamų objektų intelektualinius šuolius, kurie galbūt yra pagrįsti apgaule ar melu. Turinčius didelį reitingo nepastovumą klientus galima labiau stebėti ir vertinti jų atliekamus veiksmus.

Glicko-2 reitingo skaičiavimo papildomi parametrai labiausiai reikalingi sistemą prižiūrinčiam ekspertui ar administratoriui. Turint šias reikšmes galima bandyti ieškoti galimų taškų nesutapimų ar pervertinimų. Pagrindiniai sistemos panaudojamumo internetiniam portalui minusai išlieka tokie patys kaip ir pradinėje Glicko reitingavimo sistemoje. Privalumas – vartotojų reitingų fliktacijų stebėjimas.

2.4 Analizės išvados

Analizės metu prieita prie išvadų, jog:

- Paprasta tiesinė reitingų skaičiavimo sistema nepakankamai išpildo vartotojų lūkesčius, nes ji augina skaitliuką, tačiau neatsižvelgia į laiko ir kokybiškumo klausimus;
- ELO reitingavimo sistemos taškų skaičiavimo principai yra ganėtinai teisingi ir priimtini norint nustatyti reitinguojamo vieneto kokybiškumą, tačiau ši sistema plačiau naudojama rungtynėse, kur varžovai varžosi poromis vienas prieš kitą;
- Reitingavimo sistemos privalumas didėja, atsirandant laiko, aktyvumo bei rezultatų nepastovumo įvertinimo parametrams;
- Norint padidinti reitingavimo sistemos lankstumą, keičiantis poreikiams, kokius vienetus reitinguosime, reikia sukurti tokią duomenų bazės struktūrą, kuri kiekvieną reitinguojamą vienetą interpretuotų kaip elementarų objektą;

- Kuo platesnė vartotojų reitingavimo galimybių aibė, tuo labiau atsiranda didesnis stimulus varžytis ir lenktyniauti;
- Objektyviai reitingavimo sistemai palaikyti reikia realizuoti tokį metodą, kurio darbui atlikti būtų kuo mažiau naudojamas žmogiškasis eksperto įsikišimas.

3 Reitingavimo metodo formali koncepcija

Antrame dokumento skyriuje apžvelgtoje analizėje matome dėsnius, jog reitinguojančios internetinės svetainės naudoja įvairius objektų vertinimo algoritmus. Dalis svetainių stengiasi pateikti stipresnius reitingavimo algoritmus, kitos – vertinimui nesuteikia didelio dėmesio. Kiekviena svetainė ir kiekvienas atskiras reitingavimo algoritmas teikia naudą, tačiau turi ir trūkumų.

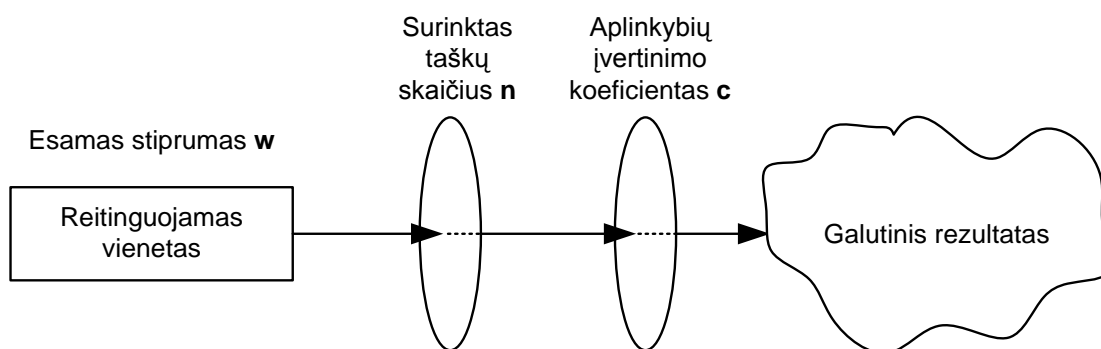
Šio skyriaus tikslas atrinkti geriausias ir tinkamiausias reitingavimo savybes internetinei žinių svetainei. Pateiksime reitingavimo modelio koncepciją: duomenų rinkimo, reitingų skaičiavimo ir pateikimo algoritmus.

3.1 Reitingų skaičiavimo prototipas

Nuo reitingų skaičiavimo principo priklauso visa tolimesnė reitingavimo sistemos fizinė ir programinė struktūra. Kursime lanksčią, lengvai tvarkomą, koreguojamą ir prižiūrimą sistemą. Reitingų skaičiavimai bus atliekami remiantis išnagrinėtais reitingavimo metodais.

3.1.1 Bendras reitingų skaičiavimo algoritmas

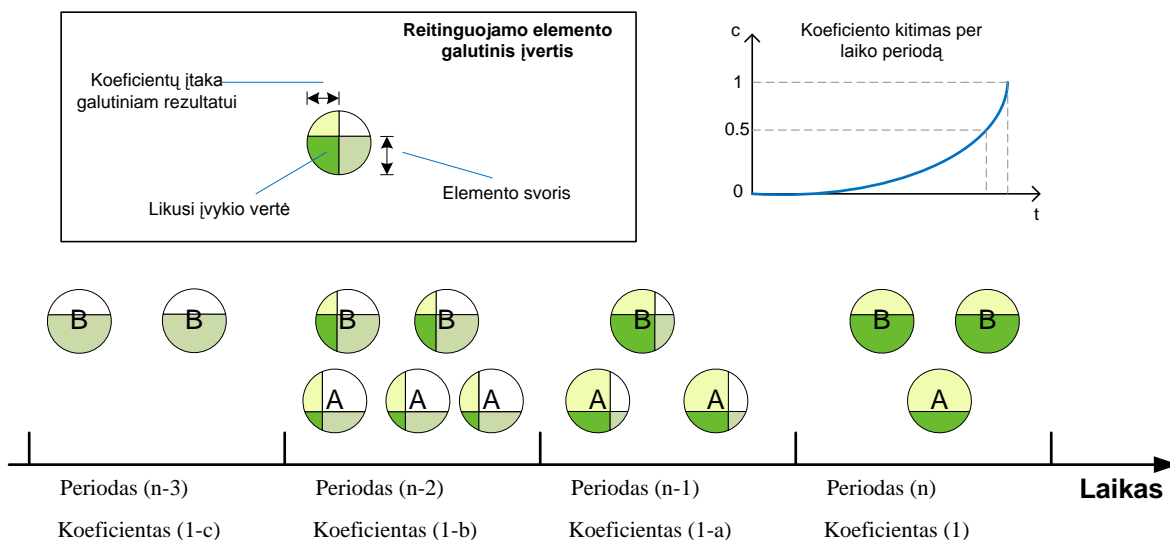
Tam, kad pasiekti didesnes galimybes reitingų skaičiavimo variacijose, daugumoje reitingavimo įvykių naudosime svorių mažinimo koeficientus bei pačius svorius. Koeficientų dydžiai priklauso nuo įvykio naujumo ar kokybiškumo. Rinkdamas savo galutinį vieno įvykio rezultatą, reitinguojamas vienetas pereina tokius filtrus, kaip pateikta **8 pav.**



8 pav. Reitinguojamo vieneto rezultato skaičiavimo kelias

Laiko klausimui reitinguojamo objekto rezultatui nuspręsti naudosime laiko periodo sąvoką. Skaičiuojant reitinguojamo vieneto visą suminį per istoriją surinktą rezultatą, turi būti sumuojami momentiniai surinkti taškai ir balai įvertinus periodų pasikeitimus. Visi laiko periodai turi savo svertinį koeficientą, kuris einant laiko juosta atgal mažėja $[0; 1]$ ribose. Einamojo laiko periodo svorio koeficientas lygus 1, o prieš tai buvusių svoriai mažėja. Periodų dydžiai bei svorių koeficientai gali būti kintami pagal poreikius. Jei gyvavimo ciklo

eigoje keičiasi periodo dydžiai ar svorio koeficientai, atitinkamai keičiasi ir reitinguojamų vienetų galutiniai įverčiai **9 pav.**



9 pav. Reitinguojamų vienetų svorio kitimas per laiko periodą

9 pav. galime matyti, kaip išsidėsto išsivaizduojamo reitinguojamo vieneto reitingo elementų įverčiai einant periodui. Vieno periodo reitinguojamų vienetų įvertis apskaičiuojamas pagal formulę:

$$r_i = \sum_{j=1}^m l_j e_{ij} \quad (16)$$

Čia:

- r_i i-tojo reitinguojamo vieneto surinktų reitingų svoris per periodą
- m Reitingo elementų skaičius
- l Reitingo elemento svoris
- e Reitinguojamo vieneto surinktas taškų skaičius už reitinguojamą elementą

Iš formulės matoma, jog reitinguojamo vieneto reikšmė per periodą gali svyruoti $[0; \infty)$ ribose. Ši atvira reitingo riba netenkina rezultatų apibrėžtumo reikalavimo. Šiam neapibrėžtumui naikinti, l_j reikšmei skaičiuoti įvesime dar vieną santykinę daliklį. Kiekvienam reitinguojamam elementui apibrėšime maksimalias galimas pasiekiamas jo skaitines ribas. Tarkim nustatysim, jog vienas autorius gali parašyti maksimumą $re^{max} = 10$ straipsnių. Tokiu atveju:

$$l_j = \frac{l_j^*}{re^{max}} \quad (17)$$

Šitaip, gauti vienetinį įvertinimą, autorius turėtų parašyti 10 straipsnių. Reitingavimo tikslas yra neleisti reitinguojamam vienetui pasiekti vienetinę reitingo ribą. Tuo tikslu, jei reitinguojamas vienetas pasiekia maksimalią ribą kokiame nors reitinguojamame elemente, tada nustatyta maksimali riba yra automatiškai padidinama pradiniu nustatytu maksimumu:

$$re_n^{max} = re_{n-1}^{max} + re_1^{max} \quad (18)$$

Čia:

n Maksimumo keitimo eilės numeris

Galutinis reitinguojamo vieneto A reitingas apskaičiuojamas įvertinus visų laiko periodų svorių koeficientus:

$$r_n = \frac{\sum_{j=1}^m l_j e_j + \sum_{i=1}^{n-1} k_i r_i}{n} \quad (19)$$

Čia:

n Periodų skaičius

m Reitingo elementų skaičius

l Reitingo elemento svoris

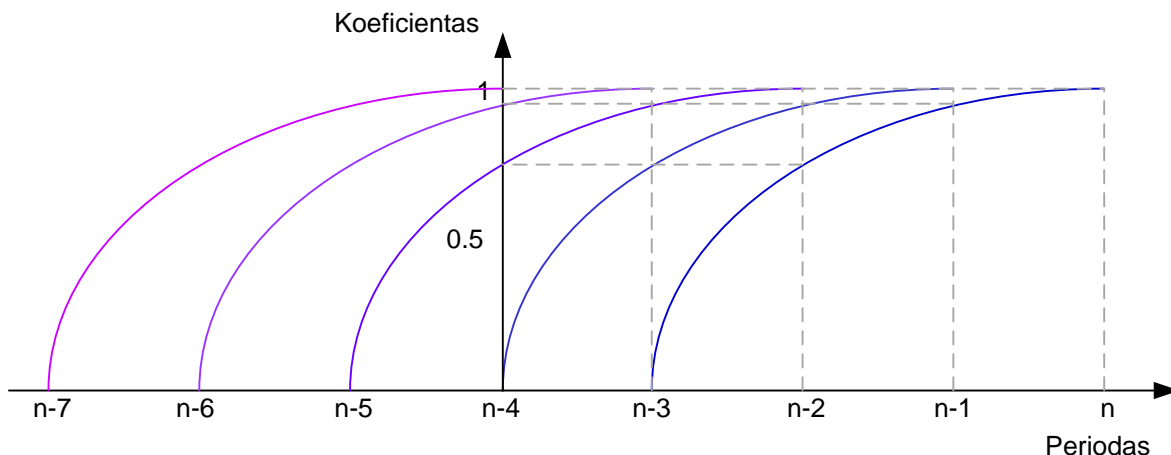
e Reitinguojamo vieneto surinktas taškų skaičius už reitinguojamą elementą

k Periodo svoris

r Periodo metu surinktas reitingas

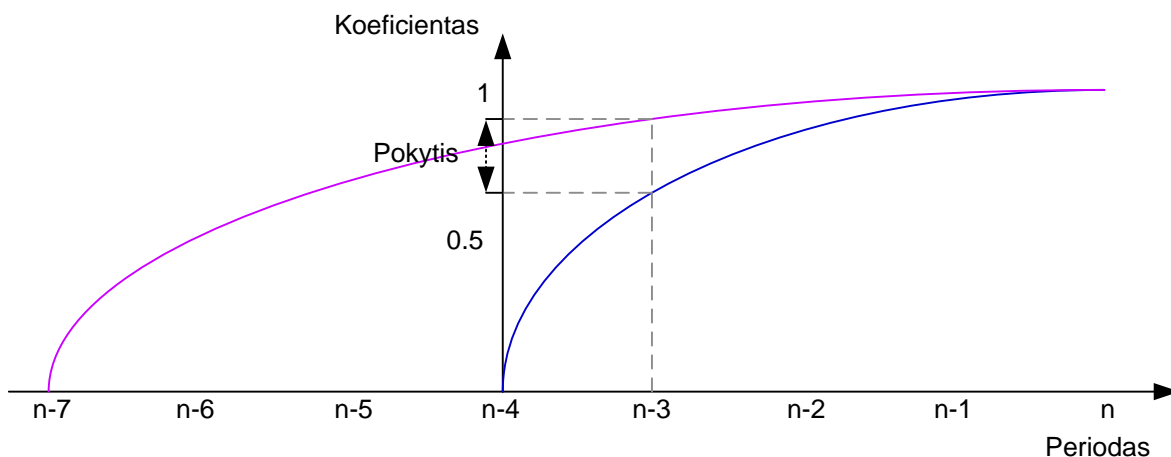
Šio algoritmo panaudojimas neleidžia reitinguojamam objektui įsigalėti visoje sistemoje dėl jo senbuviškumo. Visų ankstesniuose perioduose taškų galia santykinai mažinama atitinkamai to periodo koeficiento dydžiui. Periodo pasikeitimo momentu visų ankstesnių periodų koeficientiniai svoriai yra mažinami. Tokiu atveju atsiranda galimybė reitinguotam vienetui iškilti aukščiau kitų tik dėl to, jog kiti vienetai per ankstesnį laiko periodą atliko ar gavo mažiau reitinguojamų veiksmų (nesvarbu, net jei vienetai buvo ankstesniais periodais surinkę daugiau taškų). Šiam požymiui išlaikyti ir norimai sustiprinti, reikia periodų koeficientų parinkimui naudoti eksponentinio kitimo dėsnį. Dėsnio testinumui išlaikyti reikia

apsibrėžti periodų skaičių, kuriems praėjus periodo koeficientas tampa lygus nuliui. Periodų svorių keitimo dėsningumas pateiktas **10 pav.**



10 pav. Periodų svorių keitimo dėsnis

Iš **10 pav.** matome, jog kiekvienas tolimesnis periodas turi vis mažesnę koeficientą ir jei apibrėžtas išnulinimo periodas, pavyzdžiui, lygus ketveriems periodų vienetams, tai jiems praėjus visų reitinguojamų vienetų surinkti balai į galutinį reitingą neįsiskaičiuoja, arba kitaip sakant jie tampa lygūs nuliui. Periodo koeficiento kitimo ribos priklauso nuo to, kokį periodų skaičių laikysime neišnulinamu. Periodų koeficientų kitimo dėsnis priklausomai nuo neišnulinamo periodo ilgio pateikiamas **11 pav.**

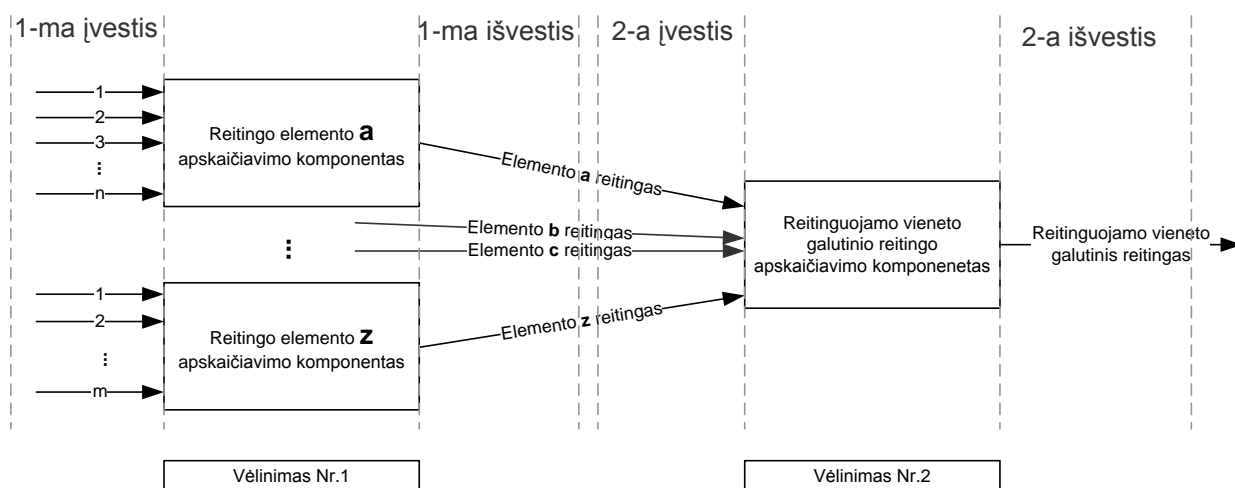


11 pav. Periodo koeficiento svorio priklausomybė nuo neišnulinamo periodo ilgio

Šis periodų pagal svorių koeficientus naudojimo algoritmas įgyvendina Glicko metodo vartotojo patikimumo faktorių – naujesniais praėjusiais periodais nedalyvavę reitinguojami vienetai praranda didžiąją dalį savo surinktų taškų.

Sistemos lengvam išplečiamumui pasiekti reikalinga objektus apibendrinanti reitingavimo struktūra. Reitinguojamo vieneto reitingo apskaičiavimui naudojamas dalinai

universalus algoritmas, kuris grafiškai pateiktas **12 pav.** Reitinguojamo vieneto galutinis reitingas apskaičiavimo komponentas yra nekintantis jei keičiasi, atsiranda ar panaikinami patys reitinguojami vienetai bei reitingų elementai. Komponentų išvestys visada yra nekintančios ir išvedamų rezultatų kiekis visada lygus 1. Įvesčių skaičius gali kisti pagal tai, kokius reitinguojamus vienetus bei elementus apskaičiuojame. Reitinguojamo vieneto galutinio reitingo apskaičiavimo komponento įvesčių skaičius kiekvienam reitinguojamam vienetai gali skirtis, tačiau pats apskaičiavimo algoritmas nesikeičia, tiesiog keičiasi kintamojo m reikšmė formulėje $\sum_{j=1}^m l_j e_{ij}$.



12 pav. Reitinguojamo vieneto reitingo apskaičiavimas

Kiekvieno komponento darbas techniškai atlieka vėlinimą. Galutinio reitingo skaičiavimo komponento vėlinimas beveik nesikeičia atsiradus naujiems reitingavimo elementams, nes sumavimo algoritmas yra paprastas ir nereikalaujantis didelio mašininio darbo, tačiau kiekvieno reitingo elemento apskaičiavimo komponento vėlinimas gali ženkliai skirtis. Pavyzdžiui apskaičiuojant straipsnio pradinį reitingą reikia įvertinti daugiau aspektų, tokių kaip autorių skaičius, autorių lygiai, autorių svoriai, tematika ir t.t. Specifinių reitingų elementų apskaičiavimo komponentų algoritmai turi būti kuriami atsakingai įvertinant sistemos laikinio darbo kainą.

3.1.2 Reitinguojamų objektų aibės išskyrimas

Internetinėje žinių svetainėje du pagrindė vyraujantys reitingavimo vienetai yra „žinios“ (straipsniai, kūriniai, ...) bei „žinių autoriai“ (esami arba būsimi straipsnių autoriai, kūriniių autoriai). Skirtingus reitingavimo vienetus reitinguojame skirtingais aspektais, tačiau kiekvienas reitinguojamas elementas turi tuos pačius taškų skaičiavimo principus – svoris už atitinkamą reitingavimo elementą. Kiekvienam reitinguojamo vieneto reitingavimo elemento

skaitiniam kaupikliui gali būti sugalvotas skirtingas taškų apskaičiavimo būdas. Šiam tikslui skaičiuojant reitingavimo elemento rezultatą visada turėsime įvairiais būdais suskaičiuotą „kiekio“ reikšmę, kuri vėliau atitinkamai visiems reitingavimo elementams dalyvaus bendroje formulėje.

3.1.3 Autorių reitingavimo savybės

Kiekvienam internetinėje svetainėje dalyvaujančiam registruotam vartotojui pradžioje reikia sukurti pradinį jo reitingą. Tam, kad galėtume praplėsti vartotojo teisių funkcionalumą, įvedame vartotojo lygio sąvoką. Lygių skaičius gali būti nustatomas pasirinktinai, tačiau dėl bendros tvarkos naudojame 3 lygius. Apsibrėžiame, jog trečias lygis yra visų žemiausias, o pirmas – aukščiausias. Visi vartotojų lygiai turi savo svorinius koeficientus, kurie naudojami skaičiuojant atitinkamus vartotojo veiksmų įvertinimus. Kiekvienam lygiui yra nustatyta taisyklė, kas gali į jį pakliūti. Taisyklės paprasčiausia apibrėžti skaičių intervalais, kurie reikštų vartotojo tinkamumą į vieną ar kitą lygį. Lygių intervalai turėtų būti padalinti norimais intervalais $[0; 1]$ ribose, pvz. $[0; 0.3); [0.3; 0.7); [0.7; 1]$. Tokiu atveju, naujai užsiregistravusio vartotojo pradinis reitingas lygus nuliui ir jis patenka į trečią lygį.

Reitingavimo elementai vartotojui gali būti įvairūs, tokie kaip pradinis vartotojo reitingas, išverstų straipsnių kiekis, autorinio indėlio į straipsnių rašymą kiekis, komentarų parašymo kiekis, recenzijų parašymo kiekis, straipsnių vertinimo kiekis ir kita. Kiekvienam reitingavimo elemento skaičiavimui gali būti taikomas vis kitoks algoritmas. Pradinis vartotojo reitingas yra vienas iš reitingų elementų, todėl jį apskaičiuojant naudojama standartinė procedūra išgaunant reitingo elemento svorį pagal periodą bei maksimumo reikšmę bei ją padauginant iš vartotojo lygio svorinio koeficiento:

$$ir^{User} = \frac{rire^{Number}}{rem^{MaxValue}} * re^{Weight} * ul \quad (20)$$

Čia:

ir^{User}	Pradinis vartotojo reitingas
$rire^{Number}$	Reitinguojamo vieneto reitingo elemento surinktas taškų skaičius
$rem^{MaxValue}$	Reitinguojamo elemento pasirinktas maksimalus taškų skaičius (pradiniam vartotojo reitingui jis bus lygus 1)
re^{Weight}	Reitinguojamo elemento svoris
ul	Vartotojo lygio svoris

Vartotojo lygis gali kilti atitinkamai dėl atliekamų veiksmų. Kiekvieną kartą pasikeitus bendram vartotojo galutiniam reitingui, jo reikšmė yra lyginama su lygių intervalų reikšmėmis ir pagal sulyginimo rezultata vartotojo lygis paliekamas toks koks buvo arba pakeičiamas. Pasikeitus vartotojo lygiui, pradinis vartotojo reitingas neperskaičiuojamas.

Jei straipsnio autorius ne vienas pats sukūrė straipsnį, tai prie šio vartotojo straipsnių parašymo kiekio prisisumuoja indėlio vertė į darbą. Atitinkamai už visus reitinguojamus elementus konkretiems vartotojams yra kaupiama surinktas įvykių kiekis, kuris vėliau dalyvauja bendruose rezultatų skaičiavimo algoritmuose.

3.1.4 Straipsnių reitingavimo savybės

Straipsniai kaip ir vartotojai turi jiems pritaikytus reitingavimo elementus, kurie nesikerta su kitų reitinguojamų vienetų reitingų elementų aibėmis. Tai gali būti pradinis straipsnio reitingas, įvertinimų kiekio reitingas, įvertinimų reitingas, išvertimų į kitas kalbas kiekis, komentavimo kiekis, recenzijų straipsniui kiekis, recenzijų įvertinimų reitingas, straipsnio perskaitymo kiekis ir kiti.

Pradinis straipsnio reitingas skaičiuojamas sudėtingiau nei pradinis vartotojo ar autoriaus reitingas. Straipsnio stiprumas tiesiogiai priklauso nuo jį parašiusių autorių patirties, todėl turime įvertinti visų autorių stiprumo ir indėlio į darbą santykį:

$$w^{Users} = \frac{\sum_{i=0}^n (w_i^{User} * c_i^{User})}{n} \quad (21)$$

Čia:

w^{Users}	Vidutinis autorių rašiusių straipsnių stiprumo koeficientas
w_i^{User}	i-tojo autoriaus turimo lygio svoris
c_i^{User}	i-tojo autoriaus indėlis į straipsnio rašymą (0; 1] ribose
n	Autorių rašiusių straipsnių kiekis

Taip pat straipsnis negali patekti į viešumą jei jo niekas nerecenzavo. Šitokiu būdu atliekamas kokybiškumo filtravimas, sudaromos galimybės kontroliuoti straipsnių moralės ir turiningumo normas. Straipsniui recenzuoti atsitiktinai parenkami keli pasitikėjimą turintys registruoti vartotojai (atrankai įtakoja vartotojo turimas lygis). Į bendrą pradinio straipsnio formulę įeina ir recenzijų vidurkis:

$$w^{Reviews} = \frac{\sum_{j=0}^m e_j^{User}}{m} \quad (22)$$

Čia:

$w^{Reviews}$	Vidutinis recenzijų įvertinimas
e_j^{User}	j-tojo recenzento įvertinimas
m	Recenzentų skaičius

Skaičiuojant pradinį straipsnio reitingą įvertinami trys aspektai: autorių lygis, recenzijų įvertinimai bei svoris suteikiamas recenzijų įvertinimui:

$$ir^{Article} = w^{Users} + w^{Reviews} * re_{weight}^{ReviewsEvaluation} \quad (23)$$

Čia:

$ir^{Article}$	Pradinis straipsnio reitingas
w^{Users}	Vidutinis autorių rašiusių straipsni stiprumo koeficientas
$w^{Reviews}$	Vidutinis recenzijų įvertinimas
$re_{weight}^{ReviewsEvaluation}$	Reitinguojamu elementų recenzijų įvertinimo svorio koeficientas

Pasikeitus straipsni rašiusių autorių lygiui ar periodui pradinis straipsnio reitingas nesikeičia, nes jo loginė vertė išlieka ta pati.

Atliekant paprastesnius veiksmus, tokius kaip straipsnio perskaitymas, išvertimas į kitą kalbą ar kitus, prie straipsnio atlikto reitingavimo elemento istorijos prisumuojamas vienetas ir perskaičiuojamas straipsnio galutinis reitingas įvertinus pasikeitimus.

3.2 Duomenų saugojimo struktūra

Svarbiausia reitingavimo modelio dalis yra duomenų saugojimo būdai. Kadangi renkama informacija yra labai svarbi tiek svetainės darbuvi tiek vartotojams, turime užtikrinti duomenų saugumą, išpildomumą bei atstatomąją informaciją. Šiuos aspektus galima realizuoti naudojantis duomenų bazių saugyklomis. Informaciją galima saugoti duomenų bazės lentelėse, veiksmingumą praplėsti naudojantis saugomomis duomenų bazėje procedūromis bei nustatant reikiamus apribojimus duomenų laukų išpildymui.

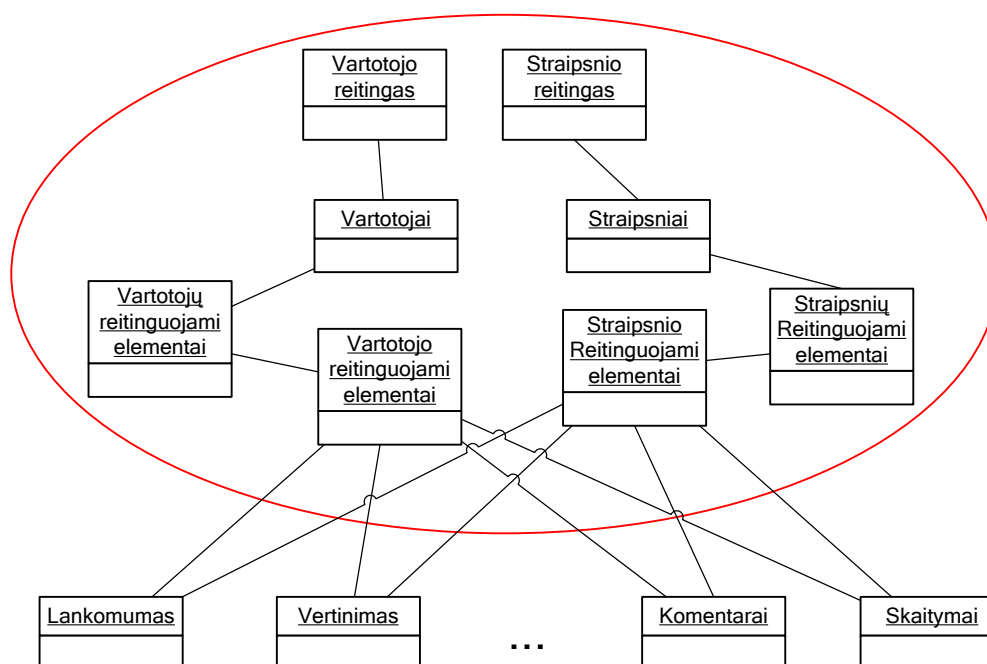
Nuo DB lentelių išdėstymo struktūros labai priklauso visos sistemos veikimo galimybės. Kadangi mes apsibrėžiame turėti lankstų ir nesunkiai išplečiamą struktūros panaudojamumą, turime kurti objektinę DB.

Standartinėje žinių svetainės sistemoje dalyvauja du pagrindiniai reitingavimo vienetai:

- Vartotojai

- Rašiniai/straipsniai

Jei kuriame reitingavimo sistemą tik šiems dviems objektams, pakanka tokios DB struktūros, kaip pateikta **13 pav.**



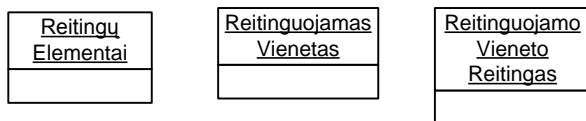
13 pav. Paprasta dviejų objektų DB schema

Naudojantis tokia schema atsirastų nepatogumų realizuojant kiekvieno atskiro reitingavimo vieneto reitingų apskaičiavimą. Taip pat atsiradus poreikiui praplėsti reitingavimo objektus atsiranda reikalingumas naujai pildyti DB. Formaliai pateikiant – auga sistemos sudėtingumo lygis. Jei p yra DB procedūrų skaičius, reikalingas apdoroti vienam reitingavimo vienetui, t - apdorojamų vienetų skaičius, gauname, jog duomenų bazės schemos sudėtingumas nusakomas priklausomybe:

$$s = p * t \quad (24)$$

Norėdami išvengti šitokio sistemos sudėtingumo, turime objektizuoti reitinguojamus vienetus. Reikia sukurti tokią struktūrą, kad vartotojų, straipsnių ir visų kitų reitinguojamų vienetų reitingai ir reitingų istorija būtų saugoma tose pačiose lentelėse. Pačių idėjinių lentelių, tokių kaip „Straipsniai“ kur saugomas straipsnio tekstas, „Skaitymai“ kur saugoma vartotojo perskaityto straipsnio istorija ir kt. apjungti negalime, tačiau renkamos reitingo informacijos struktūra galime suvienodinti. Bendrai galimas apjungti sritis galima matyti **14 pav.** apibrėžtuose objektuose.

Kadangi skirtingų reitinguojamų vienetų vertinimo kriterijai gali skirtis, turime tai įvertinti ir palikti tėvinio objekto atsekamumą. Visų reitinguojamų vienetų bendresnės aibės pateikiamos **14 pav.**



14 pav. Bendros reitinguojamų objektų aibės

Reitinguojamo vieneto aibė nusako reitinguojamą objektą: vartotoją, straipsnį, komentarą ir t.t. Reitingų elementų aibėje saugoma aspektai, pagal kuriuos galima reitinguoti reitinguojamus vienetus. Šie reitingų elementai skiriasi visiems reitinguojamiems vienetams, pvz. vartotojas gali turėti reitingą už straipsnio parašymą, o straipsnis gali turėti reitingą už tai, kiek žmonių jį perskaitė.

Įgyvendindami šią DB logiką išvengiame papildomų lentelių kūrimo atsiradus naujiems reitinguojamiems vienetams – taip atsiranda galimybė lankstumui ir nesudėtingam praplečiamumui.

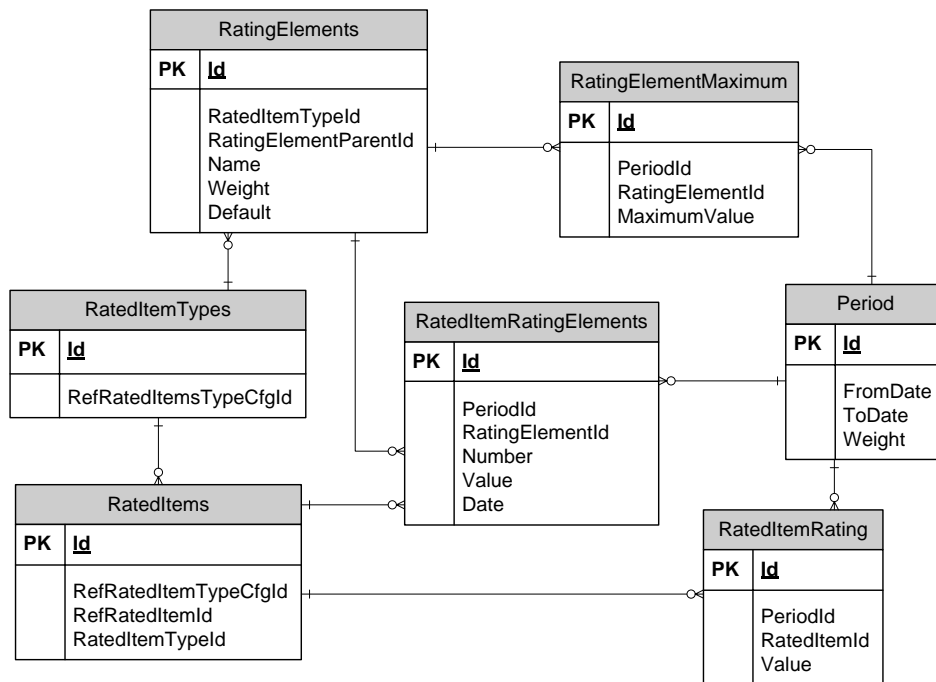
Norėdami išvengti senų objektų monopolistiškumo į DB struktūrą galime įtraukti laiko periodų lentelę, kurioje saugomi laiko periodai su atitinkamais periodų svoriais.

Norėdami saugoti reitinguojamą istoriją, turime naudoti dar vieną objektinę lentelę, kurioje saugoma visa reitinguojamų objektų reitinguotų elementų istorija. Šioje lentelėje būtų galima matyti koks reitinguojamas objektas, koku laiko periodu, už kokį reitinguotą elementą kiek gavo balų ar taškų.

Norint sumažinti DB apkrautumą darbo momentu išvengiant aritmetinių bei paieškos veiksmų galima sukurti lentelę, kurioje būtų saugomas galutinis reitinguojamo vieneto reitingas.

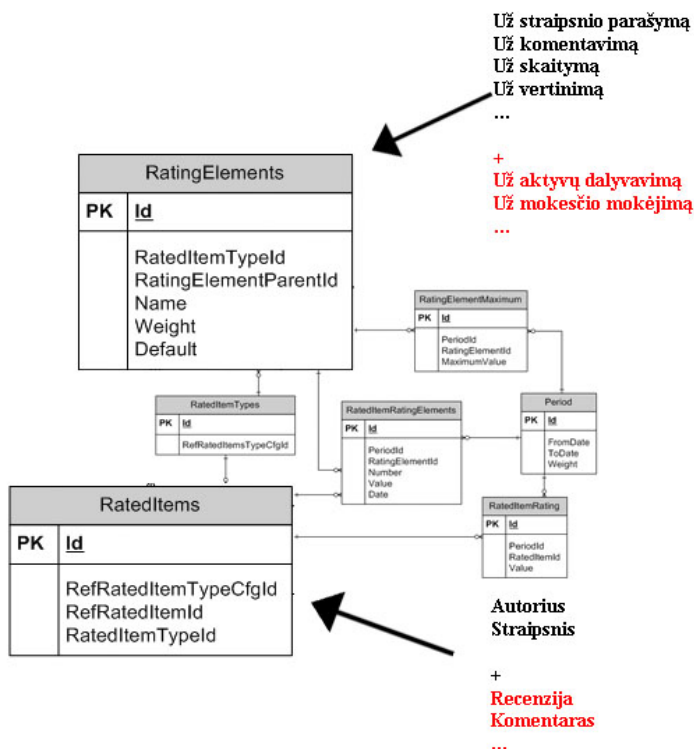
Tam, kad realizuoti reitinguojamų elementų svorių įtaką galutiniam rezultatui, kurio režiai turi būti $[0; 1]$ ribose, turime įtraukti į struktūrą dar vieną lentelę su tikslu laikyti apsibrėžtus reitinguojamų elementų maksimumus.

Remiantis anksčiau minėtais realizavimo aspektais, sukurta reitingavimo DB lentelių struktūra turi atrodyti taip, kaip pateikta **15 pav.**



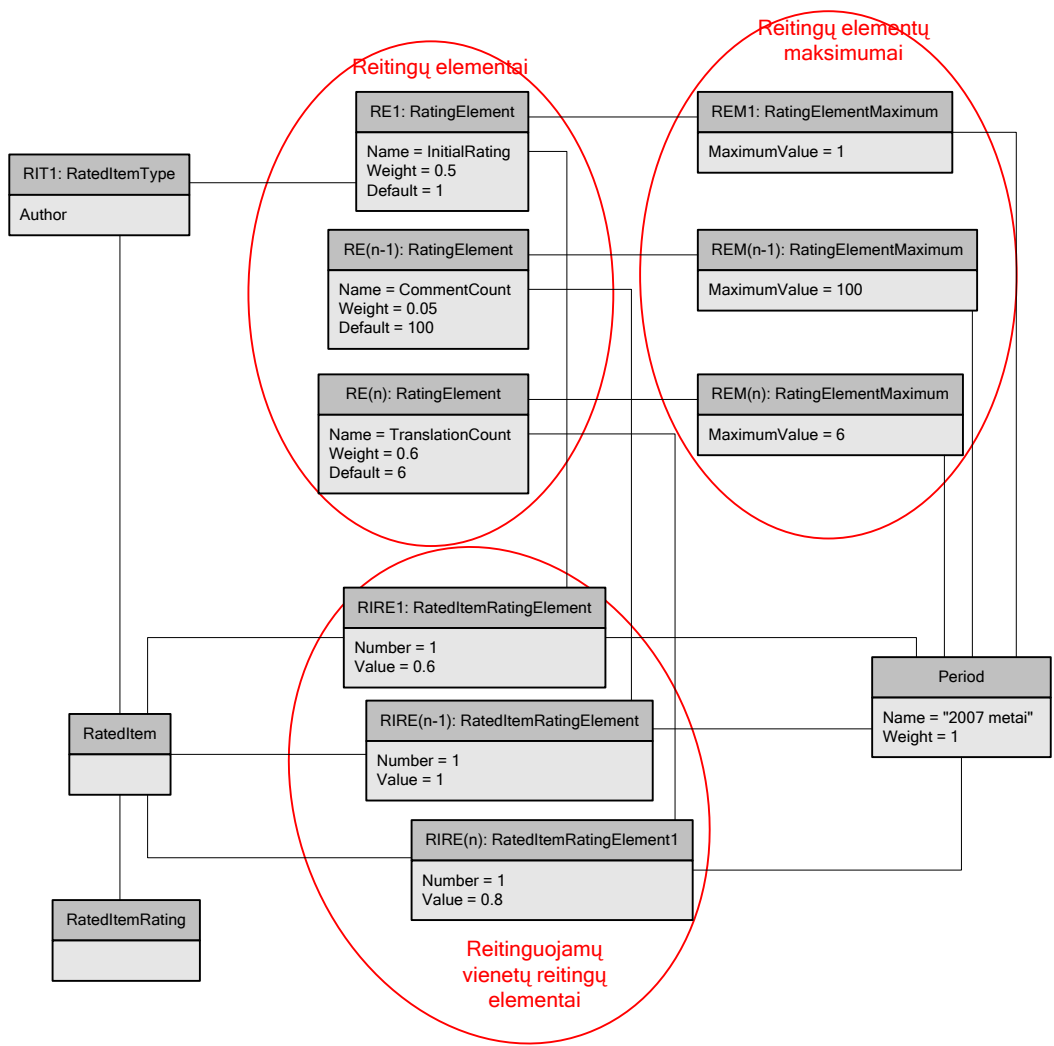
15 pav. Reitingavimo DB lentelių šakinė struktūra

Reitingų elementai bei reitinguojami vienetai saugomi atitinkamosse lentelėse suteikia lankstumą norimai pakeisti jų aibę, pridėti papildomų reitinguojamos svetainės reitingavimo galimybių ar panaikinti esamus **16 pav.**



16 pav. Reitingavimo DB lentelių galimybė norimam išplečiamumui

Pilnai išpildyta reitinguojamo vieneto reitinguojamų elementų struktūra turėtų atrodyti taip kaip pateikta **17 pav.**



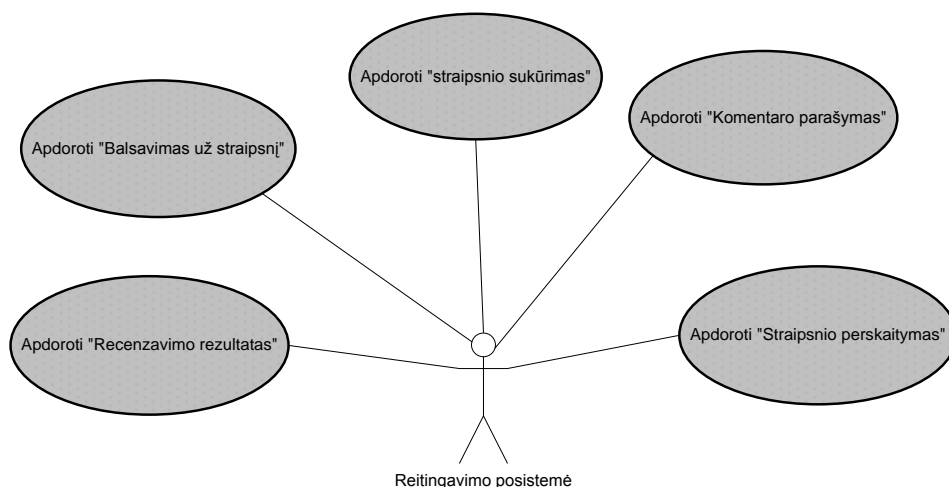
17 pav. Reitingavimo posistemio DB šablonas

4 Eksperimentinis reitingavimo modelio tyrimas

Naudojantis aukščiau apibrėžtomis teorijomis sukurtas eksperimentinis reitingavimo sistemos modelis. Modelis sukurtas naudojantis asp.NET 2.0 technologijomis, kurios suteikia dideles vartotojiškos sąsajos galimybes.

4.1 Panaudos atvejai

Reitingavimo posistemė rinkdama informaciją atlieka pagrindinius veiksmus, kurie pateikti **18 pav.** Tai recenzavimo, komentarų rašymo, straipsnių balsavimo, sukūrimo bei skaitymų apdorojimas



18 pav. Reitingavimo posistemio pagrindiniai panaudos atvejai

Visi aukščiau aprašyti panaudos atvejai smulkiau nagrinėjami žemiau pateiktose **1-5 lentelėse:**

1 lentelė. Apdoroti „Straipsnio sukūrimas“

1. PANAUDOS ATVEJIS	Apdoroti „Straipsnio sukūrimas“
Vartotojas/Aktorius	Reitingavimo posistemė
Aprašas	Sukūręs straipsnį vartotojas gauna papildomų taškų. Atitinkamai pagal sukurto straipsnio tipą galimi skirtingi taškų įverčiai
Prieš sąlyga	Vartotojas parašo straipsnį
Sužadinimo sąlyga	Valdymo posistemė pateikia informaciją apie naujo straipsnio sukūrimą
Po-sąlyga	Reitingų modulis pagal atitinkamą straipsnio tipą priskiria naujus taškus vartotojui, kuris jį sukūrė bei nustato paties straipsnio pradinį įvertį

2 lentelė. Apdoroti „Balsavimas už straipsnį“

2. PANAUDOS ATVEJIS	Apdoroti „Balsavimas už straipsnį“
Vartotojas/Aktorius	Reitingavimo posistemė
Aprašas	Balsuodamas už straipsnį balsuotojas gauna taškų už aktyvumą bei straipsnio įvertis padidėja
Prieš sąlyga	Turi būti publikuotas straipsnis
Sužadinimo sąlyga	Valdymo posistemė pateikia informaciją, kad buvo atliktas balsavimas atitinkamo vartotojo už atitinkamą straipsnį atitinkamu taškų skaičiumi
Po-sąlyga	Reitingavimo sistema pagal įverčius prideda prie atitinkamų reitinguojamų objektų atitinkamas verčių reikšmes

3 lentelė. Apdoroti „Recenzavimo rezultatas“

3. PANAUDOS ATVEJIS	Apdoroti „Recenzavimo rezultatas“
Vartotojas/Aktorius	Reitingavimo posistemė
Aprašas	Vartotojas recenzavęs straipsnį siunčia atsakymą IT portalui. To pasekoje už sėkmingą recenzavimą jis gauna taškus
Prieš sąlyga	Vartotojas turi būti gavęs straipsnį recenzavimui
Sužadinimo sąlyga	Valdymo posistemė pateikia informaciją, jog buvo gautas recenzuotas straipsnis
Po-sąlyga	Reitingavimo posistemė priskiria papildomus taškus recenzuotojui bei, pagal įvertinimą, straipsnio autoriui

4 lentelė. Apdoroti „Komentaro parašymas“

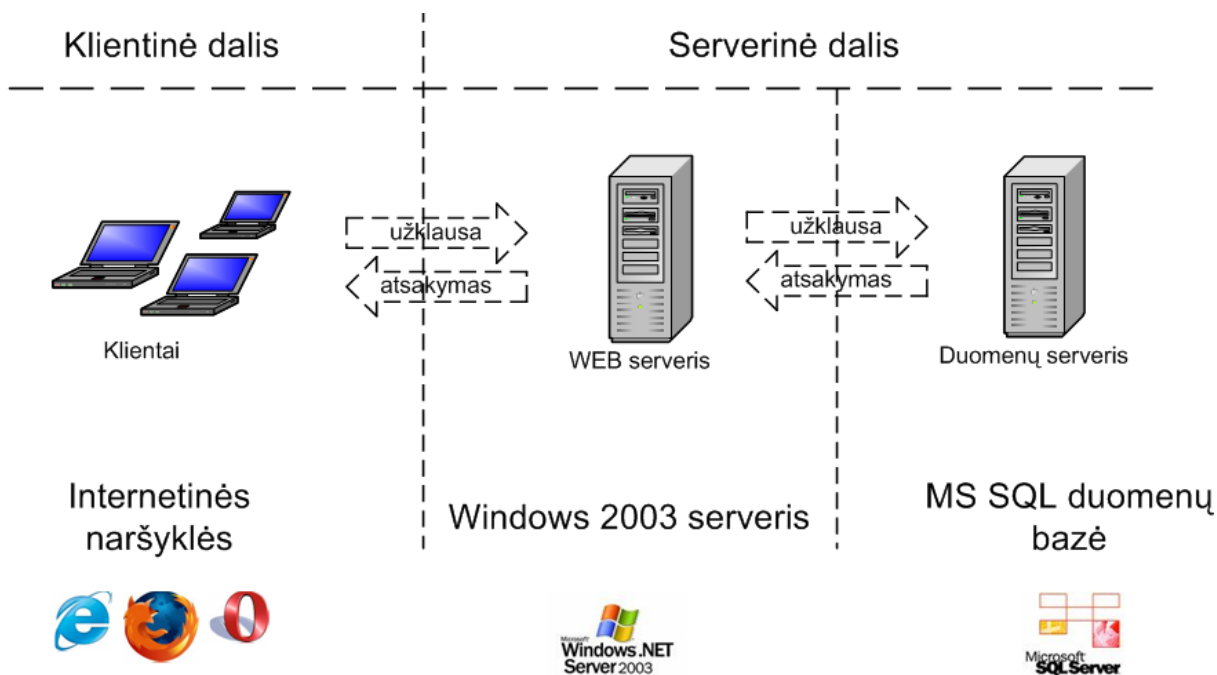
4. PANAUDOS ATVEJIS	Apdoroti „Komentaro parašymas“
Vartotojas/Aktorius	Reitingavimo posistemė
Aprašas	Už aktyvų komentavimą, vartotojas taip pat gali gauti papildomų taškų
Prieš sąlyga	Vartotojas parašo komentarą
Sužadinimo sąlyga	Valdymo posistemė pateikia informaciją apie komentuotojo pasireiškimą
Po-sąlyga	Komentuotojui skiriamas atitinkamas tašku kiekis

5 lentelė. Apdoroti „Straipsnio perskaitymas“

5. PANAUDOS ATVEJIS	Apdoroti „Straipsnio perskaitymas“
Vartotojas/Aktorius	Reitingavimo posistemė
Aprašas	Perskaičius straipsnį pakyla jo vertė. Taip pat galima kaupti informaciją apie aktyvius vartotojus, kurie šviečiasi ir domisi
Prieš sąlyga	Vartotojas perskaito kito vartotojo parašyta straipsnį
Sužadinimo sąlyga	Valdymo posistemė pateikia informaciją apie perskaitytą straipsnį ir vartotoją, kuris tą straipsnį perskaitė
Po-sąlyga	Reitingavimo posistemė skiria taškus straipsniui, straipsnio autoriui, bei skaitovui

4.2 Architektūrinio išdėstymo vaizdas

Visos sistemos architektūrinis išdėstymo vaizdas pateiktas **19 pav.** Vartotojas turi turėti asmeninį kompiuterį su galimybe prisijungti prie interneto. Taip pat jo kompiuteryje turi būti internetinė naršyklė, tokia kaip Internet Explorer, Firefox. Dėl savo plačių vartotojo sąsajos kūrimo galimybių naudojamos asp.NET technologijos, kurioms palaikyti reikalingas Windows 2003 serveris. Duomenų baze pasirinkta MS SQL produkcija. Šis serveris turi dideles panaudojimo galimybes bei tiksliausiai dirba su Microsoft programavimo įrankiais.



19 pav. Išdėstymo vaizdas

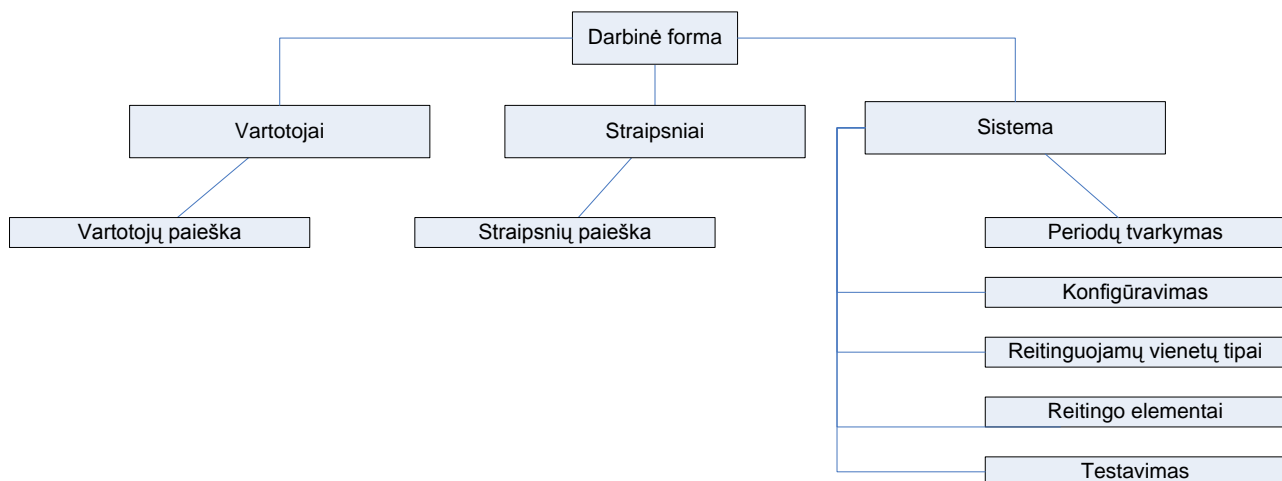
Didžiausia veikimo kokybė priklausys nuo turimo interneto greičio bei asmeninio kompiuterio techninių parametrų.

4.3 Administratoriaus sąsaja

Sistemos administratoriaus sąsajos komponentas užtikrina patikimą duomenų naudojimą. Administratorius gali lengvai naršyti po meniu ir matyti reikiamą informaciją. Norimus duomenis gali išsaugoti ataskaitų pavidalu.

- Sąsaja kontroliuoja vartotojo klaidos įvedimo pasirodymą.
- Su tiesioginiais duomenimis sąsaja turi ryšį per kontrolerius.
- Sąsajos duomenys tikrinami, žiūrimas informacijos korektiškumas.

Administratoriaus prieinama sąsajos struktūra matoma **20 pav.** Jis gali patogiai prieiti prie reitinguojamų vienetų (vartotojų, straipsnių), lengvai juos peržiūrėti, atlikti paiešką, keisti esminius duomenis, įvertinimus. Taip pat suteikiama galimybė keisti sisteminius nustatymus: tvarkyti periodus, jų įverčius, kurti bei keisti reitingo elementus, nustatyti reitingų maksimumo vertes.



20 pav. Administratoriaus sąsajos klasių diagrama

Vartotojo sąsajos prieinamų veiksmų galimybės aprašomos **6-13 lentelėse**. Pateikiamas formos trumpas aprašymas, sąsaja, naudojami resursai bei sąveikavimo savybės.

6 lentelė. Administratoriaus „Login“ forma

Pavadinimas	Login
Klasifikacija	Forma
Aprašymas	Šioje formoje administratorius suveda prisijungimo duomenis ir jungiasi prie sistemos
Sąsaja	Forma pateikiama internetinėje naršyklėje
Resursai	ASP .NET puslapis serveryje.
Sąveikavimas	Paspaudus mygtuką „Jungtis“, koduoti duomenys perduodami vartotojų kontrolieriui

7 lentelė. Administratoriaus „Vartotojų paieška“ forma

Pavadinimas	User_filter
Klasifikacija	Forma
Aprašymas	Šioje formoje administratorius suveda ieškomo vartotojo vardą arba pavardę ir atlieka filtravimą. Iš visų gautų vartotojų išsirenka norimą vartotoją. Pasirinkęs vartotoją gali matyti jo surinktus reitingus bei juos keisti. Taip pat gali keisti vartotojo lygį.
Sąsaja	Forma pateikiama internetinėje naršyklėje
Resursai	ASP .NET puslapis serveryje.
Sąveikavimas	Pasirinkus parinktį „ieškoti“ formoje pateikiami duomenys atitinkantys paieškos formą. Pasirinkus parinktį „atnaujinti“ pakeista informacija užsaugoma duomenų bazėje.

8 lentelė. Administratoriaus „Straipsnių paieška“ forma

Pavadinimas	Article_filter
Klasifikacija	Forma
Aprašymas	Šioje formoje administratorius suveda ieškomo straipsnio pavadinimą ir atlieka filtravimą. Iš visų gautų straipsnių išsirenka norimą. Čia jis gali matyti straipsnio surinktus reitingus bei juos keisti.
Sąsaja	Forma pateikiama internetinėje naršyklėje
Resursai	ASP .NET puslapis serveryje.
Sąveikavimas	Pasirinkus parinktį „ieškoti“ formoje pateikiami duomenys atitinkantys paieškos formą. Pasirinkus parinktį „atnaujinti“ pakeista informacija užsaugoma duomenų bazėje.

9 lentelė. Administratoriaus „Periodų tvarkymas“ forma

Pavadinimas	Period
Klasifikacija	Forma
Aprašymas	Formoje matomi jau sukurti periodai. Sukurtus periodus galima ištrinti. Galima sukurti naujus periodus. Sukurtiems ir esamiems periodams galima nustatyti svorio koeficientus
Sąsaja	Forma pateikiama internetinėje naršyklėje
Resursai	ASP .NET puslapis serveryje.
Sąveikavimas	Pasirinkus parinktį „sukurti“ pateikiama naujo periodo sukūrimo forma. Pasirinkus parinktį „atnaujinti“ pakeista informacija užsaugoma duomenų bazėje.

10 lentelė. Administratoriaus „Konfigūravimas“ forma

Pavadinimas	Configuration
Klasifikacija	Forma
Aprašymas	Formoje galima sukurti sistemos konfigūracinius duomenis. Tai bus pagrindiniai sistemos nustatymai, pagrindiniai vartotojų tipai, pagrindiniai straipsnių reitingavimo elementai, pagrindiniai vartotojų reitingavimo elementai.
Sąsaja	Forma pateikiama internetinėje naršyklėje
Resursai	ASP .NET puslapis serveryje.
Sąveikavimas	Pasirinkus parinktį „sukurti“ pateikiama nustatymo sukūrimo forma. Pasirinkus parinktį „atnaujinti“ pakeista informacija užsaugoma duomenų bazėje.

11 lentelė. Administratoriaus „Reitinguojamų vienetų tipai“ forma

Pavadinimas	Rated items
Klasifikacija	Forma
Aprašymas	Formoje galima sistemai priskirti kokie bus reitinguojami vienetai. Pagrinde tai bus vartotojai ir straipsniai.
Sąsaja	Forma pateikiama internetinėje naršyklėje
Resursai	ASP .NET puslapis serveryje.
Sąveikavimas	Pasirinkus parinktį „priskirti“ pasirinktas reitingavimo vienetas įrašomas į reitingavimo vienetų lentelę.

12 lentelė. Administratoriaus „Reitingo elementai“ forma

Pavadinimas	Rating elements
Klasifikacija	Forma
Aprašymas	Formoje galima sistemai priskirti kokie bus reitingo elementai. Atitinkamai leidžiama pasirinkti kokiam reitinguojamam vienetui bus priskiriami elementai. Vartotojui priskiriami konkrečiai vartotojui tinkantys reitingo elementai iš positions sąrašo, o straipsniui analogiškai – straipsniui tinkantys
Sąsaja	Forma pateikiama internetinėje naršyklėje
Resursai	ASP .NET puslapis serveryje.
Sąveikavimas	Pasirinkus parinktį „priskirti“ pasirinktas reitingo elementas įrašomas į reitingo elementų lentelę.

13 lentelė. Administratoriaus „Testavimas“ forma

Pavadinimas	Testing
Klasifikacija	Forma
Aprašymas	Formoje administratorius galės pabandyti atlikti atitinkamus procedūrinius veiksmus kuriuos vėliau atlikinės duomenų bazės procedūros su pateiktais duomenimis. Atliekant šias procedūras testavimo formoje bus pateikiami atitinkami pranešimai apie sėkmingą arba nesėkmingą procedūros prasisukimą. Informacija sukurta testavimo metu iš kart po testavimo bus atstatoma. Testavimas atliekamas tik jau su duomenų bazėje esančiais sukurtais įrašais.
Sąsaja	Forma pateikiama internetinėje naršyklėje
Resursai	ASP .NET puslapis serveryje.
Sąveikavimas	Pasirinkus parinktį „testuoti“ atliekama pasirinkta procedūra ir pateikiamas rezultatas ekrane.

Teisingam duomenų paskirstymui ir valdymui yra naudojami klasių kontroleriai. Kiekvienas kontroleris turi jam būdingo objekto elementus ir galimus veiksmus su jais. Tokiu būdu išlaikomas duomenų stabilumas bei korektiškas panaudojamumas. Kontrolerių aprašai pateikti **14-20 lentelėse**. Kaip ir sąsajos elementams, lentelėse nusakomas trumpas kontrolerio aprašymas, naudojami resursai bei sąveikavimo savybės.

14 lentelė. Kontroleris „Vartotojas“

Pavadinimas	Vartotojas
Klasifikacija	Kontroleris
Aprašymas	Klasė atsakinga už vartotojo prisijungimą prie sistemos bei vartotojo duomenų pateikimą. Atlieka informacijos tikrinimą. Atsiradus neatitikimams grąžinamas atitinkamas pranešimas.
Sąsaja	Kontroleris su forma bendrauja per HTTP POST. Su DB bendrauja per DB sąsają.
Resursai	ASP Serverio darbas. DB darbas.
Sąveikavimas	Login formoje paspaudus prisijungimo mygtuką, informacija pateikiama vartotojo kontrolieriui. Kontroleris iškviečia prisijungimo objektą, kuris tikrina įvestų duomenų korektiškumą. Users_rating formoje pasirinkus vartotojo duomenų peržiūrėjimą, koregavimą iškviečiami „set“ arba „get“ objektai, kurie atlieka transakcijas su DB.

15 lentelė. Kontroleris „Reitingu_naudojimas“

Pavadinimas	Reitingu_naudojimas
Klasifikacija	Kontroleris
Aprašymas	Klasė atsakinga už reitingų valdymo formoje pakeistos informacijos įgyvendinimą DB.
Sąsaja	Kontroleris su forma bendrauja per HTTP POST. Su DB bendrauja per DB sąsają.
Resursai	ASP Serverio darbas. DB darbas.
Sąveikavimas	Formoje pasirinkus atnaujinimo mygtuką, išskviečiamas reitingų atnaujinimo objektas, gauti duomenys tikrinami ar jie atitinka reikalavimus ir siunčiama informacija į DB.

16 lentelė. Kontroleris „Recenzijos“

Pavadinimas	Recenzijos
Klasifikacija	Kontroleris
Aprašymas	Vartotojui parašius recenziją, recenzijos kontroleris gauna visą informaciją apie atliktus vartotojo veiksmus: kada recenzija pateikta, kada atsiųsta, koks vartotojas, kokio lygio recenzija. Iš turimų duomenų skaičiuojama balų skaičius, kuris vėliau išsaugomas prie atitinkamo vartotojo.
Sąsaja	Kontroleris su forma bendrauja per HTTP POST. Su DB bendrauja per DB sąsają.
Resursai	ASP Serverio darbas. DB darbas.
Sąveikavimas	Kontroleris bendrauja su išorine straipsnių recenzavimo posisteme. Ta posistemė esant recenzijos siuntimo ar gavimo įvykiui išskviečia recenzijos kontrolerį ir perduoda visą reikiamą informaciją. Tarpiniai ir galutiniai duomenys išsaugomi DB.

17 lentelė. Kontroleris „Skaitymas“

Pavadinimas	Skaitymas
Klasifikacija	Kontroleris
Aprašymas	Vartotojui perskaičius straipsnį, skaitymo kontroleris gauna informaciją apie skaitantįjį. Iš turimų duomenų skaičiuojama balų skaičius, kuris vėliau išsaugomas prie atitinkamo vartotojo ir straipsnio bei straipsnio autoriaus.
Sąsaja	Kontroleris su forma bendrauja per HTTP POST. Su DB bendrauja per DB sąsają.
Resursai	ASP Serverio darbas. DB darbas.
Sąveikavimas	Kontroleris bendrauja su išorine straipsnių skaitymo posisteme. Ta posistemė esant straipsnio perskaitymo įvykiui išskviečia skaitymo kontrolerį ir perduoda visą reikiamą informaciją. Duomenys išsaugomi DB.

18 lentelė. Kontroleris „Komentaras“

Pavadinimas	Komentaras
Klasifikacija	Kontroleris
Aprašymas	Vartotojui pakomentavus straipsnį, komentavimo kontroleris gauna informaciją apie dalyvį. Iš turimų duomenų skaičiuojama balų skaičius, kuris vėliau išsaugomas prie atitinkamo vartotojo ir straipsnio bei straipsnio autoriaus.
Sąsaja	Kontroleris su forma bendrauja per HTTP POST. Su DB bendrauja per DB sąsają.
Resursai	ASP Serverio darbas. DB darbas.
Sąveikavimas	Kontroleris bendrauja su išorine straipsnių komentavimo posisteme. Ta posistemė esant straipsnio komentavimo įvykiui išskviečia komentavimo kontrolerį ir perduoda visą reikiamą informaciją. Duomenys išsaugomi DB.

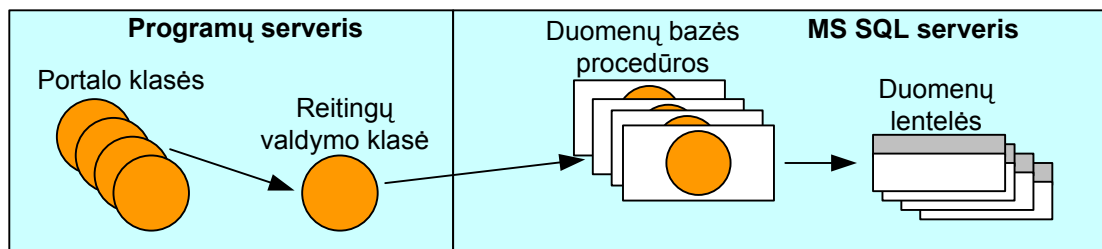
19 lentelė. Kontroleris „Straipsnio_kurimas“

Pavadinimas	Straipsnio_kurimas
Klasifikacija	Kontroleris
Aprašymas	Kai vartotojas patalpina savo straipsnį į ITŽP, išorinė posistemė pateikia informaciją apie talpinantįjį ir apie straipsnį. Pagal straipsnio tipą skaičiuojami atitinkami balai ir priskiriami straipsnį patalpinusiam vartotojui.
Sąsaja	Kontroleris su forma bendrauja per HTTP POST. Su DB bendrauja per DB sąsają.
Resursai	ASP Serverio darbas. DB darbas.
Sąveikavimas	Kontroleris bendrauja su išorine straipsnių kūrimo posisteme. Ta posistemė esant straipsnio patalpinimo įvykiui išskviečia straipsnio kūrimo kontrolerį ir perduoda visą reikiamą informaciją. Duomenys išsaugomi DB.

20 lentelė. Kontroleris „Balsavimas“

Pavadinimas	Balsavimas
Klasifikacija	Kontroleris
Aprašymas	Kiekvienas registruotas vartotojas gali balsuoti už kito vartotojo patalpintą straipsnį. Kontroleriui pateikiama balsuojančiojo ir straipsnio informacija, kuri vėliau verčiama balais ir talpinama prie straipsnio.
Sąsaja	Kontroleris su forma bendrauja per HTTP POST. Su DB bendrauja per DB sąsają.
Resursai	ASP Serverio darbas. DB darbas.
Sąveikavimas	Pagrindiniame puslapyje registruotas vartotojas spaudžia balsavimo mygtuką. Išorinė sąsaja pateikia informaciją balsavimo kontroleriui. Kontroleris patikrina duomenų korektiškumą ir pagal nustatytas reitingų vertes priskiria balus pasirinktam straipsniui.

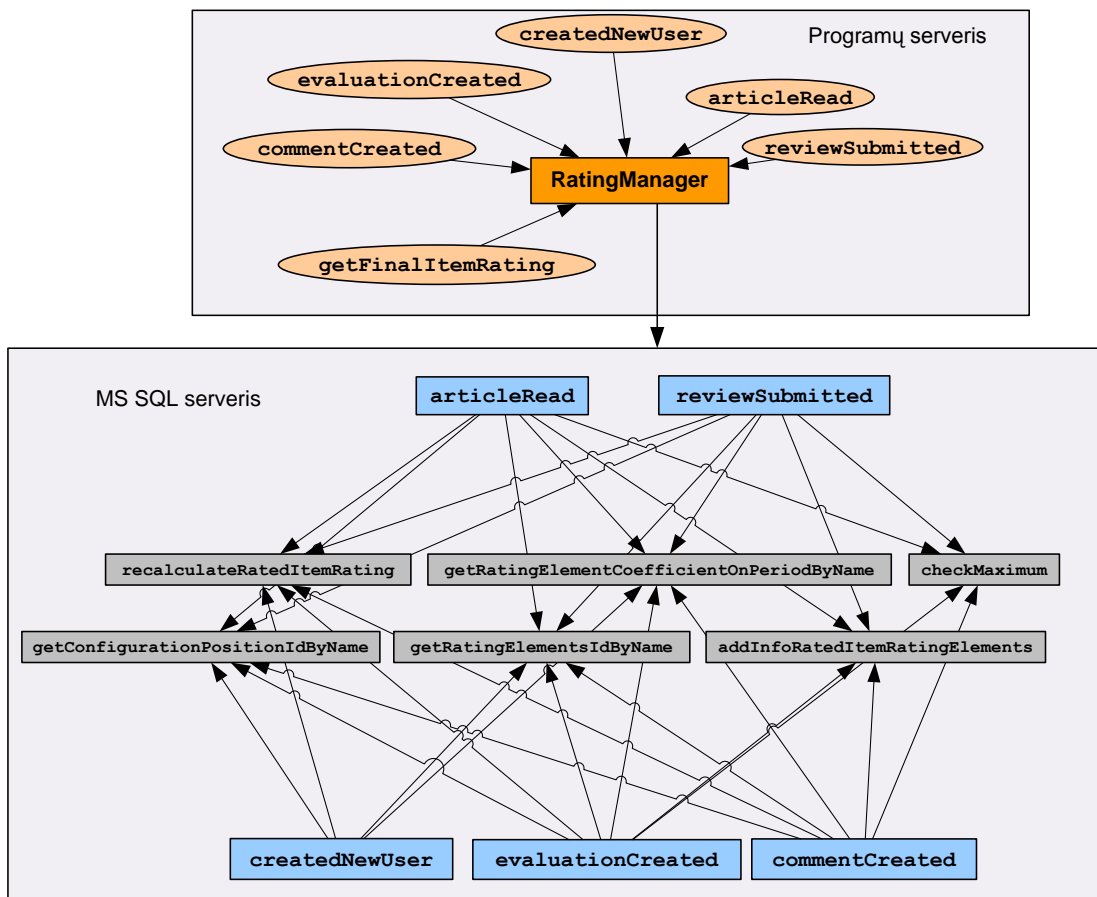
Reitingavimo sistemos funkcinė dalis išdėstyta taip, kad reitingų kontrolierio klasė patalpinta programų serveryje, o kitos internetinio portalo klasės ja naudojasi. Pati reitingo klasė duomenis valdo per SQL serveryje patalpintas duomenų bazės procedūras. Kaip matoma iš **21 pav.** užklausų ryšys yra vienpusis – iniciavimą gali pradėti tik internetinio portalo klasės.



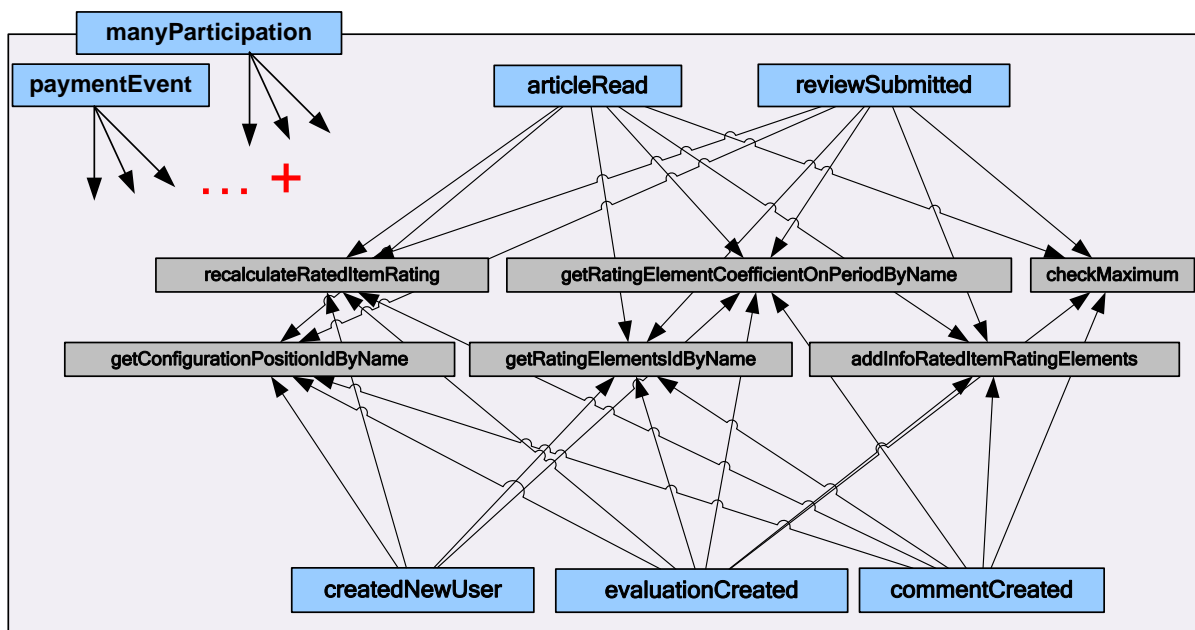
21 pav. Reitingavimo sistemos funkcinė realizacija

Programų serveryje patalpinta RatingManager klasė, kuri laukia iš sistemos išorinių kreipinių ir juos vykdo. Kiekvienas kreipinys turi reikšminę vertę – jis nusako, kokį įvykį reitingavimo posistemiui reikia apdoroti. Tai gali būti ir naujo vartotojo sukūrimas, ir straipsnio perskaitymas, ir įvertinimo sukūrimas ir t.t.

Pagrindinis skaičiavimo darbas atliekamas SQL serveryje. Iš **22 pav.** matome, jog kiekvienas kitas esminis įvykis naudoja tas pačias bazines procedūras – tai patvirtina nesudėtingo išplečiamumo galimybę **23 pav.**



22 pav. Programų ir SQL serverių komponentai



23 pav. Duomenų bazės įvykių procedūrų praplečiamumo galimybės

Administracinė vartotojo sąsaja sukurta taip, jog ja būtų galima paprastai naudotis ir lengvai suprasti. Kadangi administratoriui dažniausiai reikalingi kuo išsamesni duomenys, naudojantis lentelėmis jie pateikiami internetinio portalo puslapyje su galimybėmis lįsti giliau

ir matyti pasirinktų objektų detalesnius vaizdus. Vartotojo paieškos ir valdymo sąsaja pateikta **24 pav.** Joje galima pagal vardą arba slapyvardį ieškoti esamų internetinėje svetainėje užregistruotų vartotojų. Rezultate rodomi vartotojų surinkti reitingai, esamas lygis ir galimybė lįsti giliau prie smulkesnių duomenų.

USERS ARTICLES SYSTEM

User filter

HOME > My ToolBox > Publication

title

Name

Surname

User list

User name	Value	User level	Additional info
Radvila	0,021	..2..	..info..
Ugne	0,022	..3..	..info..
Milda	0,024	..3..	..info..
Algirdas	0,027	..3..	..info..
Jaronimas	0,033	..3..	..info..
Kestas	0,041	..3..	..info..
Vaidotas	0,122	..1..	..info..
Kristina	0,145	..3..	..info..
Vytautas	0,177	..2..	..info..

24 pav. Vartotojo paieškos sąsaja administratoriaus meniu

Konkretaus pasirinkto vartotojo valdymo sąsaja pateikta **25 pav.** Joje galima matyti išskleistus konkretaus vartotojo duomenis, peržiūrėti už ką, kada ir kiek reitingo taškų pastarasis gavo.

User filter

HOME > My ToolBox > Publication

title

Name	Vytautas
Surname	Vytautaitis
Date of birth	1980.01.01 00:00:00
Level	2
Value	0,17683333158493

Rated item ratings

Period: 2007 metai Time interval [2007.01.01 00:00:00-2008.01.01 00:00:00] Weight: 1 Default: 0			
Rating element	Number	Value	Date
ArticleAuthorCountContribution	<u>4</u>	0	2007.01.23 12:21:51
InitialRating	<u>1</u>	0,01	2007.01.23 12:21:51
CommentCount	<u>15</u>	0,0075	2007.01.23 12:21:51
TranslationCount	<u>2</u>	0,03333334	2007.01.23 12:21:51
EvaluationCount	<u>2</u>	0,006	2007.01.23 12:21:51
ReviewCount	<u>3</u>	0,12	2007.01.23 12:21:51
Total: 0,1768333			

25 pav. Konkretaus vartotojo duomenų valdymo meniu

Analogiškai kaip ir vartotojams, yra sukurta ir straipsnių paieškos forma, kuri parodyta **26 pav.** Joje galima taip pat atlikti paiešką pagal straipsnio pavadinimą, matyti straipsnio reitingą bei galimybę lįsti į gilesnį duomenų nagrinėjimą.

Articles filter

HOME > My ToolBox > Publication

title

Title

Filter >>

Articles list

Name	Value	Info
Diferencialas	0,057	..info..
CD ROM	0,438	..info..
MAC	0,281	..info..
ip v6	0,058	..info..
Laidumas	0,061	..info..
qwerty	0,054	..info..
Laidininkas	0,05	..info..

26 pav. Straipsnio paieškos forma

Pasirinkto straipsnio valdymo meniu parodytas **27 pav.** Matoma kada, už ką ir kiek reitingavimo taškų gavo pasirinktas straipsnis.

USERS	ARTICLES	SYSTEM
--------------	-----------------	---------------

Articles filter

HOME > My ToolBox > Publication

title

Rated item ratings

Period: 2007 metai Time interval [2007.01.01 00:00:00-2008.01.01 00:00:00] Weight: 1 Default: 0			
Rating element	Number	Value	Date
ReviewsEvaluation	2	0,3333333	2007.01.23 12:21:51
TranslationCount	1	0,008333334	2007.01.23 12:21:51
CommentCount	78	0,039	2007.01.23 12:21:51
EvaluationCountValue	2,3	0,0069	2007.01.23 12:21:51
InitialRating	1	0,05	2007.01.23 12:21:51
ReadingCountValue	5	0,0005	2007.01.23 12:21:51
Total: 0,4380667			

27 pav. Pasirinkto straipsnio valdymo meniu

Administratorius turi galimybę tvarkyti visus sisteminius nustatymus. Jam šią galimybę palengvina sukurta vartotojo sąsaja, parodyta **28 pav.** Čia pateikiama reitingavimo elementų maksimumo reikšmių keitimo bei sukūrimo forma.

USERS	ARTICLES	SYSTEM
--------------	-----------------	---------------

Periods Configuration Rated item types Rating elements Testing

HOME > My ToolBox > Publication

title

Choose rating element type

Rating elements

Name	Weight	Default	Action	Max
ArticleAuthorCountContribution	0	20	..configure..	..maximum..
CommentCount	0,05	100	..configure..	..maximum..
EvaluationCount	0,3	100	..configure..	..maximum..
InitialRating	0,01	1	..configure..	..maximum..
ReviewCount	0,4	10	..configure..	..maximum..
TranslationCount	0,05	3	..configure..	..maximum..

28 pav. Administratoriaus sisteminių nustatymų sąsaja

5 Pasiūlytojo reitingavimo modelio galimybių palyginimas su kitais modeliais

Išanalizavus egzistuojančius reitingavimo modelius bei palyginus jų savybes su sukurtu modeliu, matosi, jog pastarasis išpildo aibę užsibrėžtų tikslų, tokių kaip reitinguojamų vienetų konkurencingumas, lengvai praplečiama reitingavimo vienetų bei elementų sritis, nereikalinga administratoriaus priežiūra ir keletas kitų naudingų aspektų, kurie išvardinti **21 lentelėje**. Taip pat matoma, jog modelis turi ir neigiamų savybių, tokių kaip sudėtingas pradinis įdiegimas. Galima prieiti prie išvados, jog sudėtinga sukurti gerą ir tuo pačiu labai paprastą reitingavimo sistemą.

21 lentelė. Reitingavimo modelių savybių palyginimas.

	Tiesinis reitingavimas	Vidurkio reitingavimas	Konsensuso reitingavimas	Elo klasės reitingavimas	Sukurtos sistemos reitingavimas
Aiški reiting. vieneto kokybė	○	●	●	●	●
Reitingas tik kyla	●	○	○	○	●
Reitinguojamo vieneto stabilumas	○	●	●	●	●
Algoritmas sunkiai perprantamas	○	○	○	●	●
Nereikalinga admin. priežiūra	●	●	○	●	●
Sunku pasiekti reitingo viršūnę	●	○	○	○	●
Lengvai praplečiama reiting. sritis	●	●	○	○	●
Lengvas skaičiavimo algoritmas	●	●	●	○	●
Tinkamas visoms probl. sritims	●	●	○	○	○
Paprastas įdiegimas	●	●	●	○	○

6 Išvados

1. Nagrinėdami kitų internetinių puslapių naudojamus reitingavimo modelius matome, jog dominuoja tiesiniai bei vidurkio skaičiavimo principai. Šie modeliai yra paprasti realizuoti tačiau nėra lankstūs kintamumui bei vartotojui lengvai perprantami. Šie modeliai tinkami nedidelėms internetinėms svetainėms, kur reitingavimas nėra sureikšminamas arba svetainėms, kuriose nėra techninių galimybių sudėtingiems reitingavimo algoritmams realizuoti.
2. Elo ir Glicko reitingavimo modeliai skiria didesnę dėmesį vartotojo reitingavimo tikslumui. Juose iš gautų reitingo rezultatų galima nustatyti reitinguojamo vieneto vertę, taip pat įvertinti realybėje egzistuojančius įvykius tokius kaip aktyvumas, pastovumas. Šie aspektai yra reikalingi gerai reitingavimo sistemai, tačiau Elo ir Glicko reitingavimo modelių šeimos yra skirtos tarpusavyje besivaržančių žmonių vertinimui.
3. Geram žinių svetainės reitingavimo modeliui siekiama paprasto realizavimo su galimybe kuo mažiau apkrauti sisteminius resursus, tačiau pageidaujama gero interaktyvumo su vartotoju. Iš išanalizuotų egzistuojančių reitingavimo modelių išsirinktos teigiamai vertinamos savybės, kurios pritaikytos internetiniam žinių portalui: vidurkio – santykio skaičiavimai; tiesinio – reitingo augimo dėsnis; Elo – laiko bei kitų faktorių įtaka galutiniam rezultatui.
4. Siūlomame reitingavimo modelyje vyrauja santykiniai reitingų skaičiavimai, o svertinių koeficientų naudojimas sistemos gyvumo bruožus. Modelio skaičiavimo algoritmuose įtraukiami apibendrinimo parametrai, kurie leidžia praplėsti reitinguojamų vienetų aibę nekeičiant paties apskaičiavimo proceso. Modelio algoritmo parametrai ir jų naudojimas atspindi duomenų kaupimo struktūros vaizdą. Sukurta atitinkama duomenų bazės schema, kuri suteikia galimybę papildyti reitinguojamų vienetų bei elementų aibę nekeičiant esamų architektūrinių sprendimų.
5. Visi šie aprašyti metodai išbandyti sukuriant eksperimentinę reitingavimo sistemą, ją įdiegiant į žinių svetainės bendrą struktūrą. Panaudojamumo galimybės atitinka apsibrėžtus reikalavimus.
6. Reitingavimo tematika parengtas straipsnis, kuris buvo pateiktas bei pristatytas tarpuniversitetinei magistrantų ir doktorantų konferencijai.

7 Naudojamos literatūros sąrašas

- [1] Elo rating system. Kovas, 2006, *Wikipedia*, [žiūrėta 2007.02.24], prieiga per internetą: <http://en.wikipedia.org/wiki/Elo_rating_system>
 - [2] **E. Weisstein**. Logistic distribution. Balandis, 2003, [žiūrėta 2007.02.25]. Prieiga per internetą: <<http://mathworld.wolfram.com/LogisticDistribution.html>>
 - [3] **E. Weisstein**. Normal Distribution. Birželis, 2004, [žiūrėta 2007.03.04], prieiga per internetą: <<http://mathworld.wolfram.com/NormalDistribution.html>>
 - [4] **F. Ruth**. Black-Box Software Testing using Module Interface Specifications. Kovas, 1993.
 - [5] **J. T. Anderson; P. Brown; J. Roy; W. W. White**. Taking advantage of Database Extensibility. Gegužė, 2002, [žiūrėta 2007.03.27], prieiga per internetą: <<http://www.phptr.com/articles/article.asp?p=26879&rl=1>>
 - [6] **J. F. Kenney**. The Standard Deviation and Calculation of the Standard Deviation. *Princeton*, 1962, §6.5-6.6, p. 77-80.
 - [7] **J. G. Panagiotis; P. Kappos; C. Makris**. Improving the performance of Web access by bridging global ranking with local page popularity metrics. Balandis, 2002, ISSN: 1066-2243, 12 tomas, Nr. 1, p. 43 – 54. [žiūrėta 2007.03.09], prieiga per internetą: <<http://www.emeraldinsight.com/10.1108/10662240210415817>>
 - [8] **M. E. Glickman**. The Glicko system. Balandis, 2001, [žiūrėta 2007.03.06], prieiga per internetą: <<http://math.bu.edu/people/mg/glicko/glicko.doc/glicko.html>>
 - [9] **M. E. Glickman**. The Glicko 2 system. Gruodis, 2001, [žiūrėta 2007.03.06], prieiga per internetą: <<http://math.bu.edu/people/mg/glicko/glicko2.doc/example.html>>
 - [10] Ranking. Sausis, 2007, *Wikipedia*, [žiūrėta 2007.03.27], prieiga per internetą: <<http://en.wikipedia.org/wiki/Ranking>>
 - [11] Rating. Kovas, 2007, *Wikipedia*, [žiūrėta 2007.03.13], prieiga per internetą: <<http://en.wikipedia.org/wiki/Rating>>
 - [12] **R. C. Jones**. Evaluating Competitive Performance with Rankings and Ratings. Kovas, 2006, [žiūrėta 2007.03.15], prieiga per internetą: <<http://www.ratingtheory.com>>
 - [13] **T. Kardi**. Mean and Average. Briželis, 2005, [žiūrėta 2007.03.04], prieiga per internetą: <<http://people.revoledu.com/kardi/tutorial/BasicMath/Average/>>
 - [14] **Y. Shi; P. Specht; J. Stolen**. A consensus ranking for information system requirements. *Information Management & Computer Security*. Balandis, 1996, 4 tomas, Nr 1, p. 10 – 18, prieiga per internetą: <<http://www.emeraldinsight.com/Insight/viewContentItem.do?contentType=Article&contentId=862639>>
 - [15] **W.D. Cook; L. Seiford**. Priority ranking and consensus formalion. *Management Science*. 1978, 24 tomas, Nr. 16, p. 1721-1732
 - [16] Žinios. Rugsjūtis, 2006, *Wikipedia*, [žiūrėta 2007.03.27], prieiga per internetą: <<http://lt.wikipedia.org/wiki/%C5%BDinios>>
-

8 Priedai

8.1 Priedas Nr.1. Reitingavimo tematika publikuotas straipsnis

INTERNETINĖS ŽINIŲ SVETAINĖS REITINGAVIMO MODELIS

Kęstutis Vanagas

Kauno technologijos universitetas, Informacijos sistemų katedra, Studentų g. 50, Kaunas

Straipsnyje analizuojami egzistuojantys internetinių svetainių reitingavimo metodai. Sudarytas naujas internetinės svetainės reitingavimo modelis lanksčiai skaičiuoja reitinguojamų vienetų, tokių kaip vartotojai, talpinami straipsniai, reitingus, leidžia sistemai dirbti su minimaliu administruojančio asmens įsikišimu bei turi galimybę praplėsti reitinguojamų vienetų aibę. Reitingavimo sistemos modelis [6] taikomas publikuojamų žinių svetainės struktūrai ir savo pagrindinę funkcinę darbo dalį atlieka autonomiškai.

1. Reitingavimo modelių analizė

Nagrinėjant termino žinios prasmę, galima sakyti, jog tai yra organizuotos informacijos visuma bei sąryšiai tarp jos, kurių pagrindu gali veikti koks nors asmuo arba sistema. Žinios reikalauja tvirto pagrindimo ir paaiškinimo. Viena svarbiausių kriterijų – galimybė jas patikrinti bei pakartotinai surasti [10].

Reitingavimas yra individų ar grupių pozicionavimo procesas, nurodantis jų vietą eilėje ir ryšius su kitais objektais [5]. Internetinio portalo reitingavimas naudojamas pozicionuojant jame dalyvaujančius objektus, tokius kaip straipsniai, rašytojai, skaitytojai.

Tiriant **tiesinį** reitingavimą, kiekvienam reitinguojamam vienetui priskiriamas skaitliukas, kuris įtakojamas išorinių veiksnių didinamas atitinkamai pagal atlikto veiksmo stiprumą. Tokia reitingų apskaita neapkrauna procesoriaus bei duomenų bazės darbu.

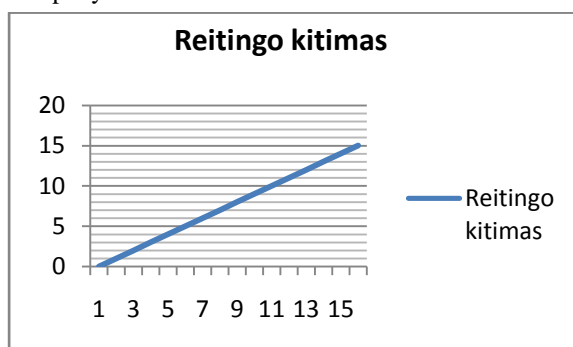
Veiksmų aibė nusakoma tokia formule:

$$R_j = R_j + v_{ij} \quad (1)$$

Čia:

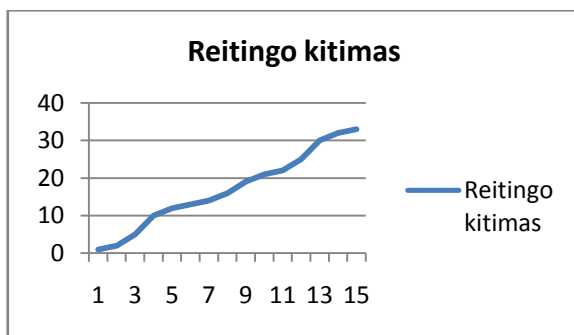
- i Reitinguojamo elemento identifikatorius
- j Reitinguojamo vieneto identifikatorius
- R_j Reitinguojamo j -tojo vieneto esamas skaitinis reitingas
- v_{ij} Gauta reitingo skaitinė vertė j -tajam reitinguojamam vienetui už i -tąjį įvykį

Jei tokios reitingavimo sistemos skaitliuko didinimo reikšmė $v = 1$ (taip būna aktyvumo matavimo reitinguose), tuo atveju visa laiko funkcija tampa tiesine **1 pav.** Iš grafiko matoma, jog už kiekvieną atliktą įvykį reitingo balas pakyla vienetu.



1 pav. Tiesinis reitingo kitimo grafikas

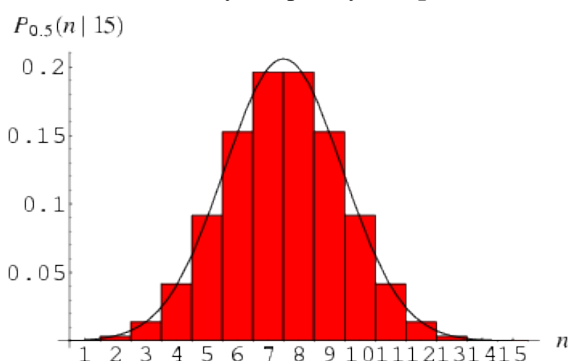
Lankstesnei tiesinei sistemai už įvykius skiriami papildomi natūralieji svoriai $v \in \mathbb{N}$ (pvz.: „už nuotraukos patikimą galima skirti nuo 1 iki 5 taškų“), tada reitinguojamo vieneto reitingo augimo grafikas yra intervalais tiesiškas (žr. **2 pav.**).



2 pav. Tiesinis reitingo kitimo grafikas įvertinus koeficientus

Tiek kiek ši sistema yra paprasta, tiek pat ji yra ir nelanksti. Čia galime įžiūrėti aibę neigiamų savybių. Viena iš tokių yra internetinės svetainės senbuviai – pastarieji jau turi prisirinkę pakankamai didelius kiekius taškų, ir naujai pasirodžiusiam svetainės dalyviui dažniausiai patekti į viršūnes būna ganėtinai sunku arba net visai neįmanoma. Iš to galima daryti išvadas, jog šis reitingavimo būdas labiau tinka vartotojų lojalumui ar senbuviškumui nusakyti, tačiau netinka reitinguojamo vieneto (vartotojo, straipsnio, ...) kokybei nusakyti.

Taškų **vidurkio** reitingavimo modeliu pasiekiamas normaliojo skirstinio arba kitaip vadinamo Gauso skirstinio dėsnis [1]. Pasiskirstymas parodytas 3 pav.



3 pav. Pavyzdinė Gauso pasiskirstymo kreivė.

Šio metodo būdu reitinguojami vienetai dažniausiai vertinami kitų to paties internetinio portalo dalyvių. Reitinguojant atitinkamus vienetus, reitinguotojui būna suteikiama galimybė išsirinkti vieną iš kelių reitingo įverčių. Pradžioje reitinguojamo vieneto galutinė reikšmė labai svyruoja nuo gauto įverčio, nes vidurkis išskaičiuojamas tik iš jau surinktų reikšmių [7]. Vėliau vidurkis nusistoja ties daugmaž bendra visų balsuojančiųjų nuomone. Vidurkio principo reitingavimo privalumas yra tas, jog jau po keleto pirmųjų reitingavimo įvykių, galima nustatyti reitinguojamo vieneto tikrąją vertinę reikšmę, ir reitinguojamo vieneto ilgas buvimas sistemoje nedaro jo monopolistu dėl surinktų taškų skaičiaus. Priešingai – kuo ilgiau reitinguojamas vienetas būna sistemoj, tuo jo vidutinė reikšmė darosi stabilesnė, ir kiekvienas naujas didelis arba mažas balas galutiniam rezultatui ypatingos įtakos nedaro.

Naudojantis **konsensuso(vienodumo)** prioritetų reitingavimo sistema galima patogiai apskaičiuoti reitinguojamo vieneto arba vienetų grupės vertę iš anksto apibrėžtomis kokybės linijoms. Sistema pagrįsta egzistuojančia aibe objektų, kurie bus vertinami pagal reitinguojamus aspektus. Reitingo uždirbtų taškų kiekis tiesiogiai priklauso nuo atliktos užduoties ir gauto rezultato skirtumo tikrojo užduoties rezultato atžvilgiu [9]. Kiekvienos užduoties teisingas rezultatas yra tam tikras sveikasis skaičius, o besivaržančio objekto gautas rezultatas gali pakliūti į trejas sritis:

- Gautas rezultatas yra didesnis nei „tikrasis“;
- Gautas rezultatas yra lygus „tikrajam“;
- Gautas rezultatas yra mažesnis nei „tikrasis“.

Žinoma, geriausias pasiekimas reitinguojamam objektui yra tas, kai jis gauna rezultatą lygų „tikrajam“ užduoties rezultatui. Cook ir Seiford metodas [8] intuityviai demonstruoja galimybę sujungti nukrypimų nuo esminio taško skirtumus tam, kad apibrėžti pastovumo reitingus. Metodas pagrįstas reitinguojamų objektų gautų taškų skirtumo iki realiosios užduoties reikšmės mažinimu.

Probleminėje srityje turime n reitinguojamų objektų – vartotojų ir m užduočių. Tuo atveju r_{ij} bus atskiro vartotojo atskiros užduoties reitingas, kai:

$$i = 1, \dots, n; (2)$$

$$j = 1, \dots, m; (3)$$

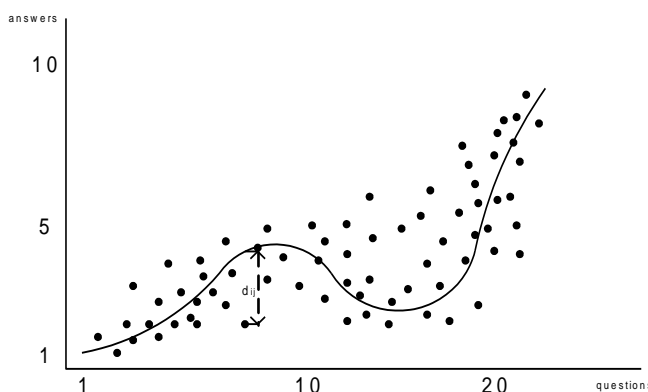
r_j^c yra tikrasis j -tosios užduoties rezultato įvertis. Tada i -tojo vartotojo absoliutus pastovumo reitingas apibrėžiamas formule:

$$d_i = \sum_{j=1}^m |r_{ij} - r_j^c|, i = 1, \dots, n. \quad (4)$$

Tuo atveju, visų vartotojų arba vartotojų grupės, komandos absoliutus pastovumo reitingas apibrėžiamas formule:

$$d = \sum_{i=1}^n d_i = \sum_{i=1}^n \sum_{j=1}^m |r_{ij} - r_j^c| \quad (5)$$

Kiekvienoje užduotyje triumfuoja tie vartotojai, kurių absoliutus nuokrypis nuo realios užduoties reikšmės yra pats mažiausias arba lygus nuliui. Skaičiuojant bendrą visų užduočių įvertinimų skirtumą d_i kiekvienam vartotojui atskirai, taip pat galime nusakyti bendrą viso uždavinyno išsprendimo kokybiškumą. **4 pav.** galime matyti galimą atsakymų išsimėtymą apie optimumo kreivę.

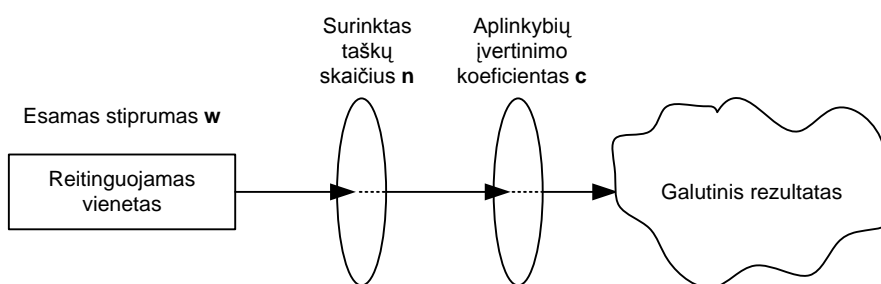


4 pav. Atsakymų išsidėstymas aplink teisingumo kreivę

Šis metodas tinka gauti reitinguojamų objektų – vartotojų reitingus, leidžiančius nusakyti kuris iš besirungusių dalyvių buvo arčiausiai teisingo rezultato. Metodas labiau tinkamas vartotojų skatinimui tobulinti savo žinias ir vis labiau gerinti atsakymų kokybę. Taip pat atsirandanti privalomoji problema yra ta, jog kiekvienam uždaviniui reikia pateikti teisingo įverčio reikšmę. Taip gaunasi, kad vartotojų aktyvumą naudojantis šiuo reitingavimo algoritmu skatina ir kuria internetinio portalo ekspertas, ruošdamas ir publikuodamas galimus reitingų elementus su jų atitinkamais įverčiais. Vartotojo veiksmai tokioje aplinkoje neatlieka kūrybinio-statybinio darbo ir naujos reikalingos žinių portalo turiniui informacijos neįneša.

2. Reitingavimo modelis internetinei žinių svetainei

Tam, kad pasiekti didesnes galimybes reitingų skaičiavimo variacijose, daugumoje reitingavimo įvykių naudojame svorius bei svorio mažinimo koeficientus. Koeficientų dydžiai priklauso nuo įvykio naujumo ar kokybiškumo. Rinkdamas savo galutinį vieno įvykio rezultatą, reitinguojamas vienetas pereina tokius filtrus, kaip pateikta **5 pav.**



5 pav. Reitinguojamo vieneto rezultato skaičiavimo kelias

Laiko klausimui reitinguojamo objekto rezultatui nuspręsti naudojame laiko periodo sąvoką. Skaičiuojant reitinguojamo vieneto visą suminį per istoriją surinktą rezultatą, turi būti sumuojami momentiniai surinkti taškai ir balai įvertinus periodų pasikeitimus. Visi laiko periodai turi savo svertinį koeficientą, kuris einant laiko juosta atgal mažėja $[0; 1]$ ribose. Einamojo laiko periodo svorio koeficientas lygus 1, o prieš tai buvusių svoriai mažėja. Periodų dydžiai bei svorių koeficientai gali būti kintami pagal poreikius. Jei gyvavimo ciklo eigoje keičiasi periodo dydžiai ar svorio koeficientai, atitinkamai keičiasi ir reitinguojamų vienetų galutiniai įverčiai.

Vieno periodo reitinguojamų vienetų įvertis apskaičiuojamas pagal formulę:

$$r_i = \sum_{j=1}^m l_j e_{ij} \quad (6)$$

Čia:

r_i	i-tojo reitinguojamo vieneto surinktų reitingų svoris per periodą
m	Reitingo elementų skaičius
l_j	j-tojo reitingo elemento svoris
e_{ij}	i-tojo reitinguojamo vieneto surinktas taškų skaičius už j-tąjį reitinguojamą elementą

Iš (6) formulės matoma, jog reitinguojamo vieneto reikšmė per periodą gali svyruoti $[0; \infty)$ ribose. Ši atvira reitingo riba netenkina rezultatų apibrėžtumo reikalavimo. Šiam neapibrėžtumui naikinti, l_j reikšmei skaičiuoti įvedame dar vieną santykinį daliklį. Kiekvienam reitinguojamam elementui apibrėžiame maksimalias galimas pasiekiamas jo skaitines ribas. Nustatome, jog vienas autorius gali parašyti maksimumą re^{max} straipsnių. Maksimumo įvertinimas matomas (7) formulėje:

$$l_j = \frac{l_j^*}{re^{max}} \quad (7)$$

Šitaip, kad gauti vienetinį įvertinimą, autorius turėtų parašyti re^{max} straipsnių. Reitingavimo tikslas yra neleisti reitinguojamam vienetai pasiekti vienetinę reitingo ribą. Jei riba pasiekama kokiam nors reitinguojamame elemente, tada nustatyta maksimali riba yra automatiškai padidinama pradiniu nustatytu maksimumu:

$$re_n^{max} = re_{n-1}^{max} + re_1^{max} \quad (8)$$

Čia:

n Maksimumo keitimo eilės numeris

Galutinis reitinguojamo vieneto reitingas apskaičiuojamas įvertinus visų laiko periodų svorių koeficientus:

$$r_n = \frac{\sum_{j=1}^m l_j e_j + \sum_{i=1}^{n-1} k_i r_i}{n} \quad (9)$$

Čia:

n Periodų skaičius

m Reitingo elementų skaičius

l Reitingo elemento svoris

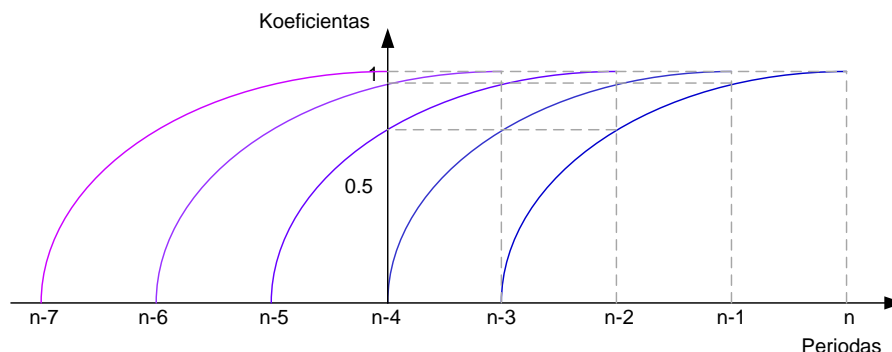
e Reitinguojamo vieneto surinktas taškų skaičius už j-tąjį reitinguojamą elementą

k Periodo svoris

r Periodo metu surinktas reitingas

Algoritmo panaudojimas neleidžia reitinguojamam objektui išsigalėti visoje sistemoje dėl jo senbuviškumo. Visų ankstesniuose perioduose taškų galia santykinai mažinama atitinkamai to periodo koeficiento dydžiui. Periodo pasikeitimo momentu visų ankstesnių periodų koeficientiniai svoriai yra mažinami. Tokiu atveju atsiranda galimybė reitinguotam vienetai iškilti aukščiau kitų tik dėl to, jog kiti vienetai per ankstesnį laiko periodą atliko ar gavo mažiau reitinguojamų veiksmų (nesvarbu, net jei vienetai buvo ankstesniais periodais surinkę daugiau taškų). Šiam požymiui išlaikyti ir norimai sustiprinti, periodų koeficientų parinkimui naudojamas eksponentinio kitimo dėsnis. Dėsnio tęstinumui išlaikyti apibrėžiamas periodų skaičius, kuriam praėjus periodo koeficientas tampa lygus nuliui.

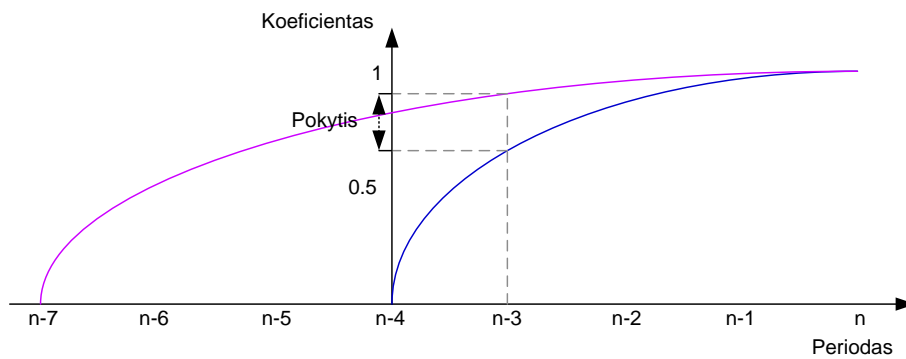
Periodų svorių keitimo dėsningumas pateiktas **6 pav.**



6 pav. Periodų svorių keitimo dėsnis

6 pav. matome, jog kiekvienas tolimesnis periodas turi vis mažesnę koeficientą ir jei apibrėžtas išnulinimo periodas, pavyzdžiui, lygus ketveriems periodų vienetais, tai jiems praėjus visų reitinguojamų vienetai surinkti balai į galutinį reitingą neįsiskaičiuoja. Periodo koeficiento kitimo ribos priklauso nuo to, kokį

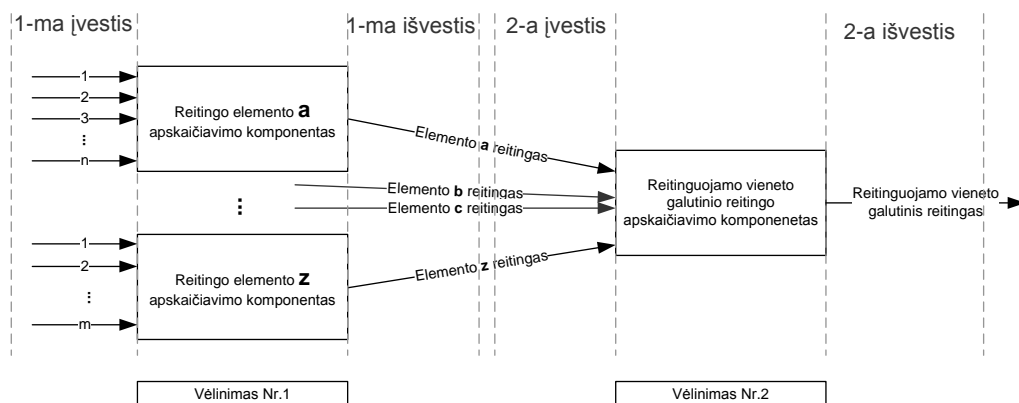
periodų skaičių laikysime neišnulinamu. Periodų koeficientų kitimo dėsnis priklausomai nuo neišnulinamo periodo ilgio pateikiamas **7 pav.**



7 pav. Periodo koeficiento svorio priklausomybė nuo neišnulinamo periodo ilgio

Šis periodų pagal svorių koeficientus naudojimo algoritmas įgyvendina Glicko metodo [3][4] vartotojo patikimumo faktorių – naujesniais praėjusiais periodais nedalyvavę reitinguojami vienetai praranda didžiąją dalį savo surinktų taškų.

Sistemos lengvam išplečiamumui [2] pasiekti reikalinga objektus apibendrinanti reitingavimo struktūra. Reitinguojamo vieneto reitingo apskaičiavimui naudojamas dalinai universalus algoritmas, kuris grafiškai pateiktas **8 pav.** Reitinguojamo vieneto galutinis reitingo apskaičiavimo komponentas yra nekintantis jei keičiasi, atsiranda ar panaikinami patys reitinguojami vienetai bei reitingų elementai. Komponentų išvestys visada yra nekintančios ir išvedamų rezultatų kiekis visada lygus 1. Įvesčių skaičius gali kisti pagal tai, kokius reitinguojamus vienetus bei elementus apskaičiuojame. Reitinguojamo vieneto galutinio reitingo apskaičiavimo komponento įvesčių skaičius kiekvienam reitinguojamam vienetai gali skirtis, tačiau pats apskaičiavimo algoritmas nesikeičia, tiesiog keičiasi kintamojo m reikšmė formulėje $\sum_{j=1}^m l_j e_j$.



8 pav. Reitinguojamo vieneto reitingo apskaičiavimas

Reitinguojamo vieneto reitingo elemento svoris periodui įvertinant reitingo elemento maksimumo reikšmę nusakomas formule:

$$r^{Unit} = \frac{r_{ire}^{Number}}{r_{em}^{MaxValue}} * r_{e}^{Weight} * p^{Weight} \quad (10)$$

Čia:

r^{Unit}	Reitinguojamo vieneto reitingas periodui
r_{ire}^{Number}	Reitinguojamo vieneto reitingo elemento surinktas taškų skaičius
$r_{em}^{MaxValue}$	Reitinguojamo elemento nustatytas maksimalus taškų skaičius
r_{e}^{Weight}	Reitinguojamo elemento svoris
p^{Weight}	Periodo svoris

3. Išvados

Nagrinėjant tiesinių bei vidurkio reitingų skaičiavimo modelius pastebėta, jog modeliai yra paprasti realizuoti, bet nėra lankstūs, jei tenka juos pritaikyti kintant reitinguojamiems vienetais, tačiau vartotojui lengvai perprantami. Geram reitingavimo modeliui reikia paprasto realizavimo neapkraunant sisteminių resursų, tačiau pageidaujama gero interaktyvumo su vartotoju. Siūlomame reitingavimo modelyje vyrauja santykiniai reitingų skaičiavimai, o svertinių koeficientų naudojimas pasižymi realistiškumo bruožais. Modelio skaičiavimo

algoritmuose įtraukiami apibendrinimo parametrai, kurie leidžia praplėsti reitinguojamų vienetų aibę nekeičiant apskaičiavimo proceso.

Literatūros sąrašas

- [1] **E. Weisstein.** Normal Distribution. Birželis, 2004, prieiga per internetą: <<http://mathworld.wolfram.com/NormalDistribution.html>>
- [2] **J. T. Anderson; P. Brown; J. Roy; W. W. White.** Taking advantage of Database Extensibility. Gegužė, 2002, prieiga per internetą: <<http://www.phptr.com/articles/article.asp?p=26879&rl=1>>
- [3] **M. E. Glickman.** The Glicko system. Balandis, 2001, prieiga per internetą: <<http://math.bu.edu/people/mg/glicko/glicko.doc/glicko.html>>
- [4] **M. E. Glickman.** The Glicko 2 system. Gruodis, 2001, prieiga per internetą: <<http://math.bu.edu/people/mg/glicko/glicko2.doc/example.html>>
- [5] Ranking. Sausis, 2007, *Wikipedia*, prieiga per internetą: <<http://en.wikipedia.org/wiki/Ranking>>
- [6] Rating. Kovas, 2007, *Wikipedia*, prieiga per internetą: <<http://en.wikipedia.org/wiki/Rating>>
- [7] **T. Kardi.** Mean and Average. Briželis, 2005, prieiga per internetą: <<http://people.revoledu.com/kardi/tutorial/BasicMath/Average/>>
- [8] **W.D. Cook; L. Seiford.** Priority ranking and consensus formalion. *Management Science*. 1978, 24 tomas, Nr. 16, p. 1721-1732
- [9] **Y. Shi; P. Specht; J. Stolen.** A consensus ranking for information system requirements. *Information Management & Computer Security*. Balandis, 1996, 4 tomas, Nr 1, p. 10 – 18, prieiga per internetą: <<http://www.emeraldinsight.com/Insight/viewContentItem.do?contentType=Article&contentId=862639>>
- [10] Žinios. Rugsėjis, 2006, *Wikipedia*, prieiga per internetą: <<http://lt.wikipedia.org/wiki/%C5%BDzinios>>

INTERNET NEWS PORTAL RANKING MODEL

In this article we analyze existence web portal ranking methods. New web portal ranking model suggested which calculates ratings for ranking units, such as users or placed articles. Also this method works without administrative user intervention but has possibility to extend ranking units set without changing data collection structure in data base. This ranking model applicable for news web portal and mainly all functional work performs by itself.