VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
KOMPIUTERIJOS KATEDRA

Baigiamasis bakalauro darbas

Automatizuotas straipsnių paėmimas iš naujienų portalų

Automated Taking of Articles from News Portals

Atliko: 4 kurso, 1 grupės studentė
Anastasija Novičkova (parašas)

Darbo vadovas:
lekt. Linas Būtėnas

Vilnius
2017

Turinys

Anotacija

Bakalauro baigiamojo darbo tikslas - sukurti įrankį automatizuotam teksto paėmimui iš naujienų portalų. Įrankio sukūrimui buvo išnagrinėtas straipsnis aktualia tema ir išanalizuoti panašaus veikimo įrankiai. Prieš kuriant įrankį buvo detaliai suprojektuota duomenų bazė, suplanuotas programos algoritmas bei nustatyta įrankio kūrimo žingsnių seka. Pagal suprojektuotą duomenų bazę buvo sukurtos atitinkamos lentelės, į jas patalpinti šablonai informacijos paėmimui. Buvo sukurtas įrankis, automatizuotai kas kurį laiką surenkantis nuorodas, straipsnius ir komentarus iš naujienų portalo „Delfi“ ir pateikiantis surinktą informaciją vartotojo sąsajoje. Taipogi buvo sukurta vartotojo sąsaja bei prisijungimo prie jos puslapis. Atlikus darbą buvo pateiktos išvados bei rekomendacijos.

Summary

Bachelor's final work ‘s purpose – create a tool for automatic text grabbing from news websites. Before tool creation were examined articles relevant topic and analyzed similar tools. Before the development of the tool were designed database, planned program ‘s algorithm and determined sequence of the tool creation steps. According to the designed database was created tables in database, and placed templates for information grabbing. Was developed a tool, automatically collecting links, articles and comments from the news website „Delfi“and presenting the collected information in the users accounts. Also, was designed user interface and login page. After all, were presented conclusion and recommendations.

Įvadas

Interneto dėka mūsų laikais įvairios tematikos informacijos kiekis bei prieinamumas yra žymiai didesnis palyginus su praeitu amžiumi. Informacijos analizė yra naudinga tiek versle, tiek mokslinėje veikloje. Sparčiai populiarėjant informacinėms technologijoms tampa aišku, kad tiek informacijos surinkimas, tik tolimesnė analizė daug efektyviau vykdoma pasitelkiant šiuolaikines informacinių technologijų galimybes. Mokslinėje veikloje, analizuojant žmonių nuomonę tam tikra tema ar įvykių, dažnai yra kuriama atitinkama programinė įranga, pagal semantiką analizuojanti pateikta tekstą atitinkama tema bei skaitytojų nuomones šia tema bei teksto turiniu. Kadangi bet kuris informacijos analizės procesas susideda iš kelių dalių (pasiruošimo - nustatoma, kokia informacija bus analizuojama, informacijos surinkimo bei organizavimo būdai, analizės metodai; bei pačio analizės proceso), norint kuo efektyviau atlikti analizę, naudinga automatizuoti kelis arba bent vieną iš aukščiau minėtų etapų. Mokslinėje veikloje, kuomet yra dažnai analizuojama tam tikrą informacija (pvz. straipsniai pagal tematiką bei skaitytojų nuomones iškelta tema), vien informacijos surinkimas ir organizavimas reikalauja nemažai laiko bei resursų.

Pagal anksčiau išvardintą informaciją pasidaro aišku, kad informacinių technologijų dėka galima žymiai paspartinti informacijos analizės procesą mokslinėje veikloje. Galime teigti, kad įstaiga, vykdanti mokslinę veiklą ir turinti reikalingą analizei informaciją, surūšiuotą pagal tematiką ir tinkamai organizuotą duomenų bazę, gali ne tik daug produktyviau vykdyti mokslinę veiklą (šiuo atveju informacijos analizę), bet ir turi patogų įrankį naujų atitinkamos srities specialistų apmokymui. Įrankis, surenkantis kas kurį laiką informaciją iš tam tikrų šaltinių, šią informaciją surūšiuojantis pagal tematiką, saugantis vienoje duomenų bazėje ir pateikianti ją naudotojams patogiu pavidalu, žymiai palengvina teksto analizės procesą mokslinėje veikloje, bei suteikia galimybę efektyviai organizuoti naujų atitinkamus srities specialistų apmokymą.

Bakalauro baigiamojo darbo tikslas - sukurti įrankį automatizuotam straipsnių, komentarų bei informacijos apie juos paėmimui iš naujienų svetainių. Šis įrankis būtų naudingas mokslinei veiklai, norint turėti galimybę parsisiųsti paruoštą tekstą bei komentarus iš sukurtos duomenų bazės tolimesnei teksto analizei. Prie įrankio turėtų prieigą Vilniaus universiteto dėstytojai bei studentai. Kadangi atsirastų galimybė keliems asmenims išanalizuoti vienodą tekstą, išsaugotą vienoje duomenų bazėje, tai leistų objektyviau vertinti sukurtų teksto analizės įrankių našumą bei produktyvumą.

Visų pirma buvo išsamiai išnagrinėti viešai prieinami įrankiai, galintys parsiųsti tik tekstą iš tam tikro internetinio puslapio bei buvo atsižvelgta į kitų panašaus įrankių autorių patirtį. Prieš kuriant įrankį buvo detaliai suprojektuota duomenų bazė, kuri turi pilnai tenkinti vartotojų poreikius. Buvo suplanuoti programos algoritmai bei nustatyta įrankio kūrimo žingsnių seka.

Patogiam ir saugiam programos naudojimui buvo sukurtas prisijungimo prie įrankio valdymo puslapis. Įrankis kuriamas CodeIgniter karkaso pagrindu (veikiančiu MVC principu), PHP programavimo kalba, duomenų saugojimui naudojamos MySQL duomenų bazės. Įrankio naudotojo sąsaja sukurta minimalistinio dizaino CSS karkasų pagalba, o puslapių formatavimas atliekamas HTML kalbos dėka.

1. Informacijos gavybos iš svetainių privalumai ir trūkumai

Informacijos paėmimas iš kitų svetainių gan paplitusi bei populiari praktika. Tai gali būti naudojama tiek naujienų agregavimo svetainėse (pvz. automatiniam informacijos paėmimui iš panašių svetainių ir tolimesnei jos analizei), tiek elektroninėse parduotuvėse (panaudojama rinkos analizei, savo bei konkurentų kainų ir pasiūlos palyginimui), tiek mokslinėje veikloje (siekiant turėti duomenų bazę su informacija tolimesniam jos apdorojimui bei analizei). Vienu iš geriausių bei žinomiausių pavyzdžių galėtų būti RSS naujienos – XML formatu sukurti įrankiai duomenų surinkimui iš naujienų portalų bei žiniatinklių. Tokių įrankiu patogumas yra tame, kad automatiškai galima sekti naujienų portalų atitinkamos rubrikos (arba kelių rubrikų) informaciją. Lietuvoje populiariausi naujienų portalai, tokie kaip „lrytas.lt“, „lrt.lt“, „delfi.lt“, nemokamai tiekia RSS paslaugą, leidžiančią stebėti naujienas iš pasirinkto portalo neužeinant į jį. Taipogi yra svetainių (kaip pvz. „geradiena.lt“, visosnaujienos.lt“), kurios surenka visą informaciją iš populiariausių naujienų portalų, ją sugrupuoja pagal tematiką ir pateikia skaitytojams vienoje vietoje.

Tad, apibendrinant aukščiau minėtą informaciją, galime daryti išvadą, kad informacijos gavimas iš kitų svetainių išties turi nemažai naudos bei privalumų:

- automatinis informacijos paėmimas iš panašių svetainių leidžia patogiai ir paprastai ją išanalizuoti;
- naudinga norint efektyviai bei greitai išanalizuoti rinką, konkurentų pasiūlymus bei palyginti kainas;
- mokslinėje veikloje yra produktyvu turėti vieną duomenų bazę su informacija iš tam tikrų šaltinių, kadangi šią informaciją galima naudoti tolimesnei analizei bei apdorojimui;
- naujienų portalai savo skaitytojų patogumui gali suteikti nemokamas RSS naujienas, kad skaitytojas matytų sugrupuotas naujienas neužeinant į svetainę.

Žinoma, kaip ir kiekvienas geras dalykas, informacijos paėmimo įrankiai turi savo trūkumų:

- reikia atidžiai stebėti, kad nebūtų pažeidžiamos autorinės teisės – t.y. pateikiant surinktą informaciją nurodyti šaltinį bei nuorodą į šaltinį;
- reikia nustatyti optimalų informacijos surinkimo intervalą, kad svetainės, iš kurių surenkama informacija, neblokuotų serverio IP adreso;
- kadangi kiekviena svetainė turi savo struktūrą, pagal kurią talpinama informacija, nėra galimybės sukurti universalaus įrankio, korektiškai surinkančio informaciją iš skirtingų svetainių. Išspęsti šią problemą gali šablonai, pagal kuriuos imama

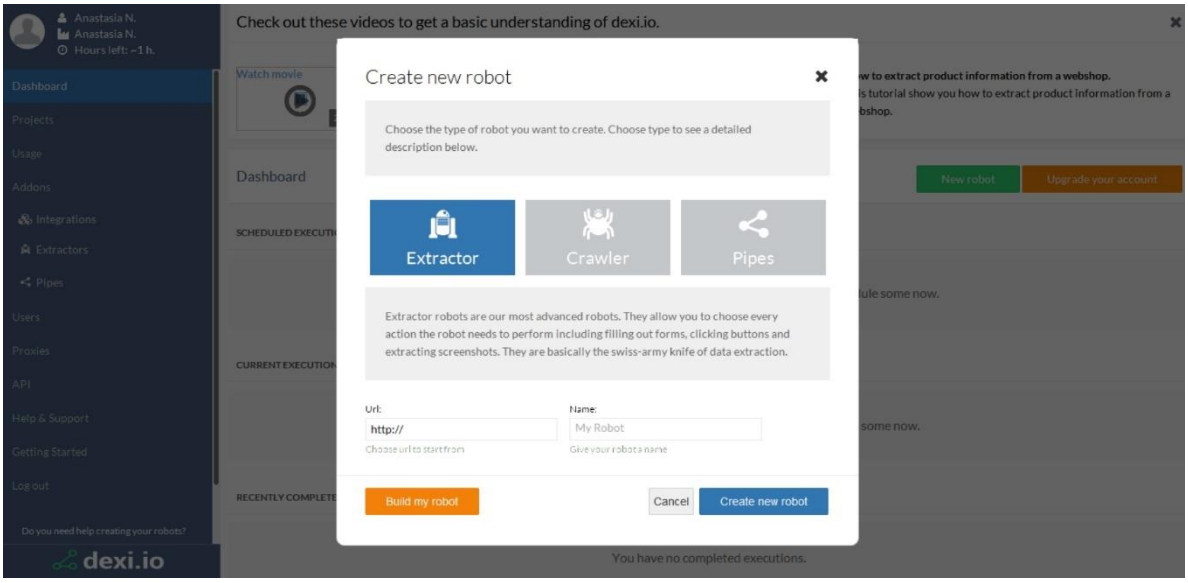
informacija, tačiau juos reikia rankiniu būdu sukurti kiekvienai atskirai svetainei ir patalpinti į duomenų bazę.

Taigi, informacijos gavyba iš svetainių vis dėlto turi daugiau privalumų, negu trūkumų, tuo labiau, kad, kaip buvo minėta aukščiau, kiekvieną trūkumą galima išspręsti, skiriant tam daugiau dėmesio. [TPe10]

2. Panašaus veikimo įrankių analizė

Be abejonės, pasaulyje yra sukurta nemažai įrankių, kurie automatizuotai paima informaciją iš tam tikrų svetainių. Tokie įrankiai būna dviejų tipų – WEB aplikacijos ir programos, parsisiunčiamos į kompiuterį. Kadangi šio bakalauro baigiamojo darbo metu buvo kuriama WEB aplikacija, buvo ieškomos tik panašaus veikimo WEB aplikacijos.

2.1. Įrankis „Dexi.io“ (taip pat žinomas kaip „CloudScrape”)



1 pav. Įrankis „Dexi.io“ – robotuko kūrimas.

Šis įrankis yra gan populiari WEB aplikacija. Kad būtų galimybė naudotis įrankiu, reikia sukurti paskyrą arba prisijungti su esama „Google“ ar „Github“ paskyra. Iš esmės įrankis gali imti informaciją iš beveik visų svetainių, kadangi pats informacijos paėmimas yra paremtas konkrečiai svetainei pritaikyto robotuko kūrimu. Prisiregistravęs vartotojas gali sukurti robotukus, juose nurodant URL adresą svetainės bei žingsnius informacijos paėmimui (pvz. skilties / nuorodos atidarymas, paveikslėlio, teksto ir pan. paėmimas). Robotuko kūrimas nėra itin sudėtingas, bet dirbti su juo reikia išmokti, kadangi kiekvienas žingsnis, nurodomas robotui informacijos paėmimui, yra skirtingo tipo, kuris taip pat turi papildomus nustatymus, tad korektiškam

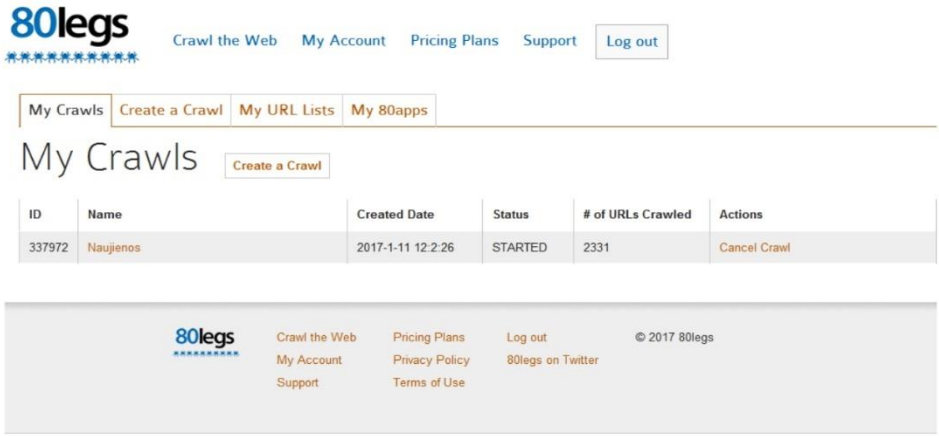
informacijos paėmimui reikia tinkamai sukonfigūruoti robotuką. Dar vienas įrankio privalumas – informacijos paėmimui be robotuko yra sukuriama užduotis, kurioje galima nurodyti, kas kiek laiko informacija turi būti paimama. Šis įrankis galėtų būti nepakeičiamu sprendimu norintiems paimti informaciją iš svetainių, tačiau juo galima naudotis tik užsiregistravusiems vartotojams, taipogi nemokama įrankio versija turi pakankamai mažai funkcionalumo. Yra galimybė užsakyti mokamą versiją, tačiau jos kaina yra $119.00 mėn. vienam vartotojui. [Dex16]

2.2. Įrankis „Import.io“

2 pav. Įrankis „Import.io“ – robotuko kūrimas.

Įrankis „Import.io“ - tikriausiai populiariausiais įrankis informacijos paėmimui. Taip pat kaip ir „dexi.io“ reikalauja sukurti paskyrą arba prisijungti su esama „Facebook“, „Google“, „Github“ arba „LinkedIn“ paskyra. Taipogi yra kuriami robotukai / užduotys informacijos paėmimui. Įrankis suteikia mokamuosius vaizdo įrašus ir kelis pavyzdžius, robotuko kūrimui informacijos paėmimui iš populiariausių svetainių („ebay“, „ikea“). Testuojant įrankį, deja, iš „delfi.lt“ naujienų portalo informacijos paimti ir sukurti robotuko nepavyko. Iš svetainių „lrytas.lt“ bei „15min.lt“ informaciją paimti pavyko, tačiau ne pačiu patogiausiu pavidalu – „lrytas.lt“ atveju buvo surūšiuoti straipsnių pavadinimai (kurie kartu yra ir nuorodos į straipsnius) bei komentarų skaičius (taip pat nuoroda į komentarus) pagal rubrikas, tačiau dėl neaiškių priežasčių buvo paimti ne visų straipsnių komentarai, o tik kai kurių; „15min.lt“ atveju rūšiavimo pagal tematiką nebuvo, į atskirus stulpelius buvo surūšiuoti straipsnių nuotraukos, pavadinimai (kartu ir nuoroda į straipsnį), komentarų skaičius (tuo pat ir nuoroda į komentarus). Įrankis suteikia 30 d. nemokamą laikotarpį išbandymui ir testavimui, vėliau, norint naudoti įrankį, reikia pirkti licenciją, kurios kaina yra 249$ - 799$ mėn. (užklausų kiekis 50 000 – 400 000 mėn.) arba 99$ metams (5000 užklausų per metus).[Imp16]

2.3. Įrankis „80legs.com“



3 pav. Įrankis „80legs.com“ – indeksavimo užduočių sąrašas.

Galima spėti, kad šis įrankis yra netoks populiarus kaip aukščiau minėti „dexi.io“ ir „import.io“, kadangi „Google“ paieškos rezultatuose šis įrankis nėra pateikiamas pirmajame nuorodų 10-ke. Taip pat, skirtingai nuo jau minėtų įrankių, jis neleidžia sukonfigūruoti robotuko tam tikrai svetainei. Norint naudotis įrankiu reikia sukurti paskyrą ir prisijungti (deja, prisijungti su „Facebook“, „Google“ ar kita esama paskyra nėra galimybės). Visų pirma po registracijos galima sukurti indeksavimo („nuskaitymo“) užduotį, kuriai panaudoti jau esamą svetainių sąrašą arba sukurti naują. Kas yra patogu, viename sąraše galima nurodyti kelis adresus, iš kurių turi būti paimta informacija, šį sąrašą išsaugoti, tad informacija bus imama iš skirtingų svetainių vienu metu. Taip pat galima indeksavimo užduoties kūrimo metu nurodyti nuorodų paėmimo „gylį“. Testuojama buvo su sukurtu sąrašu „Naujienos“, į kurį buvo įkeltos nuorodos į populiariausius Lietuvos naujienų portalus – „delfi.lt“, „lrytas.lt“, „15min.lt“, ir buvo pasirinktas gylis 5. Taipogi įrankis leidžia pasirinkti, kokią informaciją reikėtų paimti – testavimui buvo pasirinkta paimti tik nuorodas. Nemokamai besinaudojant įrankiu vienu metu galima paimti iš svetainių iki 10000 puslapių, tačiau galima paliesti tik vieną užduotį vienu metu. Mokamų planų kainos yra 29$ - 299$ mėn. ir leidžia paimti iš svetainių 100 000 – 10 000 000 puslapių vienos užduoties metu bei paleisti vienu metu iki 2 – 5 užduočių. [80l16]

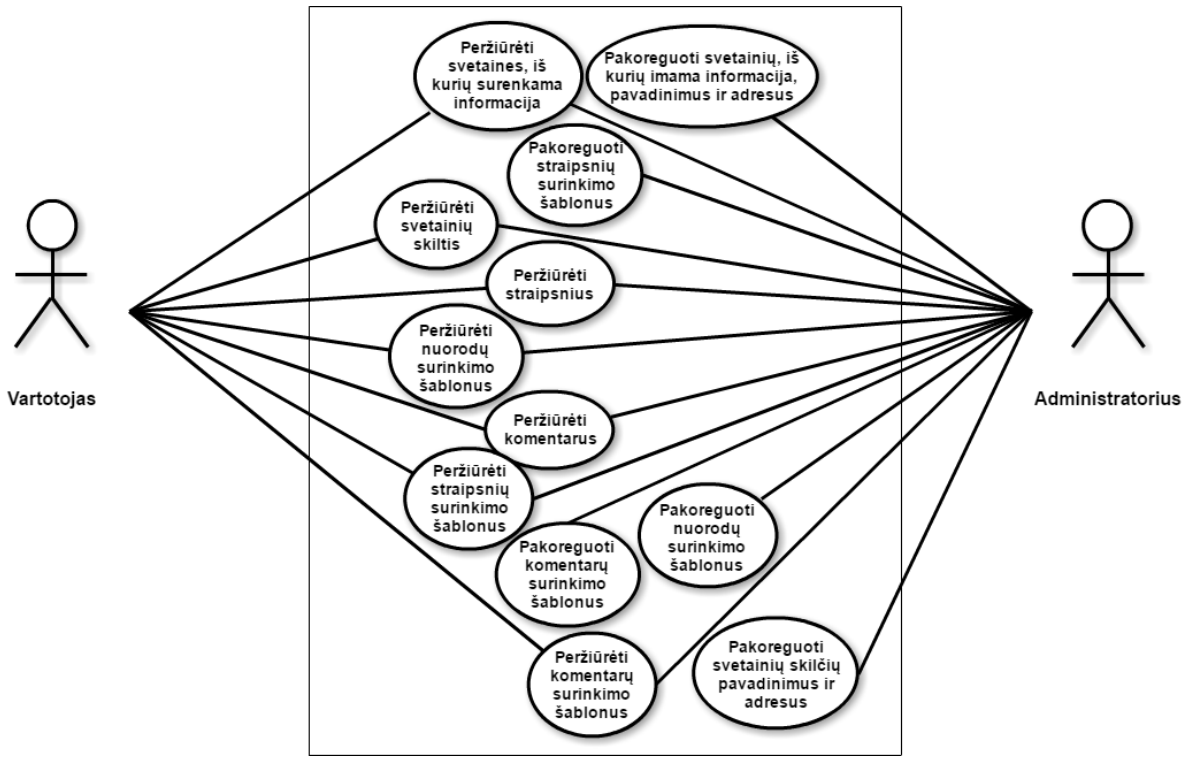
3. Įrankio modelis ir įgyvendinimas

Šioje dalyje aprašomas kuriamas įrankis – jo nauda, suteikiamos galimybės bei funkcionalumas. Yra nurodomi įrankiai, reikalingi įrankio sukūrimui. Taipogi aprašyta suprojektuota duomenų bazė bei joje esančių lentelių paskirtis ir atributai. Šioje dalyje taip pat yra aprašomi algoritmai, pagal kuriuos įgyvendinamas įrankis.

3.1. Įrankio aprašymas

Šio baigiamojo bakalauro darbo metu sukurtas automatizuoto straipsnių paėmimo iš naujienų portalų įrankis yra skirtas, visų pirma, straipsnių paėmimui, jų saugojimui duomenų bazėje bei tolimesnei mokslinei informacijos analizei (pvz. programų, analizuojančių tekstą, kūrimui). Įrankis, kartu su straipsnių tekstais paima informacija apie juos, saugo straipsnius su nuorodomis į juos pagal skiltis (pvz. „Mokslas“, „Verslas“), taipogi surenka bei saugo atitinkamus straipsnių komentarus (pvz. skaitytojų nuomonių analizei). Kadangi kiekviena svetainė, tame tarpe ir naujienų portalai, yra sukurta pagal savo šabloną, tam, kad informacija būtų surenkama korektiškai, duomenų bazėje yra saugomi portalų (darbo metu buvo patalpinti ir ištestuoti tik „Delfi“ portalo šablonai) šablonai, pagal kuriuos ir paimama informacija (pvz. šablonai, nurodantis vietas, kur prasideda ir baigiasi straipsnio tekstas, komentaras ir t.t.).

Prisijungimas prie įrankio vartotojo sąsajos vykdomas su el. paštu bei slaptažodžiu. Yra du naudotojų lygiai – administratorius bei paprastas vartotojas. Administratorius, skirtingai nuo paprastų vartotojų, gali ne tik matyti paimtą iš svetainių informaciją, bet ir koreguoti šablonus informacijos paėmimui, svetainių bei skilčių pavadinimus. Prisijungę vartotojai gali peržiūrėti jau paimtą informaciją iš svetainių, surūšiuoti matomą informaciją pagal svetaines bei tematiką ir matyti, pagal kokius šablonus informacija yra paimama.



4 pav. Panaudos atvejų (angl. Use Case) diagrama.

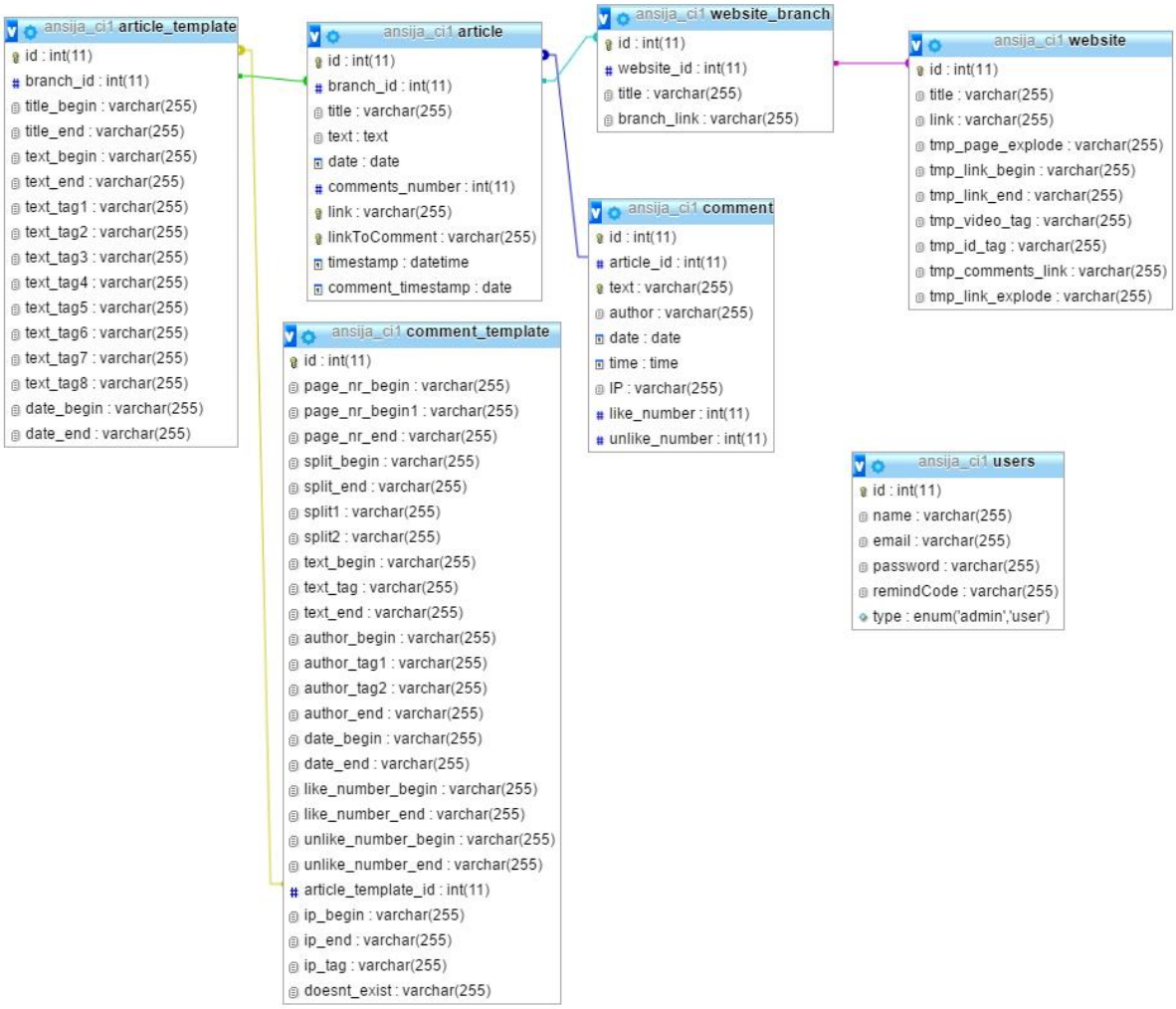
Apibendrinant, įrankis yra WEB aplikacija, prisijungimas prie kurios vartotojo sąsajos vykdomas tam tikrame puslapyje su el. paštu bei slaptažodžiu. Prisijungę vartotojai gali peržiūrėti bei surūšiuoti surinktą iš naujienų portalų informaciją bei pamatyti šablonus, naudojamus informacijos surinkimui; administratoriaus lygio vartotojai be visų išvardintų veiksmų gali ir koreguoti šablonus bei naujienų portalų ar jų skilčių pavadinimus. Informacijos surinkimas vykdomas kas kurį laiką, o surinkta informacija patalpinama į duomenų bazę saugojimui.

3.2. Įrankio sukūrimui naudojamos priemonės

Įrankio sukūrimui naudojama PHP (5.5 versija) programavimo kalba, apipavidalinimui – CSS stilių karkasai ir HTML formatavimo kalba. Vartotojo sąsajai buvo pritaikytas ir papildomai pakoreguotas AdminLTE temos šablonas. Duomenys saugomi duomenų bazėje, kurios valdymas atliekamas su MariaDB 10 versijos (MySQL 5.6 atitikmuo) atviro kodo atviro kodo reliacinių duomenų bazių valdymo sistema. Įrankis sukurtas su CodeIgniter 3.1.0 versijos karkasu, veikiančiu MVC pagrindu.

3.3. Duomenų bazė

Įrankio informacijos paėmimui bei sukauptos informacijos saugojimui yra reikalinga duomenų bazė. Prieš pradedant kurti įrankį buvo detaliai suprojektuota duomenų bazė, nustatyti sąryšiai taip lentelių bei lentelių raktai ir atributai, jų tipai, ilgis bei paskirtis.



5 pav. Duomenų bazės schema.

- Lentelė „users“ skirta įrankio vartotojų prisijungimo duomenims bei statusui (vartotojas ar administratorius) saugoti. Lentelę sudaro keturi atributai – identifikacinis numeris, el. paštas, slaptažodis bei statusas.

Eilutė	Tipas	Dydis	Aprašymas
id	int	11	Vartotojo identifikacinis numeris
name	varchar	255	Vartotojo vardas ir pavardė
email	varchar	255	Vartotojo el. paštas, naudojamas kaip prisijungimo vardas
password	varchar	255	Vartotojo slaptažodis prisijungimui prie įrankio
remindCode	varchar	255	Kodas slaptažodžio priminimui, kas kartą generuojamas naujas, po slaptažodžio keitimo šalinamas
status	enum		Vartotojo statusas („admin“ / „user“)

1 lentelė. Duomenų bazės lentelės „users“ artibutai.

- Lentelė „website“ skirta naujienų portalų pavadinimų bei nuorodų į juos saugojimui. Sudaryta iš dešimties atributų – identifikacinio numerio, pavadinimo, nuorodos, bei šablonų, nuorodų į straipsnius pradžiai ir pabaigai rasti.

Eilutė	Tipas	Dydis	Aprašymas
id	int	11	Naujienų portalo identifikacinis numeris
title	varchar	255	Naujienų portalo pavadinimas (pvz. „DELFI“, „15min“ ir t.t.)
link	varchar	255	Nuoroda į naujienų portalą
tmp_page_explode	varchar	255	Šablonas puslapio dalinimui į elementus, kiekvienas kurių turi nuorodą į straipsnį
tmp_link_begin	varchar	255	Šablonas nuorodos į straipsnį pradžiai rasti
tmp_link_end	varchar	255	Šablonas nuorodos į straipsnį pabaigai rasti
tmp_video_tag	varchar	255	Šablonas patikrinimui, ar nuoroda nėra į vaizdinę medžiagą, ne straipsnį
tmp_id_tag	varchar	255	Šablonas patikrinimui, ar nuorodoje yra dalis šablono, pagal kurį sudaroma nuoroda į komentarus
tmp_comments_link	varchar	255	Šablonas, pridėdamas prie nuorodos į straipsnį tam, kad būtų gauta nuoroda į komentarus
tmp_link_explode	varchar	255	Šablonas, atskiriantis, kur baigiasi nuoroda į straipsnį ir prasideda komentaro nuorodos šablonas

2 lentelė. Duomenų bazės lentelės „website“ artibutai.

- Lentelė „website_branch“ skirta tam tikrų naujienų portalų dalių (pvz. „Mokslas“, „Verslas“) informacijos saugojimui. Lentelė sudaryta iš keturių atributų – identifikacinio numerio, rakto į lentelę „website“, skilties pavadinimo ir nuorodos į ją.

Eilutė	Tipas	Dydis	Aprašymas
id	int	11	Naujienų portalo skilties identifikacinis numeris
website_id	int	11	Naujienų portalo identifikacinis numeris, raktas į lentelę „website“
title	text	65535	Naujienų portalo skilties pavadinimas (pvz. „Mokslai“, „Verslas“, „Sportas“ ir t.t.)
branch_link	varchar	255	Nuoroda į naujienų portalo skiltį

3 lentelė. Duomenų bazės lentelės „website_branch“ artibutai.

- Lentelė „article“ skirta straipsnių informacijos saugojimui. Sudaryta iš dešimties atributų – identifikacinio numerio, skilties identifikacinio numerio, pavadinimo, teksto, datos, komentarų skaičiaus, nuorodos į straipsnį, nuorodos į komentarų puslapį ir straipsnio informacijos bei komentarų surinkimo laikų.

Eilutė	Tipas	Dydis	Aprašymas
id	int	11	Straipsnio identifikacinis numeris
branch_id	int	11	Raktas į lentelę „website_branch“
title	varchar	255	Straipsnio pavadinimas
text	text	65535	Straipsnio tekstas
date	date		Straipsnio atnaujinimo data
comments_number	int	11	Komentarų skaičius
link	varchar	255	Nuoroda į straipsnį
linkToComment	varchar	255	Nuoroda į komentarus
timestamp	datetime		Straipsnio paėmimo data ir laikas
comments_timestamp	date		Straipsnio komentarų paėmimo data

4 lentelė. Duomenų bazės lentelės „article“ artibutai.

- Lentelė „article_template“ skirta šablonų straipsnių informacijos paėmimui saugojimui. Lentelę sudaro šešiolika atributų – šablono identifikacinis numeris, šablonai rasti pavadinimo, teksto, datos, komentarų skaičiaus pradžiai ir pabaigai rasti, šablonai ne teksto dalių (paveikslėlių, nuorodų bei skriptų) pašalinimui iš tekto, šablonas komentarų nuorodai sudaryti bei rakto į lentelę „website_branch“.

Eilutė	Tipas	Dydis	Aprašymas
id	int	11	Šablonų identifikacinis numeris
branch_id	int	11	Raktas į lentelę „website_branch“
title_begin	varchar	255	Šablonas straipsnio pavadinimo pradžiai rasti
title_end	varchar	255	Šablonas straipsnio pavadinimo pabaigai rasti
text_begin	varchar	255	Šablonas straipsnio teksto pradžiai rasti
text_end	varchar	255	Šablonas straipsnio teksto pabaigai rasti
text_tag1	varchar	255	Šablonas ne teksto dalių (paveikslėlių, nuorodų, skriptų) pašalinimui iš teksto
text_tag2	varchar	255	Šablonas ne teksto dalių (paveikslėlių, nuorodų, skriptų) pašalinimui iš teksto
text_tag3	varchar	255	Šablonas ne teksto dalių (paveikslėlių, nuorodų, skriptų) pašalinimui iš teksto
text_tag4	varchar	255	Šablonas ne teksto dalių (paveikslėlių, nuorodų, skriptų) pašalinimui iš teksto
text_tag5	varchar	255	Šablonas ne teksto dalių (paveikslėlių, nuorodų, skriptų) pašalinimui iš teksto
text_tag6	varchar	255	Šablonas ne teksto dalių (paveikslėlių, nuorodų, skriptų) pašalinimui iš teksto
text_tag7	varchar	255	Šablonas ne teksto dalių (paveikslėlių, nuorodų, skriptų) pašalinimui iš teksto
text_tag8	varchar	255	Šablonas ne teksto dalių (paveikslėlių, nuorodų, skriptų) pašalinimui iš teksto
date_begin	varchar	255	Šablonas straipsnio publikavimo datos pradžiai rasti
date_end	varchar	255	Šablonas straipsnio publikavimo datos pabaigai rasti

5 lentelė. Duomenų bazės lentelės “article-template” artibutai.

- Lentelė „comment“ skirta straipsnių komentarų ir informacijos apie juos saugojimui. Sudaryta iš devynių atributų: identifikacinio numerio, straipsnio identifikacinis numerio, teksto, autoriaus, datos, laiko, IP adreso, "patinka" bei "nepatinka" paspaudimų skaičių .

Eilutė	Tipas	Dydis	Aprašymas
id	int	11	Komentaro identifikacinis numeris
article_id	int	11	Raktas į lentelę „article“
author	varchar	255	Komentaro autorius
text	varchar	255	Komentaro tekstas
IP	varchar	255	Komentaro autoriaus IP adresas
date	date		Komentaro parašymo data
time	time		Komentaro parašymo laikas
like_number	int	11	„Patinka“ paspaudimų skaičius
unlike_number	int	11	„Nepatinka“ paspaudimų skaičius

6 lentelė. Duomenų bazės lentelės „comment“ artibutai.

- Lentelė „comment_template“ skirta šablonų komentarų bei jų informacijos paėmimui saugojimui. Lentelė sudaryta iš keturiolikos atributų – identifikacinio numerio, rakto į lentelę „article_template“, šablonų komentarų autorių, IP adreso, datos, laiko, teksto, „patinka“ ir „nepatinka“ paspaudimų kiekio pradžiai ir pabaigai rasti.

Eilutė	Tipas	Dydis	Aprašymas
id	int	11	Komentaro identifikacinis numeris
article_template_id	int	11	Raktas į lentelę „article“
page_nr_begin	varchar	255	Šablonas komentarų puslapių skaičiaus pradžiai rasti
page_nr_begin1	varchar	255	Alternatyvus šablonas komentarų puslapių skaičiaus pradžiai rasti, jeigu nerastas **page_nr_begin** šablonas
page_nr_end	varchar	255	Šablonas komentarų puslapių skaičiaus pabaigai rasti
author_begin	varchar	255	Šablonas komentaro autorius pradžiai rasti
author_tag1	varchar	255	Šablonas ne teksto elementų pašalinimui iš iškirptos eilutės „autorius“

author_tag2	varchar	255	Šablonas ne teksto elementų pašalinimui iš iškirptos eilutės „autorius“
author_end	varchar	255	Šablonas komentaro autorius pabaigai rasti
text_begin	varchar	255	Šablonas komentaro teksto pradžiai rasti
text_tag	varchar	255	Šablonas ne teksto elementų pašalinimui iš komentaro teksto
text_end	varchar	255	Šablonas komentaro teksto pabaigai rasti
ip_begin	varchar	255	Šablonas komentaro autoriaus IP adreso pradžiai rasti
ip_tag	varchar	255	Šablonas komentaro autoriaus IP adreso pabaigai nustatyti, jeigu toks šablonas yra aptinkamas iškirptoje eilutėje
ip_end	varchar	255	Šablonas komentaro autoriaus IP adreso pabaigai rasti
date_begin	varchar	255	Šablonas komentaro parašymo datos bei laiko pradžiai rasti
date_end	varchar	255	Šablonas komentaro parašymo datos bei laiko pabaigai rasti
like_number_begin	varchar	255	Šablonas „Patinka“ paspaudimų skaičiaus pradžiai rasti
like_number_end	varchar	255	Šablonas „Patinka“ paspaudimų skaičiaus pabaigai rasti
unlike_number_begin	varchar	255	Šablonas „Nepatinka“ paspaudimų skaičiaus pradžiai rasti
unlike_number_end	varchar	255	Šablonas „Nepatinka“ paspaudimų skaičiaus pabaigai rasti
doesnt_exist	varchar	255	Šablonas patikrinimui, ar komentaras nebuvo pašalintas arba jau nebeaktualus

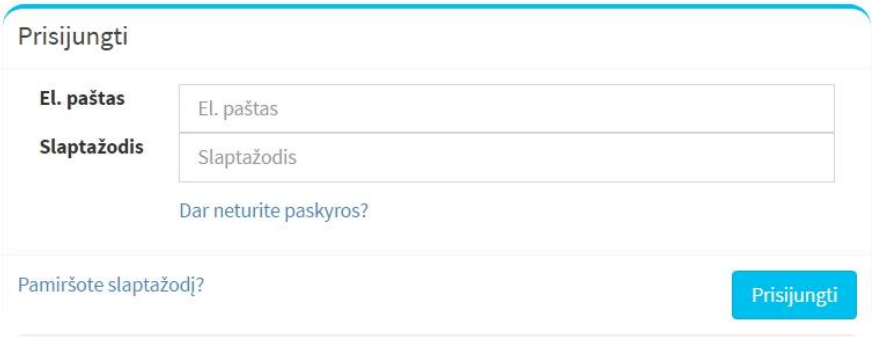
7 lentelė. Duomenų bazės lentelės „comment_template“ atributai.

3.4. Įrankio programinio kodo aprašymas

Įrankis yra sudarytas iš dviejų esminių dalių – matomos vartotojui (vartotojo sąsajos, kurioje atvaizduojama jau surinkta informacija, ir vartotojui nematomos, kurioje yra vykdomas informacijos surinkimas kas kurį laiką pagal periodines užduotis (angl. cronjob).

Pats pirmas puslapis, matomas įrankio naudotojui - puslapis prisijungimui prie įrankio valdymo skydelio. Naudotojų autorizacija vykdoma pagal e. paštą bei slaptažodį. Taipogi naudotojas gali sukurti paskyrą užėjus į registracijos puslapį arba pasinaudoti slaptažodžio priminimu.

3.4.1. Vartotojo sąsajos aprašymas

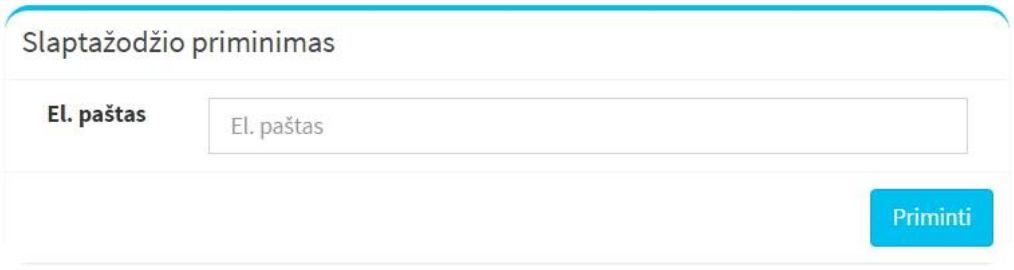


6 pav. Pradinis įrankio puslapis

Pradinio puslapio atvaizdavimas aprašomas faile „/application/views/login.php”, prisijungimo puslapio šablonas yra paimtas iš “Admin LTE” šablono. Vartotojui įvedus prisijungimo duomenis kontroleryje „Login“ vykdomas tikrinimas, ar tokių duomenų kombinacija (el. paštas ir slaptažodis) egzistuoja duomenų bazės lentelėje „users“ (užklausa į duomenų bazę siunčiama modelio „LoginModel“). Jeigu informacija nėra randama, naudotojas grąžinamas į prisijungimo puslapį ir jam pateikiamas klaidos pranešimas dėl nekorektiškų duomenų. Jeigu prisijungimo duomenys yra korektiški, į sesiją įrašomi vartojo el. paštas bei „1“ (reikšmė „logged_in“) , kuris naudojamas apsaugai nuo neautorizuoto prisijungimo prie vartotojo paskyros. Tad, vartotojui įvedus korektiškus prisijungimo duomenis, jis yra peradresuojamas į vartotojo paskyros sąsają, kurioje taipogi vykdomas patikrinimas pagal duomenis, esančius sesijoje (t.y. jeigu sesijoje išsaugota „logged_in“ reikšmė lygi 1, vartotojas mato vartotojų sąsają, kitu atveju vartotojas peradresuojamas į prisijungimo puslapį).

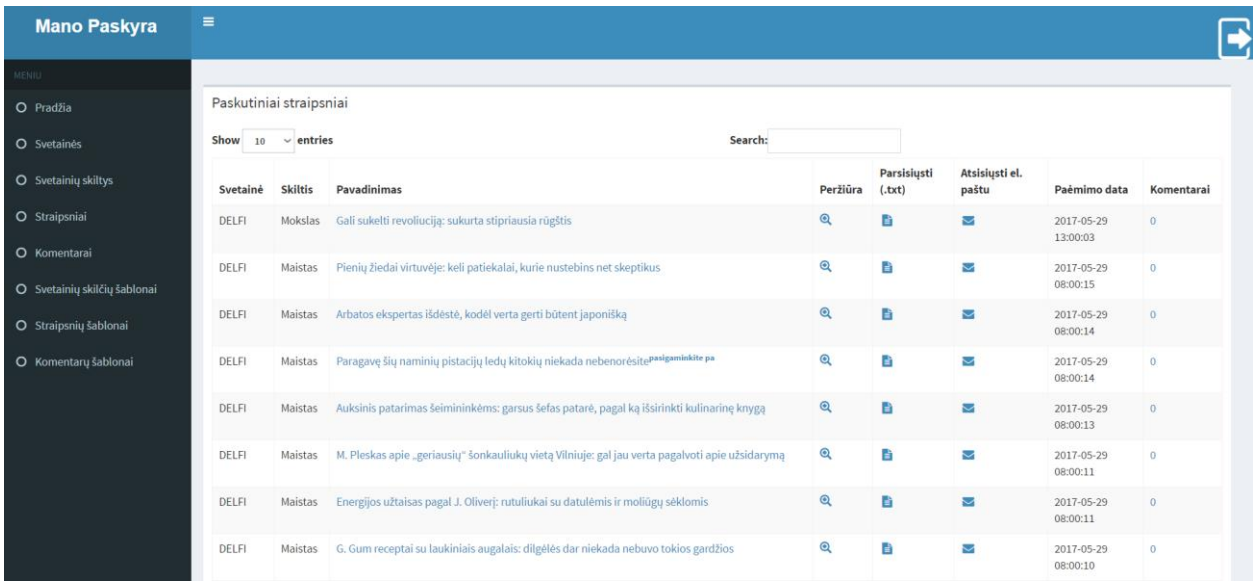
7 pav. Registracijos puslapis

Vartotojams yra suteikiama galimybė sukurti paskyrą, tačiau visų naujų užsiregistravusių vartotojų tipas yra „user“, tad tokie vartotojai gali tik peržiūrėti, bet nekoreguoti informaciją. Puslapio atvaizdavimas yra vykdomas „register“ vaizdo dėka, duomenis yra apdorojami kontroleryje „Register“, duomenis patalpina į duomenų bazę modelyje „LoginModel“ esanti funkcija „create“. Registracijai reikia nurodyti 3 parametrus – vardą ir pavardę, el. paštą bei slaptažodį. Vartotojui įvedus reikalingas reikšmes tikrinama, ar duomenų bazėje jau nėra naudotojo su tokiu el. paštu (siunčiama užklausa iš modelio „LoginModel“ funkcijos „checkUser visų įrašų su nurodytu el. paštu paėmimui). Jeigu iš modelio funkcijos grąžinama „FALSE“ reikšmė (jau egzistuoja toks vartotojas), vartotojas peradresuojamas į prisijungimo puslapį ir jam atvaizduojamas atitinkamas klaidos pranešimas. Kitu atveju paskyra yra sėkmingai sukuriama, apie ką taipogi pranešama vartotojui jį peradresavus į prisijungimo puslapį.



8 pav. Slaptažodžio priminimo puslapis.

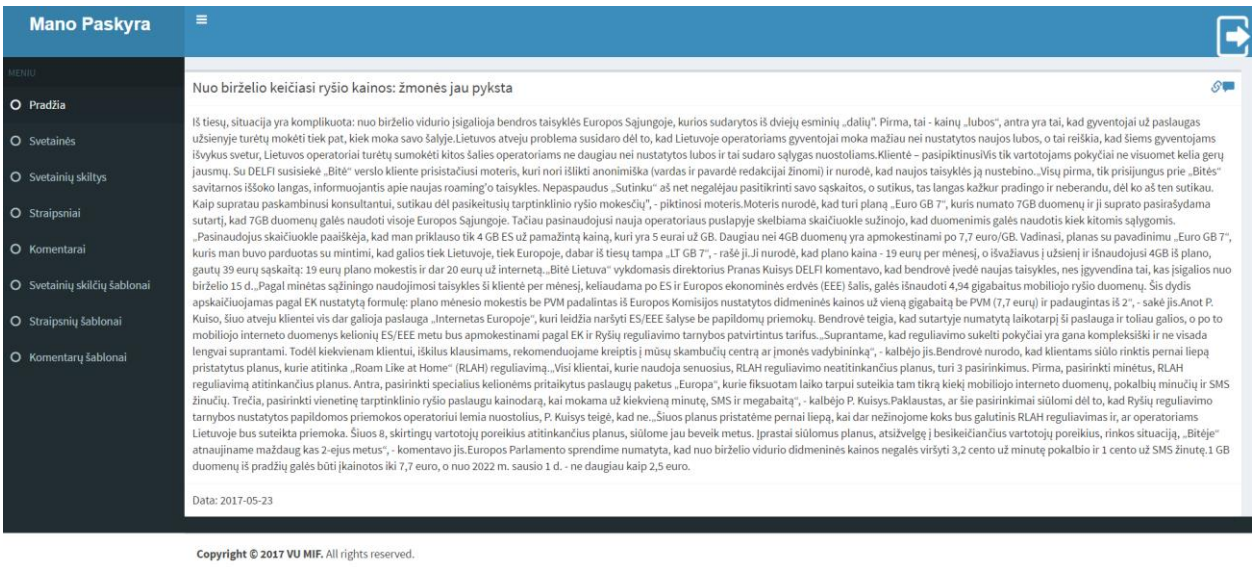
Esant poreikiui, vartotojas gali pasinaudoti slaptažodžio priminimu. Tam reikia slaptažodžio priminimo puslapyje (kontreleris “Remind”, vaizdas “remind”) nurodyti paskyros kontaktinį el. paštą. Vartotojui inicijavus priminimą yra sugeneruojama atsitiktinių 20 simbolių eilutė, kuri patalpinama į duomenų bazę aktualiam varotojui bei naudojama slaptažodžio keitimo funkcijoje (“Remind/password”) kaip “id”. Šį nuoroda išsiunčiama vartotojui el. paštu ir paspaudus ją vartotojas galės įvesti naują slaptažodį. Pakeitus slaptažodį priminimo kodas yra pašalinamas iš duomenų bazės.



9 pav. Pagrindinis paskyros puslapis

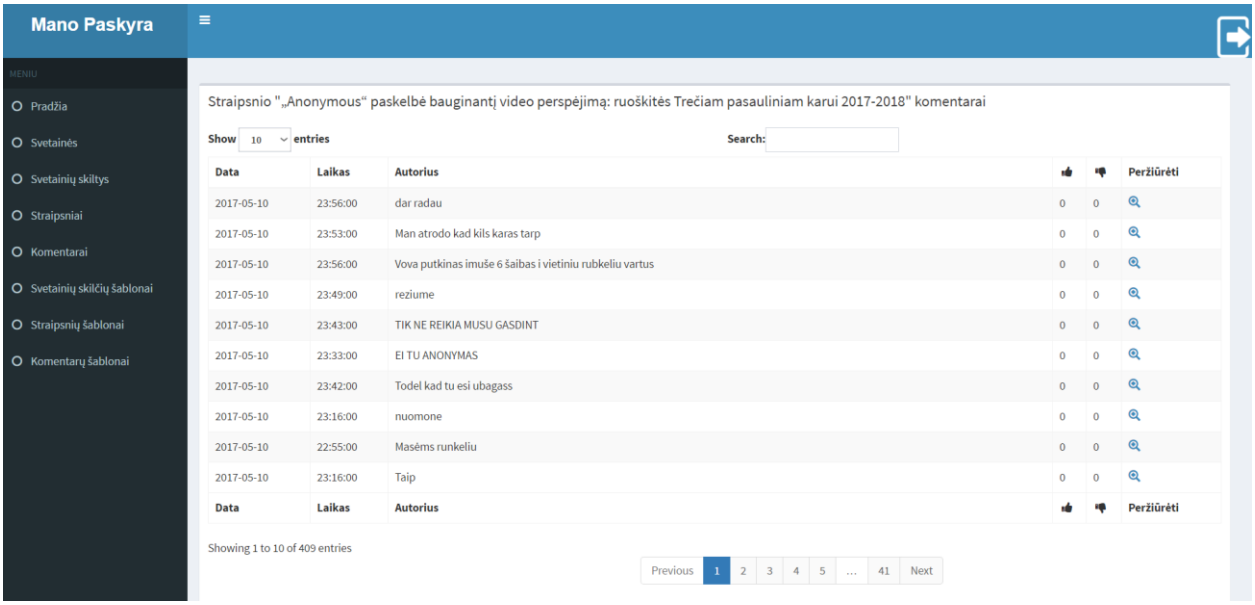
Prisijungęs vartotojas mato sąrašą iš paskutinių 20 surinktų straipsnių. Taip pat vartotojas gali pasirinkti, kokia informacija jis nori peržiūrėti:

- duomenų bazėje išsaugotus naujienų portalus (pvz. „Delfi“) – pasirinkę šią skiltį vartotojas matys svetaines, paspaudęs ant kurių matytų tik iš atitinkamos svetainės surinktą informaciją;
- duomenų bazėje išsaugotas naujienų portalų skiltis (pvz. „Mokslai“, „Verslas“ ir pan.) – paspaudęs ant tam tikros skilties vartotojas matytų tik atitinkamo naujienų portalo skiltyje surinktą informaciją;



10 pav. Straipsnio atvaizdavimas

- duomenų bazėje išsaugotus straipsnius – pasirinkęs šią skiltį vartotojas matytų visų surinktų straipsnių pavadinimus, straipsnio šaltinį (svetainę ir skiltį), komentarų skaičių bei paėmimo datą. Taip pat vartotojas gali parsisiųsti straipsnį „.txt“ formatu arba jį atsisiųsti į savo el. paštą (naudojamą prisijungimui). Paspaudus ant pavadinimo vartotojas matytų iš atitinkamo straipsnio surinktą informaciją (pavadinimą, datą, tekstą, šaltinį bei komentarų peržiūros mygtuką). Paspaudus ant komentarų skaičiaus vartotojas matytų atitinkamo straipsnio komentarus (autorių, datą, laiką, IP adresą, „patinka“ ir „nepatinka“ skaičius bei straipsnį, kuris buvo pakomentuotas). Paspaudus peržiūros ikonėlę ties komentaru vartotojas matytų detalesnę komentaro informaciją – t.y. aukščiau minėtą bei tekstą;



11 pav. Straipsnio komentarų atvazidavimas

- duomenų bazėje išsaugotus komentarus – pasirinkęs šią skiltį vartotojas matytų visų surinktų komentarų datą, laiką, autorių, paspaudimų „Patinka“ ir „Nepatinka“ skaičių bei kokiam straipsniui buvo paliktas aktualus komentaras. Paspaudus peržiūros ikonėlę ties komentaru vartotojas matytų detalesnę komentaro informaciją – t.y. aukščiau minėtą bei tekstą;
- duomenų bazėje išsaugotus skilčių šablonus – vartotojas galėtų pamatyti, pagal kokius šablonus buvo surinktos nuorodos į straipsnius iš pvz. „Delfi.lt“ svetainės;
- duomenų bazėje išsaugotus straipsnių šablonus – šioje skiltyje vartotojas matytų, pagal kokius šablonus yra surenkami straipsniai ir jų informacija;
- duomenų bazėje išsaugotus komentarų šablonus – šioje skiltyje yra matoma, pagal kokius šablonus buvo surenkami straipsnių komentarai ir jų informacija.

Vartotojai yra dviejų tipų – paprasti naudotojai bei administratoriai. Paprasti vartotojai prisijungę prie vartotojo sąsajos gali tik matyti surinktą informaciją bei šablonus. Administratoriai taipogi gali matyti tą pačią informaciją kaip ir paprasti vartotojai, tačiau papildomai gali keisti šablonus duomenų bazėje iš vartotojo sąsajos.

Atsijungimas nuo vartotojo sąsajos vykdomas paspaudus mygtuką „Atsijungti“, esantį sąsajos dešiniame viršutiniame kampe. Vartotojus paspaudus mygtuką „Atsijungti“ yra šalinama informacija iš sesijos ir vartotojas peradresuojamas į prisijungimo puslapį.

3.4.2. Informacijos surinkimo algoritmo aprašymas

Esminė įrankio dalis yra minėta informacijos surinkimui skirta dalis, nematoma vartotojui. Kadangi kiekvienas naujienų portalas turi savo sandarą, informacija yra paimama pagal paruoštus ir patalpintus duomenų bazėje šablonus (t.y. pagal šiuos šablonus ieškoma kiekvieno elemento pradžia ir pabaiga). Informacijos surinkimas vykdomas keliais etapais kas kurį laiką pagal periodines užduotis:

- iš duomenų bazės yra paimama nuoroda į naujienų portalo skiltį bei šablonai informacijos iš skilties paėmimui;
- įrankis, surinkęs visą informaciją iš pateikto adreso, skaido ją pagal nuorodų paėmimo šabloną (ieškoma nuorodos pradžia) ir šias dalis įkelia į masyvą. Kiekvienas masyvo elementas cikle „foreach“ yra toliau skaidomas pagal šablonus (nuorodos pradžios ir pabaigos paieškai), kol nėra surinktos tik nuorodos be papildomų elementų. Šios surinktos nuorodos cikle „while“ (kuriame tikrinama, ar nėra pasiektas masyvo, saugančio nuorodas, tuščias elementas – t.y. pabaiga) yra apdorojamas, pridedant prie jų šabloną, reikalingu sudaryti nuorodas į komentarus

(paprastai naujienų svetainėse prie straipsnio nuorodos yra pridėdami papildomi elementai / kintamieji, taip sudarant nuorodą į komentarų puslapį). Šios nuorodos į komentarų puslapius yra patalpinamos į atskirą masyvą, o vėliau, prieš keliant duomenis į duomenų bazę, abudu vienmačiai masyvai yra apjungiami į vieną dvimatį masyvą. Šio dvimačio masyvo kiekvienas elementas yra masyvas su indeksais „straipsnis“ ir „komentarai“. Sudarius šį masyvą, jo duomenys cikle „foreach“ yra perduodami į duomenų bazės lentelę „article“;

- įrankis paima iš duomenų bazės 20 nuorodų į straipsnius (jie sudedami į masyvą), kurių „timestamp“ atributas lygus „NULL“, bei šablonus straipsnio informacijos paėmimui. Kiekviena nuoroda yra paeiliui atidaroma, iš jos paimama straipsnio informacija – pagal šablonus yra ieškomas kiekvienas reikalingas elementas (pavadinimas, data, tekstas). Kiekvieno straipsnio informaciją yra sudedama į masyvą su indeksais „pavadinimas“, „data“ , „tekstas“ ir „nuoroda“. Kiekvienas masyvas yra perduodamas į modelį „ArticleModel“, iš kurio duomenys patalpinami į duomenų bazės lentelę „article“;
- iš duomenų bazės lentelės „articles“ yra paimamos nuorodos į komentarų puslapius, įdedamos į masyvą ir kiekviena iš eilės atidaroma cikle „foreach“. Atidarius kiekvieną nuorodą surenkamas visas puslapio turinys ir visų pirma pagal duomenų bazėje esančius šablonus ieškomas komentarų puslapių skaičius. Cikle „for“ pagal šabloną yra sudaroma nuoroda į kiekvieną komentarų puslapį. Kiekvienos nuorodos turinys analogiškai, kaip buvo imamos nuorodos į svetaines, yra skaidomas į dalis pagal kiekvieno komentaro pradžią ir patalpinamas į masyvą. Sukurtas masyvas yra toliau apdorojamas kitų funkcijų, kuriose pagal šablonus apkerpamas kiekvienas reikalingas elementas ir patalpinamas į atitinkamą masyvo poziciją (t.y. autoriai patalpinami į masyvo indeksą „autorius“, IP adresai į „IP“ indeksą ir t.t.). Kiekvieno komentaro masyvas perduodamas į modelį „CommentModel“, iš kurio duomenys yra patalpinami į duomenų bazes lentelę „comment“.

Išvados ir rekomendacijos

Baigiamojo bakalauro darbo metu buvo išnagrinėta literatūra darbo tema bei jau sukurtos panašaus veikimo WEB aplikacijos. Atliekant tai, buvo padaryta išvada, kad informacijos paėmimas išties yra naudingas ir pakankamai paplitęs tiek versle, tiek mokslinėje veikloje. Taipogi buvo pastebėta, kad:

1. korektiškam informacijos surinkimui yra būtini iš anksto paruošti šablonai kiekvieno elemento pradžios ir pabaigos paieškai. Kadangi kiekviena svetainė talpina turinį pagal skirtingas temas (kartais ir unikalias), deja, nėra įmanoma rasti universalaus sprendimo, kaip rinkti informaciją iš skirtingų svetainių, tam nenaudojant iš anksto paruoštų šablonų. Tai yra aktualu tik tuo atveju, jeigu reikalinga rinkti tik korektiškai surūšiuotą informaciją, be jokių papildomų elementų ir tik reikalinga informacija (pvz. yra įrankių, kurie gali surinkti tik nuorodas iš svetainės, tačiau yra surenkamos visos nuorodos, tad šiuo atveju toks įgyvendinimas nėra tinkamas);
2. dažnai naujienų portalai turi sudėtingesnę sandarą negu įprastos svetainės, tad kai kuriems iš jų reikia ne tik parinkti atskirus šablonus, bet ir suprogramuoti atskiras funkcijas. Taip yra reikalinga būtent dėl korektiško informacijos surinkimo, be papildomų elementų (banerių, nuorodų tekste ir t.t.);
3. net kuriant įrankį tik vienai svetainei reikia kruopščiai išstudijuoti skirtingus informacijos pateikimo atvejus (šiuo atveju straipsnius), kadangi net ir vienoje svetainėje informacijos pateikimas gali žymiai skirtis (pvz. priklausomai nuo straipsnio pastraipų, komentarų puslapių skaičiaus).

Sukurtas įrankis gali būti įdiegtas visuose serveriuose, kuriuose yra PHP 5.5 versijos(su kitomis PHP versijomis testuota nebuvo), ir MariaDB 10 versijos (arba MySQL 5.6 versijos) palaikymas. Diegimui reikalinga įkelti svetainės failus į „public_html“ katalogą bei importuoti duomenų bazę („.sql“ failą). Papildomai reikia patikrinti ir , jeigu reikia, pakeisti duomenų bazės pavadinimą bei prisijungimo duomenis konfigūraciniame faile „application/config/database.php“, taipogi pakeisti svetainės adresą („base_url“) faile „application/config/config.php“.
Tolimesniam įrankio tobulinimui siūlyčiau bei rekomenduočiau:

1. pritaikyti įrankį informacijos surinkimui ir iš kitų naujienų portalų (pvz. „lrt.lt“, „15min.lt“, „lrytas.lt“). Tam, visų pirma reikės įkelti atitinkamus šablonus į duomenų bazę ir patestuoti įrankio veikimą. Žinoma, kaip buvo paminėta anksčiau, kartais skirtingų svetainių sandara žymiai skiriasi, tad gali prireikti sukurti papildomas funkcijas atskiroms svetainėms – šios funkcijos gali būti nesudėtingai sukurtos

remiantis jau esančiomis, tik pakoreguojant, kokį kiekį nereikalingos informacijos reikia išimti iš elemento (pvz. teksto). Ir toliau tobulinant įrankį, jis galėtų tapti puikia mokymosi priemonė bei pagalbininku mokslinėje veikloje.;

2. pritaikyti įrankį Vilniaus Universiteto naudotojams – sukurti registravimo sistemą tik su Vilniaus Universiteto el. paštais (t.y. kad sistema tikrintų registruojama el. paštą pagal galūnę). Šiuo metu yra sukurta standartinė registracijos forma ir el. pašto tikrinimas nėra atliekamas.

Literatūros sąrašas

[Dex16] Dexi.io - web data extraction tool for professionals. –
URL: https://dexi.io/ . 2016.11.25

[Imp16] Import.io | Extract data from the web. –
URL: https://www.import.io/ .2016.11.27

[80l16] 80legs - Custom Web Scraping & Powerful Web Crawling. –
URL: http://80legs.com/ . 2016.12.02

[TPe10] Y. R. Tausczik and J. W. Pennebaker. The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods. Journal of Language and Social Psychology, Nr. 29(1), URL: http://www.cs.cmu.edu/~ylataus/files/TausczikPennebaker2010.pdf . 2010