

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Jure Ribič

**Medsebojno povezovanje vsebin v različnih
družabnih omrežjih**

DIPLOMSKO DELO
NA UNIVERZITETNEM ŠTUDIJU

Mentorica: doc. dr. Mojca Ciglarič

Ljubljana, 2012

Rezultati diplomskega dela so intelektualna lastnina Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavljanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje Fakultete za računalništvo in informatiko ter mentorja.



Št. naloge: 01791/2011

Datum: 01.12.2011

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **JURE RIBIČ**

Naslov: **MEDSEBOJNO POVEZOVANJE VSEBIN V RAZLIČNIH DRUŽABNIH OMREŽJIH**

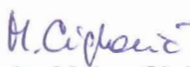
INTERCONNECTING CONTENT FROM DIFFERENT SOCIAL NETWORKS

Vrsta naloge: Diplomsko delo univerzitetnega študija

Tematika naloge:

Opredelite pojem spletnih družabnih omrežij in njihove značilnosti. Podrobneje opišite funkcionalnost in podatke največjih in pri nas najbolj obiskanih družabnih omrežij. Preverite, katera nudijo dostop do podatkov o uporabniških profilih tudi prek programskega vmesnika in kakšen način avtorizacije uporabljajo za tovrsten dostop. Na temelju teh ugotovitev naredite načrt aplikacije, ki bo omogočala zajem vsebine iz različnih izbranih družabnih omrežij in bo podatke tudi znala medsebojno povezovati. Aplikacijo tudi realizirajte s pomočjo aktualnih spletnih tehnologij. V zaključku kritično komentirajte, kako in zakaj bi se lahko uporabilo povezane podatke ter kako bi lahko aplikacijo še nadgradili.

Mentor:


doc. dr. Mojca Ciglarič



Dekan:


prof. dr. Nikolaj Zimic

IZJAVA O AVTORSTVU

diplomskega dela

Spodaj podpisani **Jure Ribič,**

z vpisno številko **63050099,**

sem avtor diplomskega dela z naslovom:

Medsebojno povezovanje vsebin v različnih družabnih omrežjih

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom (naziv, ime in priimek)
doc. dr. Mojce Ciglarič
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.)
ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki »Dela FRI«.

V Ljubljani, dne 15.2.2012

Podpis avtorja:

Zahvala

Zahvaljujem se mentorici doc. dr. Mojci Ciglarič, da mi je nudila strokovne nasvete in vsestransko pomoč pri izdelavi diplomskega dela.

Hvala tudi vsem prijateljem, ki so me podpirali, spodbujali, mi pomagali z nasveti in verjeli vame.

Zahvala gre tudi staršem, ki so mi omogočili študij in me med študijem podpirali. Brez njiju to ne bilo mogoče.

Kazalo

Povzetek	1
Abstract.....	2
1. Uvod	3
1.1. Področje raziskave	3
1.2. Cilji	4
1.3. Sorodna dela	4
1.4. Sorodne aplikacije.....	5
2. Spletna družabna omrežja.....	7
2.1. Opredelitev pojma spletnih družabnih omrežij in njihove značilnosti	7
2.2. Pregled in zgodovina družabnih omrežij	9
2.3. Največja družabna omrežja danes.....	10
2.4. Podrobnejši opis nekaterih večjih družabnih omrežij	10
2.4.1. Facebook.....	10
2.4.2. Google+	11
2.4.3. LinkedIn	12
2.4.4. Twitter	12
2.4.5. MySpace	13
2.4.6. Netlog	14
3. Raziskovana družabna omrežja	15
3.1. Zakaj takšna izbira?	15
3.2. Vsebina pridobljenih podatkov	15
3.2.1. Protokol za avtorizacijo – OAuth	15
3.2.2. Pridobljeni podatki s Facebooka	17
3.2.3. Pridobljeni podatki z Google+	18

3.2.4.	Pridobljeni podatki z LinkedIn-a	18
3.2.5.	Pridobljeni podatki s Twitterja.....	19
4.	Programska rešitev.....	21
4.1.	Specifikacije funkcionalnosti programske rešitve.....	21
4.2.	Zahteve	22
4.3.	Uporabljene tehnologije	22
4.3.1.	PHP	22
4.3.2.	CodeIgniter.....	22
4.3.3.	cURL	23
4.3.4.	MYSQL.....	23
4.3.5.	HTML5	23
4.3.6.	CSS.....	23
4.3.7.	JAVASCRIPT	24
4.3.8.	AJAX	24
4.4.	Podatkovni model.....	25
4.5.	Opis rešitve.....	27
4.5.1.	Registracija in prijava	27
4.5.2.	Prijatelji.....	40
4.5.3.	Prva stran.....	45
5.	Ugotovitve in možnosti za razširitve	47
6.	Sklep	51
	Kazalo slik.....	53
	Kazalo tabel.....	55
	Literatura.....	56

Seznam uporabljenih kratic in simbolov

API	programski vmesnik (ang. application programming interface)
RIS	raba interneta v Sloveniji
DNS	sistem domenskih imen (ang. domain name system)
MDP	MySpace razvojni center (ang. MySpace developer center)
OAuth	protokol za avtorizacijo (ang. open authorization)
XML	preprost jezik, ki nam omogoča format za opisovanje strukturiranih podatkov ali arhitektura za prenos podatkov in njihovo izmenjavo med več omrežji [54]. (ang. extensible markup language)
JSON	preprost format za izmenjavo podatkov, enostaven za ljudi, in sicer tako za branje, kot za pisanje. (ang. JavaScript Object Notation)

Povzetek

V diplomski nalogi smo obravnavali družabna omrežja, povezovanje med njimi in podatke, ki jih lahko pridobimo iz njih.

V prvem delu smo se osredotočili na družabna omrežja na splošno. Predstavili smo kratko zgodovino razvoja družabnih omrežij in našeli današnja najpopularnejša omrežja. Nekatera izmed le-teh smo tudi podrobno opisali.

V osrednjem delu smo podrobno analizirali izbrana družabna omrežja. Na kratko smo predstavili razloge za takšno izbiro in opisali protokol za avtorizacijo s platformami družabnih omrežij. Sledi poglobljena analiza o tem, katere podatke je moč pridobiti iz družabnih omrežij, kako so kategorizirani in katere pravice so zanje potrebne.

Zadnji del je namenjen opisu programske rešitve in ugotovitvam. Najprej smo predstavili tehnologije, ki smo jih uporabljali, nato pa še podatkovni model. Sledi podroben opis aplikacije, ki povezuje družabna omrežja med seboj in prikazuje podatke na enem mestu. Na koncu pa smo predstavili še rezultate, do katerih smo prišli med razvojem, in možne razširitve aplikacije.

Ključne besede: družabna omrežja, povezovanje družabnih omrežij, pridobivanje podatkov, vmesnik API.

Abstract

In this thesis we deal with social networks, linkages between them and what kind of information we can acquire from them.

In the first part we focus on the social networks in general, we have presented a brief history of the development of social networks and listed most popular networks today. Some of these networks are also described in details.

In the middle section we thoroughly analyzed selected social networks. We presented the reasons for choosing selected social networks and described the protocol for the authorization with social networking platforms. Followed by a detailed analysis of what information can be obtained from social networks, how they are categorized and what rights they require.

The last part is dedicated to the description of software solutions and our findings. First we present the technology we have used and then the data model. Follows a detailed description of the application that connects social networks with each other and displays the data in one place. Finally, we present the results which we obtained during the development of application and possible extensions.

Key words: social networks, integration of social networks, data retrieve, API interface.

1. Uvod

S pojavitvijo svetovnega spleta smo uporabniki kar na enkrat dobili dostop do velike količine informacij, ki se nahajajo na različnih spletnih straneh. Zelo pomembno vlogo pri komuniciranju človeka s tehnologijo imajo v procesu uporabe spletne strani uporabniški vmesniki, ki so močno povezani z uporabniško izkušnjo. Če je torej uporabnik zadovoljen, se bo na spletno stran še vračal. Ker so spletna družabna omrežja trenutno eden izmed popularnejših načinov spletne komunikacije, jih je potrebno oblikovati v skladu s pričakovanji in hkrati tudi zahtevami uporabnikov. Dobro uporabniško izkušnjo lahko ustvarimo le tako, da analiziramo uporabnost spletnih strani iz različnih vidikov in ponudimo najboljšo rešitev. V letu 2011, ko je bilo na spletu že 286 milijonov spletnih strani, se je med le-temi pojavilo tudi veliko družabnih omrežij. Uporabniki so sprva izbirali sebi najljubše družabno omrežje, s časom pa je prišlo do tega, da so si začeli ustvarjati profile na vseh popularnejših portalih. Nekateri zato, ker jih je tja pripeljal nekdo izmed prijateljev oziroma znancev, drugi pa zgolj zaradi lastnih interesov.

V tej diplomski nalogi bomo poskušali povezati nekatera večja družabna omrežja v eno, kjer bodo na enem mestu zbrane informacije teh omrežij. S tem bi uporabnikom družabnih omrežij prikazali podatke na enem mestu in jim prihranili čas, hkrati pa bi imeli boljši pregled nad podatki o posameznem prijatelju oziroma njegovimi povezavami (stiki ali prijatelji). Hkrati pa bomo preučevali tudi podatke, ki jih lahko pridobimo iz raziskovanih družabnih omrežij.

1.1. Področje raziskave

V diplomski nalogi se bomo osredotočili na družabna omrežja. Najpopularnejša med njimi bomo podrobno predstavili ter analizirali podatke, ki so na teh družabnih omrežjih na voljo. Preko analize podatkov se bo ugotavljalo, kako bi bilo smiselno povezati podatke med seboj.

1.2. Cilji

Cilj diplomske naloge je zgraditi aplikacijo, ki bo povezovala več družabnih omrežij in prikazala podatke le-teh na enem mestu ter omogočala objavo lastnih statusov in novic iz enega centra. S tem bi se skrajšal čas pregleda zadnjih aktualnih novic, saj bi bile le-te vidne na enem mestu.

1.3. Sorodna dela

Družabna omrežja so v zadnjih letih zelo popularno področje za raziskovanje [4]. Bila so preučevana iz različnih zornih kotov, in sicer od varnosti, psihološkega in sociološkega vidika, do zbiranja in analize podatkov. Pod slednje spada tudi to diplomsko delo, zato bomo v tem podpoglavju predstavili sorodna dela, ki pokrivajo to smer raziskav.

A. Narayanan in V. Shmatikov [1] sta se ukvarjala z algoritmi za razbijanje anonimizacije družabnih grafov omrežij. Ti grafi so sestavljeni iz vozlišč, ki ponazarjajo osebe, in povezav, ki predstavljajo povezanost med temi osebami. Imajo minimalno število podatkov, tako da je za posamezno vozlišče težko sklepati, katero osebo predstavlja. Z različnimi pristopi se nato poskuša takšen graf anonimizirati in zbrati čim več podatkov. Raziskovalca se navezujeta na to, da je večina profilov privzeto javnih, oziroma da je takšen vsaj del profila, ter da zelo malo število uporabnikov spremeni te nastavitve [2]. Raziskovalci iz Amerike in Nemčije A. Mislove, B. Viswanath, K. P. Gummadi in P. Drushel [3] so raziskovali, ali bi lahko ugotovili manjkajoče attribute uporabnikov s pomočjo družabnega grafa omrežja. S svojo raziskavo so prišli do podobnih zaključkov kot A. T. Fiore in J. S. Donath [48], ki pravi, da se pogosto povezujejo ljudje, ki imajo podobne lastnosti oziroma interese.

J. Bonneau in ostali [5] so se osredotočili na vprašanje, katere podatke lahko pridobijo iz družabnih omrežij. Podobno sta raziskovala tudi B. Krishnamurthy in C. E. Wills [53], ki sta se ukvarjala z izpostavljenostjo podatkov v družabnih omrežjih. Ugotovila sta, da je zaradi premajhne uporabe zasebnih nastavitve veliko profilov izpostavljenih in je posledično zelo preprosto pridobiti njihove podatke. Največje javno zbiranje podatkov družabnih omrežij sta izvedla D. H. Chau [6] in A. Mislove [7] s svojima raziskovalnima skupinama. Trije raziskovalci iz univerze v Kaliforniji (S. Ye, J. Lang, F. Wu [8]), so preučevali težave pri

pridobivanju podatkov iz omrežij, kjer so izpostavili delno zaprte profile uporabnikov, dinamične strani ponudnikov družabnih omrežij in spremembe dizajna strani profilov uporabnikov. Vse to onemogoča lažje in hitrejše zbiranje podatkov. Predvsem uporabniki, ki se odločijo, da njihovi profili ne bodo javno dostopni, s tem močno otežijo preprosto zbiranje njihovih podatkov ter hkrati vsaj malo poskrbijo za varovanje svoje zasebnosti.

1.4. Sorodne aplikacije

Obstaja kar nekaj aplikacij, ki znajo povezati družabna omrežja, vendar prevladujejo predvsem namizne aplikacije. Slabost namiznih aplikacij je, da je ob vsaki menjavi računalnika potrebna ponovna namestitev aplikacije, oziroma dostop do nje celo ni možen, če uporabljamo kakšno javno dostopno točko, kjer uporabnik nima pravic za namestitev aplikacij. Kot primer spletne aplikacije smo našli storitev Friendfeed, ki omogoča združevanje družabnih strani in storitev. V Friendfeed lahko dodamo praktično vsak družabni medij, ki omogoča RSS, in potem uporabnik spremlja vse na enem mestu. Pri namiznih aplikacijah pa je spisek veliko daljši: Miranda, Trillian, Nimbuzz, Digsby, Paltalk, Yoono, itd. [32]. Vsem je skupno, da podpirajo večja družabna omrežja tako, da ima uporabnik na razpolago spisek prijateljev določenega družabnega omrežja, s katerimi se lahko pogovarja neposredno preko aplikacije in spremlja tudi spremembe statusov svojih prijateljev. Nimajo pa te aplikacije podrobnega pregleda zgodovine objav statusov. Prav tako večina teh programov ne omogoča ogleda galerij in videov. Na njih si je možno pogledati le zadnje objavljene slike. Sledenje komentarjem ni omogočeno, prav tako si ne moramo ogledovati javnih profilov, ki jih nimamo med svojimi stiki. Težava teh aplikacij je tudi delovanje, saj se v primeru, če storitev ponudnika aplikacije ne deluje, ne moremo prijaviti v aplikacijo in posledično ne moremo uporabljati storitve, čeprav vsa ostala družabna omrežja delujejo.



Slika 1. Namizna aplikacija Yoono.

2. Spletna družabna omrežja

V tem poglavju bomo naprej povedali nekaj besed o družabnih omrežjih na splošno, nato bomo preleteli zgodovinski razvoj le-teh ter predstavili nekatera večja oziroma popularnejša družabna omrežja ta hip. Druga polovica poglavja pa podrobneje opisuje šest družabnih omrežij: Facebook, LinkedIn, Twitter, Google+, MySpace in Netlog. Za prva štiri družabna omrežja smo se odločili, ker so globalno najbolj popularna, MySpace predstavlja družabno omrežje, ki je bilo včasih med bolj obiskanimi, zadnja leta pa obisk strani upada. Netlog pa je zanimiv zaradi priljubljenosti med slovenskimi uporabniki.

2.1. Opredelitev pojma spletnih družabnih omrežij in njihove značilnosti

Spletna družabna omrežja so internetna omrežja, v katera se združujejo različne generacije ljudi oziroma posamezniki, ki jih povezujejo določeni skupni interesi. Ta omrežja se lahko formirajo v sklopu podjetij, šol ali univerz, združenj, zabave in podobno. Z razvojem tehnologije se dopolnjujejo in nadgrajujejo tudi sama družabna omrežja. Omogočajo izgradnjo javnih ali poljavnih profilov in oblikovanje seznama drugih uporabnikov, s katerimi si delijo povezave. Predstavljajo nek nov socialni in ekonomski fenomen, ki je pritegnil že na milijone uporabnikov, z njimi pa je prišlo tudi do novega načina komuniciranja ter nove vrste medosebnih odnosov.

Obstaja več tipov družabnih omrežij [38]:

- Družabne povezave (ohranjanje stikov s prijatelji in družinskimi člani; ta vrsta omrežij je najbolj razširjena; primer Facebook [41]).
- Izmenjava multimedijskih vsebin (te vrste omrežij omogočajo preprosto skupno rabo video vsebin in fotografij na spletu; primer Youtube [42]).
- Profesionalna omrežja (ta vrsta omrežij je zasnovana za zagotavljanje karierni rasti; nekatera omrežja te vrste razpolagajo s forumi za strokovnjake, druga pa so osredotočena na določene poklice ali interese; primer LinkedIn [47]).

- Informacijska omrežja (to vrsto omrežij sestavljajo ljudje, ki iščejo odgovore na vsakdanje težave; primer Do-It-Yourself Community [49]).
- Izobraževalna omrežja (omrežja sestavljajo študenti z namenom sodelovati z drugimi študenti na akademskih projektih; primer The Student Room [51]).
- Hobiji (spletne strani, ki temeljijo na uporabniku najljubših hobijih; povezovanje ljudi z istimi interesi; primer Oh My Bloom [50]).
- Akademsko omrežje (namenjeno je akademskim raziskovalcem, ki želijo deliti svoje raziskave; primer Academia.edu [52]).

Kljub raznolikostim med družabnimi omrežji, ki se kažejo navzven, pa je svet spletnih omrežij dokaj enovit, saj gre v vseh primerih za izpostavljanje uporabnikov okolju in za samopromocijo - predvsem pravnih oseb. Razvijalci družabnih omrežij so ponudili virtualno okolje, ki mu uporabniki zaupajo in nato svoje podatke po lastni presoji nudijo na vpogled le določenemu krogu prijateljev. Pri vseh družabnih omrežjih so prisotne varnostne nastavitve, ki določajo prikaz vsebine, vendar mora bit uporabnik zelo pozoren pri nastavljanju le-teh, saj v primeru nepazljivosti lahko omogoči uporabnikom ogled njegove vsebine, čeprav si tega ne želi. V raziskavah [39] so bili prikazani zaskrbljujoči podatki: le 20-odstotkov uporabnikov spremeni privzete nastavitve glede zasebnosti (kaj od objavljenih vsebin uporabnika je vidno ostalim uporabnikom), ki so na začetku nastavljene običajno zelo »odprto«. Zaradi občutka resnosti in zaupanja postajajo družabna omrežja zlata jama osebnih podatkov, ki jih uporabniki objavljajo vse več. Osebni podatek pa ni zgolj ime in priimek ali naslov posameznika, temveč tudi fotografija in video vsebine. Fotografiranje osebe brez njene privolitve je sicer dovoljeno, sporna pa je objava fotografij v tiskanem in spletnem mediju. Za objavo fotografij je potrebno imeti dovoljenje fotografirane osebe. S pojavitvijo družabnih omrežij pa je postalo praktično nemogoče nadzorovati objavljanje fotografij, saj drugi uporabniki lahko naše fotografije objavijo, nas na njih označijo ter jih komentirajo brez naše vednosti in privolitve. Slednje je najbolj značilno za mlajše uporabnike družabnih omrežij, kjer se nekateri ne zavedajo posledic. Vendar Oblakova [29] poudarja, da bi bilo napačno sklepati, da najstnikom ni mar za njihovo zasebnost, saj nekatere raziskave kažejo, da so mladostniki zelo premišljeni o tem, katere osebne informacije bodo s kom delili. Raziskave so pokazale, da so težave predvsem v omejenosti vmesnikov, ki ponujajo premalo možnosti za nastavljanje zasebnosti. Facebook se je na to odzval in v letu 2011 temeljito spremenil ter razširil vmesnik z nastavitvami za zasebnost.

Družabna omrežja nudijo uporabnikom preprost način komuniciranja s prijatelji, sorodniki in znanci. Poleg tega imajo možnost, da konstantno zadovoljujejo nekatere lastne želje. Raziskave med mladimi uporabniki so pokazale, da je glavni razlog za njihovo uporabo

družabnih omrežij ostati v stiku s prijatelji ter želja po pozitivnih povratnih informacijah. Povratne informacije vrstnikov naj bi močno vplivale na uporabnikovo samospoštovanje. Samospoštovanje pa posledično vpliva na psihično blagostanje. Izkazalo se je, da je uporabnikom všeč odprtost in preglednost družabnih omrežij, saj lahko na preprost in hiter način pridobijo informacije o drugih. Pridobivanje informacij o osebi, za katero se tako ali drugače zanimajo, pa je eden izmed pomembnejših razlogov uporabe družabnih omrežij.

V množici podatkov objavljenih na družabnih omrežjih se moramo vprašati, kako je z zasebnostjo in varnostjo na teh omrežjih. Kot razlog, zakaj uporabniki ignorirajo tveganje, ki se tiče objave njihovih osebnih informacij, pa se kaže pomanjkanje zavedanja nevarnosti. Uporabniki namreč ne vidijo svoje zasebnosti in varovanja svojih osebnih informacij na spletu kot nekaj slabega oziroma nevarnega.

2.2. Pregled in zgodovina družabnih omrežij

Družabna omrežja so se najprej pojavila v Združenih državah Amerike, ko je bila leta 1995 ustanovljena spletna stran Classmates.com in leta 1996 spletna stran SixDegrees.com, ki sta imeli vse lastnosti novjših družabnih omrežij. Širjenje omrežij pa se je začelo v začetku tega tisočletja z razvojem LinkedIn-a in Flickrja. Za veliko priljubljenost pa so poskrbeli Friendster, MySpace in Facebook predstavljeni med letoma 2002 in 2004.



Slika 2. Pregled razvoja družabnih omrežij.

Družabna omrežja so v zadnjih letih postala priljubljena tudi v Sloveniji. Raziskava RIS v letu 2007 je pokazala, da je med desetimi najbolj obiskanimi spletnimi stranmi v Sloveniji tudi družabno omrežje MySpace, ta trend se je v zadnjih letih le še povečeval. Tako naj bi bilo v letu 2011 na Facebooku registriranih že skoraj 700 000 Slovencev.

2.3. Največja družabna omrežja danes

Najpopularnejša družabna omrežja, merjena po številu unikatnih uporabnikov na mesec, v novembru 2011 [40] so bila:

	Družabno omrežje	Število unikatnih mesečnih obiskovalcev
1	Facebook	700,000,000
2	Twitter	200,000,000
3	LinkedIn	100,000,000
4	MySpace	80,500,000
5	Ning	60,000,000
6	Google+	32,000,000

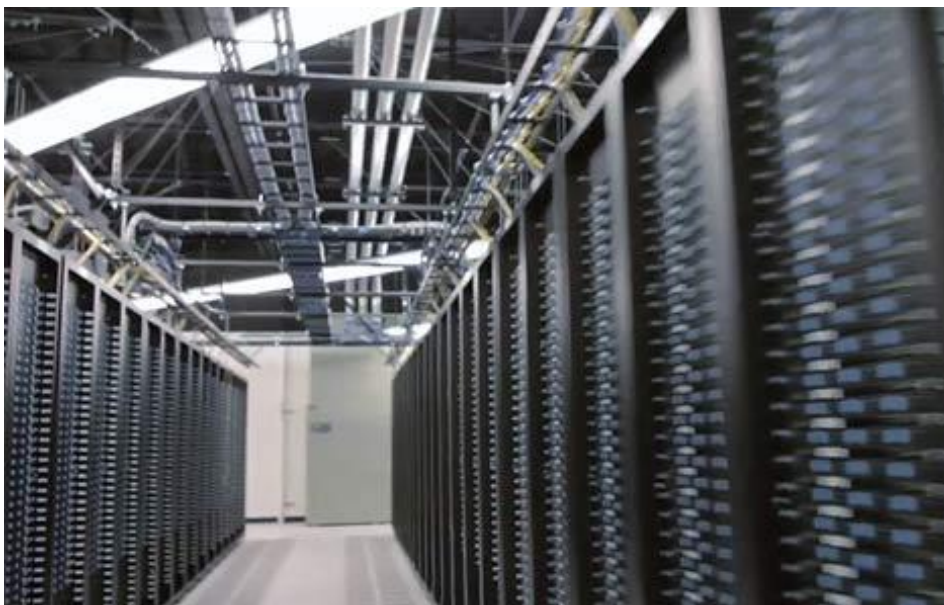
Tabela 1. Število unikatnih mesečnih obiskovalcev družabnih omrežij. [40]

2.4. Podrobnejši opis nekaterih večjih družabnih omrežij

2.4.1. Facebook

Spletno družabno omrežje Facebook je bilo ustanovljeno 4. februarja 2004. 7 mesecev kasneje, 11. septembra 2006, pa je postalo dostopno vsakomur. Pripada podjetju Facebook, Inc. Brezplačen dostop do strani uporabnikom omogoča, da se povežejo in tako lažje komunicirajo z ostalimi ljudmi istega omrežja. Stran ima, po podatkih Google's DoubleClick, več kot 800 milijonov aktivnih uporabnikov ter več kot 600 milijard ogledov na mesec. Vsak dan je naloženih okoli 100 milijonov slik, mesečno je objavljenih približno 30 milijard vsebin, dnevno pa nastane za več kot 150 TB dnevniških zapisov.

Infrastruktura je sestavljena tako, da podpira hitro širjenje. Poganja jo več kot 60 000 strežnikov.



Slika 3. Facebookov podatkovni center. [13]

Facebook je največji uporabnik memcached tehnologije (tehnologijo se uporablja za pospeševanje dinamičnih baz podatkov spletnih strani, s čimer zmanjšamo pretok podatkov in poizvedb) in ima eno izmed največjih MySQL baz (poleg tega uporabljajo še Cassandra, ki je NoSQL podatkovna baza, prav tako razvita s strani Facebooka). Podatki so shranjeni v 9-ih podatkovnih centrih, ki so vsi locirani znotraj ZDA. Za shranjevanje milijard slik skrbi Haystack, ad-hoc rešitev, ki jo je razvil Facebook in prinaša optimizacijo in samo dodajanje (brez brisanja) novih slik. Stran je v večini napisana v PHP-ju v navezi z HipHop, ki transformira PHP kodo v močno optimizirano C++, nato pa sledi prevajanje z g++, kjer se zgradi binarna koda. S tem je Facebook dosegel nekje 50-odstotno zmanjšanje obremenjevanja CPU-ja s streženjem enakega obsega spletnega prometa, API poizvedbe pa lahko servirajo dvakrat več prometa. Razvil je tudi ogrodje (ang. framework), ki uporablja RPC (remote procedure call). Le-ta poveže skupaj vse vrste storitev v katerem koli programskem jeziku gnanem na kateri koli platformi.

2.4.2. Google+

Je najnovejše družabno omrežje, ki je trenutno najbolj v vzponu. Njegov začetek sega v 28. junij 2011, registracije pa so bile za celotno javnost odprte 20. septembra istega leta. Hkrati je bila objavljena tudi platforma API za razvijalce. Trenutno je Google+ storitev, ki je na voljo kot spletna aplikacija, načrtujejo pa tudi aplikacijo za namizje. Že v juliju je novo družabno

omrežje doseglo 10 milijonov registriranih uporabnikov, oktobra pa je preseglo številko 40 milijonov .

Programski jezik je standarden za Google. Java servlets uporabljajo za kodo na strežniku, skriptni jezik javascript pa uporabljajo na strani uporabnika za grafični vmesnik, ki je večinoma zgrajen z odprtokodnim ogrodjem Closure.

2.4.3. LinkedIn

Poslovno družabno omrežje LinkedIn je bilo ustanovljeno decembra 2002, javno pa je uradno začelo delovati 5. maja 2003. Po zadnjih podatkih (november 2011) je registriranih že 135 milijonov uporabnikov iz več kot 200 držav.

LinkedIn strežniki imajo nameščen Solaris operacijski sistem. Koda je v 99-odstotkih napisana v programskem jeziku Java z izjemo nekaterih delov, ki so napisani v C++, Ruby on Rails ter Groovy/Grails. Za hranjenje podatkov uporabljajo Voldermort, ki je NoSQL podatkovna baza. Skrbi za avtomatsko podvajanje podatkov na več strežnikih ter za transparentnost pri izpadu strežnikov. Podatki pa so namenjeni samo za branje. LinkedIn posodobitve pa se prenašajo preko Hadoop-a (ogrodja, ki je del Apacheja in omogoča porazdeljeno obdelavo velike količine podatkov prek grozdov računalnikov) v Voldermort, kjer se potem poskrbi za prehod na nove podatke .

V oktobru 2008 je LinkedIn omogočil tudi platformo za razvijanje, ki omogoča drugim spletnim stranem, da preko vmesnika dostopajo do podatkov omrežja.

2.4.4. Twitter

Twitter je spletno družabno omrežje in mikroblog storitev, ki omogoča uporabnikom objavljanje in branje objav omejenih na 140 znakov (»tweets«) . Sprva je bil prvi prototip Twitter uporabljen kot storitev znotraj podjetja, polna različica pa je bila javno predstavljena 15. julija 2006. S hitro širitvijo pa je v letu 2011 Twitter dosegel preko 200 milijonov uporabnikov.

Družba je doživela hitro rast. V letu 2007 je bilo v enem četrletju objavljenih okoli 400 000 tweetov. V enem letu se je število tweetov na četrletje povečalo za faktor 250, v letu 2011 pa dosega 200 milijonov objav vsak dan (skupaj za približno 20TB podatkov), za katerimi stoji 200 milijonov uporabnikov. Slednji so tudi zadolženi za preko 1,6 milijard iskalnih poizvedb dnevno. Julija 2011 je bila ustvarjena že milijonta Twitter aplikacija, za katere pri Twitterju pravijo, da igrajo ključno vlogo pri pomoči ljudem dobiti najboljše iz Twitterja. Zanimiv je

tudi podatek, da je Twitterjev API promet približno 10-krat večji od prometa oziroma dostopov do same strani.

Po ocenah naj bi Twitter imel med 2000 in 4000 strežnikov (točna številka ni bila objavljena v javnosti). Kot spletni strežnik uporabljajo Mongrel, ki je odprtokodna http knjižnica, in spletni strežnik napisan v Rubyu, ki poganja Ruby spletne aplikacije. Pri izdelavi aplikacij pa si pomagajo z odprtokodnim ogrodjem ROR. Poleg naštetega uporabljajo še orodje za spremljanje razpoložljivih virov omrežja Munin, ki jim pomaga pri analiziranju težav, in Nagios, odprtokodni računalniški sistem in programsko orodje za spremljanje omrežja. Ta analizira gostitelje in storitve ter opozarja, ko gre kaj narobe. Twitter je v začetku za podatkovni center uporabljal klaster MySQL strežnikov z memcached tehnologijo, zaradi obremenjenosti pa so začeli uporabljati še NoSQL. NoSQL podatkovne baze, za razliko od klasičnih podatkovnih baz, ne potrebujejo tabelarične sheme, običajno ne uporabljajo »join« operacij in so v večini primerov horizontalne.

2.4.5. MySpace

Spletno družabno omrežje MySpace se je pojavilo avgusta 2003. Med leti 2005 in 2008 je bilo najbolj obiskano družabno omrežje na svetu, nato pa ga je v aprilu 2008 prehitel Facebook. Od takrat naprej število uporabnikov tega omrežja le še upada, čeprav ima mesečno še vedno več kot 80 milijonov unikatnih obiskovalcev. Posebnost tega družabnega omrežja je glasba, saj lahko vsak izvajalec naloži lastne skladbe na svoj profil in se s tem promovira. Omrežje pa omogoča tudi oblikovanje profila po lastnih željah. Vse kar je za to potrebno, je nekaj znanja html in css.

MySpace za svoje delovanje uporablja širok nabor različnih tehnologij. Storitve UltraDNS, ki upravlja z DNS-jem, je zadolžena za obravnavanje IP zahtev uporabnikov. Na podlagi zasedenosti strežnikov porazdeljuje promet na manj zasedene strežnike. Proxy strežnik je postavljen med uporabnika in glavni strežnik Myspacea. Njegova naloga je serviranje predpomnjenja (ang. cached) vsebine uporabnikom, s čimer se zmanjša število zahtev po aplikacijskih strežnikih Myspacea. Podatkovna baza je sestavljena iz gruč Isilon IQ 1920i strežnikov. Programski jezik je .NET 2.0, vse aplikacije pa tečejo na Windows platformi.

Platforma za razvijalce MDP je bila aktivirana v začetku oktobra 2008 in bazira na Open Social API, ki ga je razvil Google.

2.4.6. Netlog

Netlog je belgijsko družabno omrežje posebej namenjeno evropski mladini. Stran je bila ustanovljena julija 2003. Omrežje ima tudi posebno tehnologijo lokalizacije, ki zagotavlja, da je celotna vsebina jezikovno prirejena za profil uporabnika. Trenutno je na voljo v 40-ih jezikih in ima več kot 93 milijonov uporabnikov.

3. Raziskovana družabna omrežja

V nadaljevanju sledi nekaj besed o tem, katera družabna omrežja smo izbrali za raziskovanje in razlogi za takšno odločitev. Podrobno je predstavljena vsebina, ki jo je mogoče pridobiti preko vmesnikov API, ter protokol OAuth za avtentikacijo s platformami družabnih omrežij.

3.1. Zakaj takšna izbira?

Facebook je trenutno najpopularnejše družabno omrežje, v kategoriji mikroblogov je daleč najbolj razširjen Twitter, LinkedIn pa spada med popularnejša poslovna družabna omrežja. Google+ je družabno omrežje, ki je trenutno najbolj v vzponu. Vsa družabna omrežja razpolagajo z odličnimi vmesniki API in tako razvijalcem olajšajo delo pri komuniciranju z njihovimi platformami ter pridobivanju podatkov.

3.2. Vsebina pridobljenih podatkov

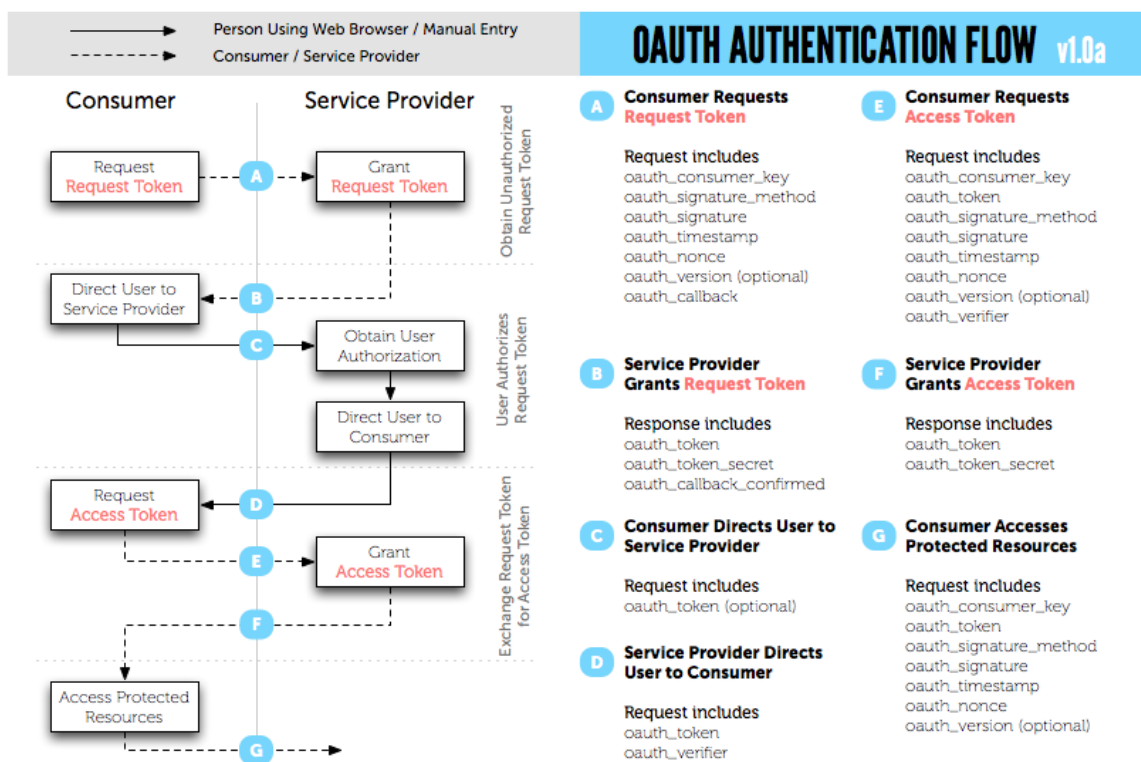
Vsa obravnavana družabna omrežja ponujajo svojo API platformo za pridobivanje podatkov. Nihče ne ponuja vseh podatkov objavljenih v omrežju, imajo pa podoben oziroma enak način avtorizacije.

3.2.1. Protokol za avtorizacijo – OAuth

Pri vseh obravnavanih omrežjih smo za avtorizacijo z API vmesnikom uporabljali OAuth protokol [27]. To je odprt protokol, ki dovoljuje varno avtorizacijo s preprostimi in standardnimi metodami. Omogoča, da uporabniki upravljajo z žetoni (ang. »tokens«), namesto da uporabljajo lastne pravice za dostop (uporabniško ime in geslo), ki jih ima pri

določenem ponudniku storitev. Vsak žeton omogoča dostop do specifične strani za točno določene stvari in za točno določeno obdobje.

OAuth protokol se je začel razvijati v novembru 2006, ko so ugotovili, da ni odprtih standardov za dostope do API platform. Tako je bila julija 2007 pripravljena začetna specifikacija protokola, 3. oktobra 2007 pa je bila objavljena različica OAuth 1.



Slika 4. Prikaz toka avtorizacije s protokolom OAuth 1.0 [35].

V sredini leta 2010 je bila objavljena že druga različica protokola, ki pa ni združljiva s prejšnjo različico. Ključne razlike med njima so:

- Boljša podpora za namizne in mobilne aplikacije.
- OAuth 2.0 ne zahteva več kriptografije pri uporabniku. Aplikacija lahko izvede zahtevo preko protokola HTTPS.
- Podpisi so veliko manj kompleksni, ker ni potrebnih dodatnih razčlenjevanj, urejanj in kodiranja.
- Dostopni žetoni so kratkotrajni, pri različici 1.0 so bili lahko shranjeni za eno leto ali pa celo niso potekali. Pri novi različici pa so uvedli osveževanje žetonov.

- Ločevanje vlog med strežnikom odgovornim za obdelavo zahtev OAuth in strežnikom zadolženim za avtorizacijo uporabnikov.

3.2.2. Pridobljeni podatki s Facebooka

Facebookov vmesnik API uporablja protokol OAuth 2.0 in je tudi največji uporabnik tega protokola. Število dostopov do Facebookovega vmesnika ni omejeno. Njegov vmesnik API, poimenovan »Graph API«, je jedro platforme Facebook in razvijalcem omogoča branje in pisanje podatkov.

Njegov vmesnik API je najobsežnejši med vmesniki, ki smo jih raziskovali. Razdeljen je na:

- Uporabnike,
- prijatelje,
- statuse,
- strani,
- skupine,
- albume,
- slike,
- povezave,
- objave,
- beležke,
- sporočila,
- videe,
- vprašanja,
- dogodke in
- komentarje.

Kljub obširnemu dostopu do podatkov pa preko vmesnika API ne moremo pridobiti nekaterih bolj osebnih podatkov, kot so na primer telefonske številke in elektronski naslovi.

Nekateri podatki pa so dodatno varovani še s tako imenovanimi »razširjenimi pravicami«, kjer moramo pri dostopu do teh podatkov preko vmesnika podati dodatne parametre, s katerimi se uporabnika še dodatno opozori, do katerih podatkov bo aplikacija dostopala in če se uporabnik s tem ne strinja, lahko zavrne dostop aplikacije do teh podatkov.

3.2.3. Pridobljeni podatki z Google+

Najmanj trenutno ponuja Googlov vmesnik API, kar je v bistvu tudi razumljivo, saj se je razvil zadnji. Vmesnik je bil javno dostopen šele konec septembra 2011. Prav tako kot Facebook tudi Google API ponuja OAuth 2.0 protokol za avtentikacijo. Število dostopov do vmesnika je omejeno na 1000 dostopov dnevno, je pa možno Google tudi zaprositi, da nam število dostopov poveča.

Vmesnik API je trenutno ločen v 3 skupine:

- Ljudje,
- aktivnosti,
- komentarji.

Tako preko vmesnika ne moremo pridobivati povezav, zasebnih sporočil, obvestil, hkrati pa ne moremo pridobiti niti lastnega elektronskega naslova, ki smo ga uporabili pri registraciji.

3.2.4. Pridobljeni podatki z LinkedIn-a

Družabno omrežje LinkedIn za avtentikacijo z API platformo uporablja različico 1.0 protokola OAuth. Njihov API omogoča preprosto in konsistentno predstavitev ljudi, podjetji, delovnih mest in povezav med njimi. Podatke se lahko pridobi v formatu XML ali JSON.

Podatki so razdeljeni v 5 večjih skupin:

- Ljudje in povezave,
- skupine,
- podjetja,
- delovna mesta,
- komunikacija.

Iz dokumentacije je razvidno, da so pomembnejši podatki, ki se jih ne da pridobiti preko vmesnika, elektronski naslov in telefonske številke. Kljub temu, da lahko uporabljamo klice za pridobivanje zasebnih sporočil, pa ta še niso podprta. Trenutno kot rezultat vrne, da uporabnik nima pravic. Razvijalci pri LinkedIn-u sporočajo, da delajo na tem, ni pa še znano, kdaj bo to podprto.

Tudi LinkedIn vmesnik API ima omejeno število dostopov, ki so razdeljeni v tri skupine: dostop aplikacije, dostop uporabnika in dostop razvijalca. Število dostopov je omejeno na dan in je različno glede na podatke, do katerih dostopamo. Na primer aplikacija lahko v eni uri

dostopa do 100 000 profilov, pošlje 5000 sporočil, pridobi 500 000 posodobitev v omrežju, itd.

3.2.5. Pridobljeni podatki s Twitterja

Tako kot za LinkedIn tudi za Twitter velja, da uporablja še prvo verzijo OAuth protokola ter dopušča, da je veljavnost žetona pri avtentikaciji brez omejitev, torej da ne poteče. Število možnih dostopov je 350 na uro.

Twitterjev vmesnik API je razdeljen na:

- Objave,
- iskanje,
- direktna sporočila,
- prijatelje in privrženice,
- uporabnike.

Prav tako kot vsi prejšnji vmesniki API, pa tudi zadnji med opisanimi ne dopušča pridobitve elektronskega naslova.

4. Programska rešitev

Na podlagi aplikacije, ki smo jo razvili, bomo v tem poglavju predstavili tehnologije, ki smo jih uporabljali pri razvijanju, podrobno opisali podatkovni model, na koncu pa je natančno opisana zgradba in delovanje aplikacije.

4.1. Specifikacije funkcionalnosti programske rešitve

Z aplikacijo želimo skupaj povezati več družabnih omrežij. Glavni cilj je ta, da bi na enem mestu videli podatke z vseh povezanih družabnih omrežij in tako olajšali uporabniku pregledovanje novosti, hkrati pa bi povezali vse njegove stike v pregledno celoto. V primeru da bi vsi vmesniki API omogočali pretok vseh podatkov z družabnih omrežij, bi aplikacije kot je naša, lahko v celoti nadomestile obiskovanje strani omrežij, saj bi lahko z ene točke upravljali s celotno vsebino. Tega so se zavedali tudi razvijalci družabnih strani in vmesnikov API, zato so njihove vmesnike omejili ter s tem preprečili razpoložljivost vseh podatkov, hkrati pa je večina vpeljala še omejeno število dostopov do vmesnikov (omejenih bodisi na uporabnika, aplikacijo, dan ali uro). Zato smo bili primorani sklepati kompromise. Omejevanje dostopov do vmesnikov smo ublažili s predpomnjenjem (tako se osvežuje podatke preko vmesnikov API na vsako uro), pri povezovanju stikov smo morali namesto telefonskih števil in elektronskih naslovov uporabljati kombinacijo imena, priimka, rojstnega kraja in datuma. Prav tako je omejeno pošiljanje zasebnih sporočil preko vmesnika API, saj nekatera omrežja tega ne podpirajo oziroma tega nimajo implementiranega, tako lahko trenutno sporočila samo sprejemamo. Ena izmed funkcionalnosti pa je tudi objavljanje statusov, saj lahko preko ene forme objavimo zelen nov status na vseh omrežjih hkrati.

4.2. Zahteve

Za uporabo aplikacije je potreben spletni brskalnik, ki mora imeti omogočen JavaScript in podpira HTML5 in CSS3. Zato smo omejeni na naslednje brskalnike:

- Internet Explorer 9 ali novejši,
- Firefox 3.6 ali novejši,
- Chrome 9 ali novejši,
- Safari 5 ali novejši,
- Opera 11 ali novejša.

4.3. Uporabljene tehnologije

Spletna aplikacija se izvaja na strežniku z operacijskim sistemom Linux, kjer je nameščen Apache, PHP 5.3 in baza MySQL. Kot programski jezik se posledično uporablja PHP z dodatkom html5 in JavaScript s spletno tehniko AJAX. Kot razširitev pri php-ju uporabljamo ogrodje Codeigniter, pri skriptnem jeziku JavaScript pa se uporablja knjižnica jquery. Klici vmesnikov API pa potekajo preko cURL. Za programiranje uporabljamo IDE programske orodje Netbeans, različne spletne brskalnike za testiranje aplikacije ter grafično programsko orodje photoshop za oblikovanje.

4.3.1. PHP

PHP (Personal Home Page Tools) je odprtokodni skriptni programski jezik, ki se uporablja za razvoj dinamičnih spletnih strani. Primarno teče na spletnem strežniku, kjer jemlje izvorno kodo PHP za vhod in generira spletno stran na izhodu (kot del PHP-ja je možno poganjanje skript v ukaznem načinu in kreiranje grafičnih aplikacij).

4.3.2. CodeIgniter

CodeIgniter je močno odprtokodno ogrodje, zgrajeno za PHP razvijalce, ki potrebujejo preprosto orodje za razvijanje spletnih aplikacij. Cilj ogrodja je omogočiti razvijalcem hitrejše

razvijanje in ne pisanje kode iz nič, saj zagotavlja bogat nabor knjižnic, ki so pogosto potrebne za razvoj aplikacij. Ogrodje temelji na popularnem vzorcu MVC, kjer ločimo tri nivoje: model, pogled in upravljaliec. Slednji skrbi za komunikacijo s »front-end« delom aplikacije, pogled je namenjen prikazu podatkov, model pa je povezan s podatki. Skrbi za pridobivanje, pošiljanje in manipuliranje s podatki.

4.3.3. cURL

To je računalniški projekt, ki prinaša knjižnico libcurl in orodje za prenos podatkov cURL, ki temelji na sintaksi URL. PHP podpira knjižnico libcurl, ki omogoča povezovanje in komuniciranje z različnimi tipi strežnikov in veliko različnimi protokoli. Med pomembnejšimi protokoli so: http, https, ftp, gopher, telnet, dict, file in ldap. Podpira pa tudi certifikate.

4.3.4. MYSQL

Je najpopularnejša odprtokodna implementacija za upravljanje z relacijskimi podatkovnimi bazami, ki za delo s podatki uporablja jezik SQL. Napisana je v programskem jeziku C in C++, deluje na večini danes poznanih operacijskih sistemov ter nudi programski vmesnik za veliko število danes uporabljenih programskih jezikov.

4.3.5. HTML5

HTML5 je jezik za oblikovanje in predstavitev vsebin na spletu. To je že 5. revizija standarda HTML in je bila novembra 2011 še vedno v razvoju. Njegov glavni cilj je izboljšati podporo za najnovejšo multimedijo, pri čemer pa je berljiv s strani ljudi, dosledno pa ga razumejo tudi računalniki in naprave.

4.3.6. CSS

CSS (angl. Cascading Style Sheets) je slogovni jezik, ki se uporablja za opisovanje predstavitev semantičnih dokumentov (videz in oblika) napisanih v označevalnem (ang. markup) jeziku. Najpogosteje se uporablja za oblikovanje spletnih strani napisanih v HTML in XHTML, možna pa je tudi uporaba na vseh vrstah XML dokumentov. Primarno je namenjen ločitvi vsebine dokumenta od predstavitve. Ima enostavno sintakso in uporablja veliko angleških besed za določitev različnih lastnosti sloga. Slogi so sestavljeni iz seznama pravil, vsako pravilo pa je sestavljeno iz enega ali več selektorjev in blokov. Slednji pa

vsebujejo seznam izjav, ki so sestavljene iz predmeta in pripisane vrednosti in ju ločuje dvopičje. Objavljena je že različica CSS3, ki jo je W3C začel priporočati v letu 2011.

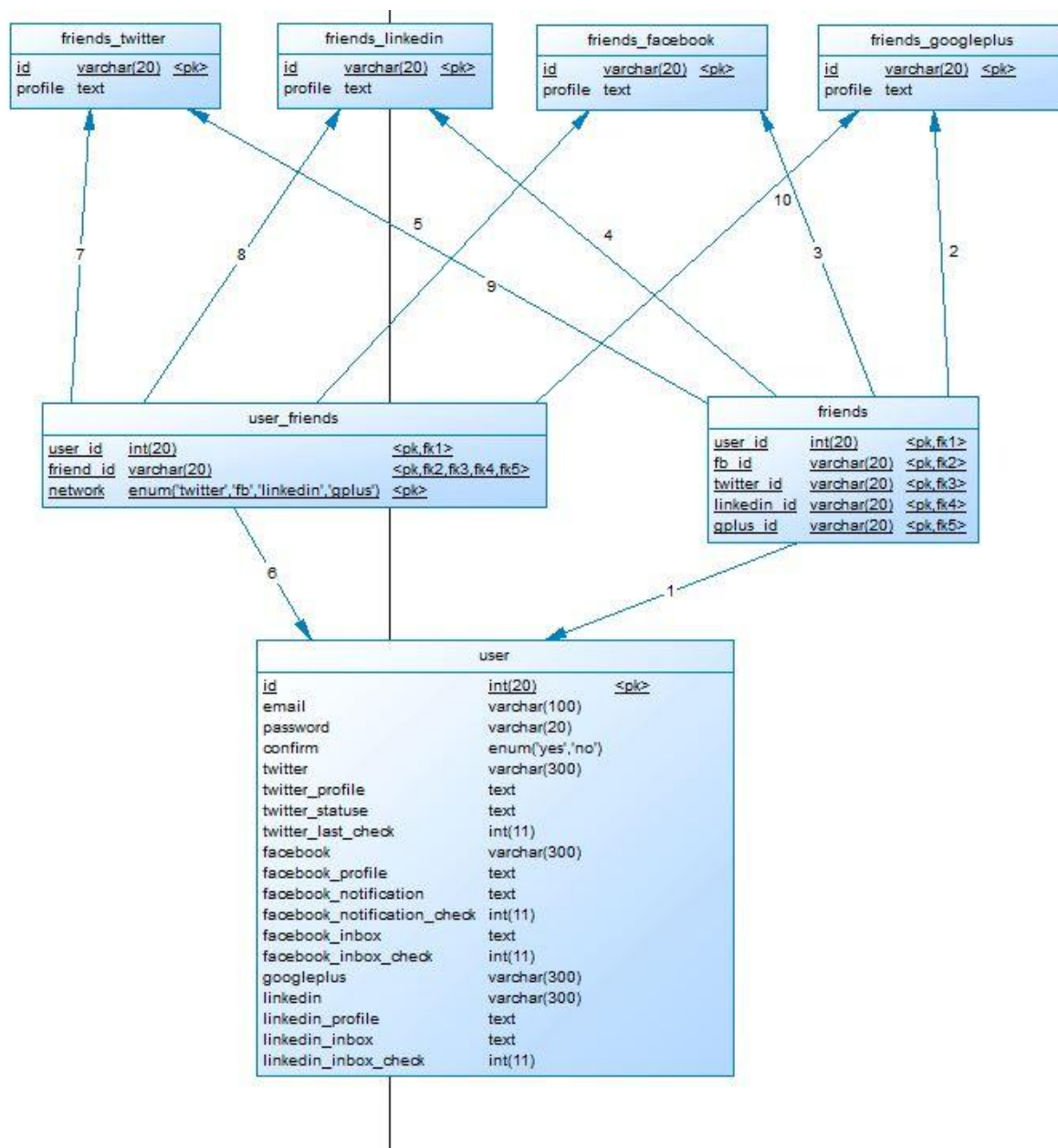
4.3.7. JAVASCRIPT

JavaScript je objektni skriptni programski jezik, ki ga je razvil Netscape z namenom ustvarjanja interaktivnih spletnih strani. Sintaksa jezika delno sledi programskemu jeziku C, ki prav tako nima vgrajenih vhodno izhodnih funkcij. Napisano kodo se vgradi ali vključi v HTML, da opravlja naloge, ki niso možne samo s statično stranjo. Na žalost vsi spletni brskalniki ne izpostavijo enakih objektov za uporabo, zato je potrebno včasih napisati več različnih funkcij za isto stvar, da deluje v vseh brskalnikih.

4.3.8. AJAX

Kratica pomeni asinhroni JavaScript in XML. Tehnologija AJAX je skupek medsebojno povezanih spletnih razvojnih tehnik, ki se uporabljajo za ustvarjanje interaktivnih spletnih aplikacij. Spletne strani s pomočjo te tehnologije izmenjujejo podatke s strežnikom v ozadju brez potrebe po ponovnem nalaganju strani.

4.4. Podatkovni model



Slika 5. Podatkovni model spletne aplikacije.

Podatkovni model je sestavljen iz 7 tabel. V tabeli »user« se nahajajo vsi registrirani uporabniki. Obvezni polji sta samo »email« in »password«, ki je kriptiran s hash funkcijo MD5 ter »id«, ki se generira avtomatsko. Polja povezana z družabnimi omrežji se napolnijo, ko uporabnik doda določeno družabno omrežje v aplikacijo. Tako se na primer ob dodajanju Twitter omrežja v polje »twitter« zapiše žeton za avtentikacijo s Twitterjevim vmesnikom

API, v »twitter_profile« se zapišejo vsi podatki profila, zadnje objave uporabnika se zapišejo v »twitter_statuses«, v polje »twitter_statuses_check« pa se zapiše trenutni Unix časovni žig, ki služi za preverjanje, kdaj je uporabnik zadnjič prebral objave s Twitterja preko vmesnika API. Podobno je z ostalimi polji povezanimi z družabnimi omrežji. Za omrežje Facebook uporabljamo 6 polj, kamor se zapišejo podatki profila, sporočila in obvestila uporabnika ter Unix časovni žigi. Google+ ima poleg žetona za avtentikacijo le še dodatno polje namenjeno shranjevanju profila uporabnika. Za omrežje LinkedIn pa so namenjena 4 polja: za žeton, profil uporabnika in sporočila oziroma obvestila.

Ostalih 6 tabel je namenjenih shranjevanju povezav oziroma prijateljev uporabnikov. Tako imamo za vsako družabno omrežje svojo tabelo, kjer je shranjen identifikacijski ključ družabnega omrežja prijatelja ter njegovi podatki iz profila. Tabela »user_friends« je vmesna tabela, ki služi povezovanju uporabnikov in njihovih prijateljev (en uporabnik ima lahko več prijateljev in en prijatelj je lahko povezan z več uporabniki aplikacije). Tabela »friends« pa je namenjena povezavi profilov različnih omrežij, kjer je en zapis v tabeli sestavljen iz identifikacijskega ključa uporabnika, ki mu zapis pripada, ostala 4 polja pa predstavljajo ključne omrežij prijateljevega profila. Programske smo izdelali omejitev povezanega profila. Zapis obstaja, če sta povezani vsaj dve različni omrežji, se pravi, da ima prijatelja, ki je na primer registriran v družabnem omrežju Facebook, hkrati pa ima račun tudi v LinkedIn. Tako bosta polji »fb_id« in »linkedin_id« vsebovali ključa profilov, polji za Google+ in Twitter pa bosta vsebovali vrednosti »NULL«.

4.5. Opis rešitve

4.5.1. Registracija in prijava

Spletna aplikacija je zasnovana tako, da je za njeno uporabo potrebna prijava v sistem, s čimer se uporabnik identificira in se nato iz podatkovne baze preberejo podatki, ki so povezani s tem uporabnikom. Pred prvo prijavo je potrebno ustvariti račun, zato je bil za ta namen izdelan registracijski obrazec.

Registracija

Elektronski naslov:	
Elektronski naslov ponovno:	
Geslo:	
Geslo ponovno:	
Potrdi	

Slika 6. Forma za registracijo.

Registracija je zelo preprosta, saj sta zanjo potrebna le dva podatka: elektronski naslov in geslo. Za še hitrejšo registracijo je bila dodana JavaScript koda, ki na gumbih za pomikanje gor in dol ustvari funkciji, s katerima se pomikamo po poljih. Ob potrditvi obrazca registracije se izvede preverjanje podatkov s tehniko AJAX, pri kateri ni potrebno naložiti celotne strani, ampak se prikažejo le spremembe. Preveri se format elektronskega naslova in ali ta elektronski naslov ni bil že uporabljen pri kakšni registraciji. Seveda elektronski naslov ne sme obstajati v bazi, sicer se pojavi obvestilo »Ta elektronski naslov je že uporabljen!«. Omejitev pri geslu pa je dolžina, le-to mora biti dolgo vsaj 8 znakov. Če je preverjanje podatkov uspešno, se podatki shranijo v bazo, še prej pa se zaradi varnostnih razlogov geslu doda predpono in se ga kodira z MD5 funkcijo. Ob koncu registracije se prikaže tudi

obvestilo, ki pove, da je bila registracija uspešna, dodana pa je tudi povezava, ki preusmeri na prijavo v sistem.

Po uspešni registraciji lahko začnemo uporabljati aplikacijo. V obrazec za prijavo vpišemo elektronski naslov in geslo, ki smo ga podali ob registraciji, in potrdimo.

The image shows a login form titled "Prijava" in green text. Below the title, there are two input fields: "Elektronski naslov:" and "Geslo:". To the right of the "Geslo:" field, there are two links: "Pozabljeno geslo ?" and "Registracija". At the bottom right, there is a button labeled "Potrdi".

Slika 7. Forma za prijavo v aplikacijo.

V primeru da so podatki napačni, sistem javi napako »Napaka. Elektronski naslov in/ali geslo nista pravilna!«, če pa je kombinacija pravilna, se ustvari nova seja za uporabnika, kamor se zapiše elektronski naslov ter žetoni družabnih omrežij, ki jih je uporabnik aplikaciji že dodal. Nato pa sledi preusmeritev uporabnika na prvo stran aplikacije.

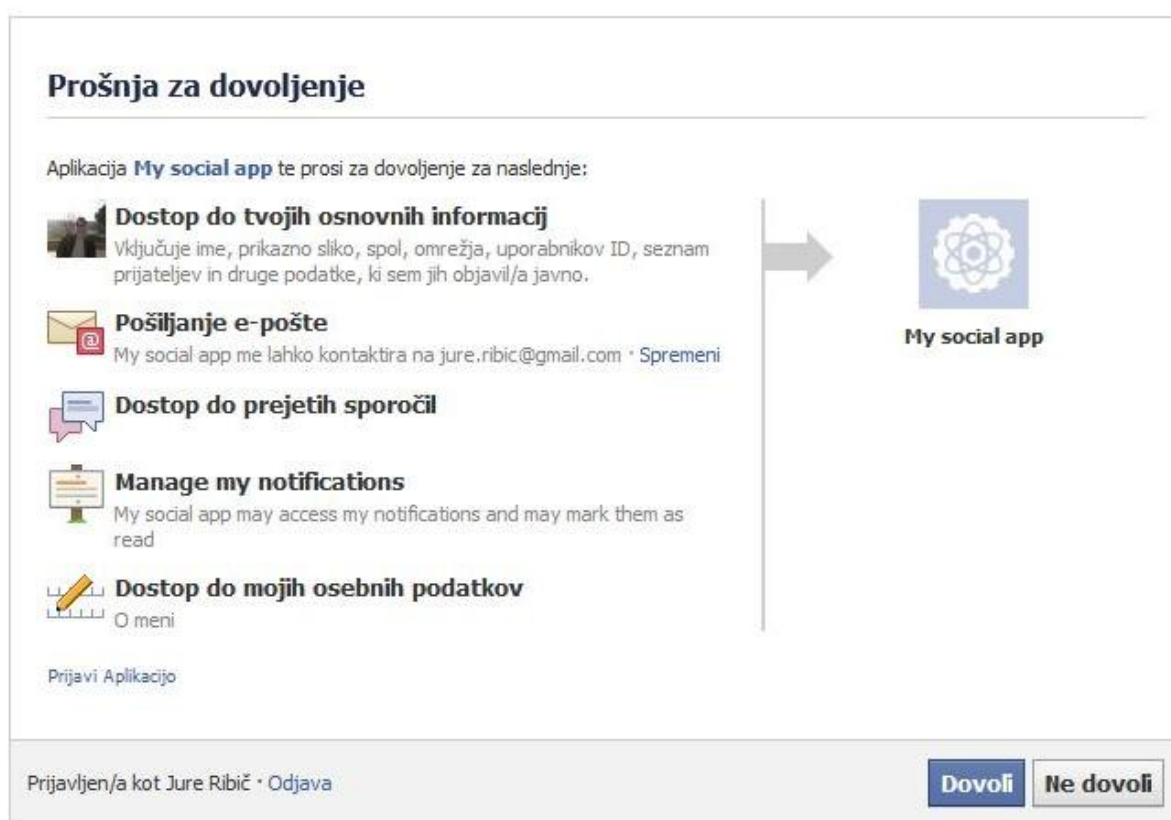
Izdelana je bila tudi funkcija za ponastavitev gesla uporabnika v primeru, da je bilo geslo pozabljeno. Uporabnik lahko zahteva novo geslo, tako da izbere povezavo »Pozabljeno geslo?«, kjer vpiše elektronski naslov, s katerim se je registriral. S tehnologijo AJAX se pokliče funkcijo, ki preveri, če elektronski naslov v podatkovni bazi obstaja. V primeru obstoja, se generira naključno 5-mestno geslo, ki ga pošljemo uporabniku na elektronski naslov. Geslu pa se, tako kot pri registraciji, doda predpono, kriptira se ga z MD5 funkcijo ter zapiše v bazo.

Ob uspešni prijavi na prvi strani opazimo zelo nazorno prikazane ikone za posamezna omrežja.



Slika 8. Ikone za dostop do podstrani družabnih omrežij.

Družabna omrežja, ki so predstavljena z netransparentnimi ikonami, so že bila aktivirana. Potek dodajanja računa družabnega omrežja v aplikacijo je zelo preprost. Uporabnik s klikom na zeleno omrežje požene funkcijo, ki izvede dodajanje njegovega družabnega računa v aplikacijo.



Slika 9. Prošnja za dovoljenje dostopa aplikacije do podatkov uporabnika pri omrežju Facebook. [41]

Pri tem je postopek za vsa omrežja zelo podoben, saj vsi vmesniki API uporabljajo OAuth protokol za avtentikacijo. Pri avtentikaciji z vmesnikom API na osnovi protokola OAuth sta obvezna dva parametra, in sicer ključ API in skrivni ključ. Dobimo ju tako, da si na straneh za razvijalce družabnega omrežja kreiramo aplikacijo. Ta korak je nujno potreben za vsakega razvijalca, ki hoče ustvariti aplikacijo v določenem družabnem omrežju, uporabniki teh aplikacij pa potem le dodelijo pravice aplikaciji, da lahko aplikacija dostopa do njihovega profila.

Company Info

* Company: Jure social connection

Account Administrators: You will be assigned as an account administrator.

Additional Administrators:

Start typing the name of a connection

Administrators appearing here will be account administrators for all applications from this company. Administrators can edit application and add/remove other administrators and developers.

Application Info

* Application Name:

* Description:

Integration URL:

Example URL where the integration will go live.

App Logo Secure URL:

URL of an 80x80 logo for your app. SSL is required now.

JavaScript API Domain:

Fully-qualified domain name of all pages that will call the JavaScript API with this key (required if using JS API).

* Application Type: Web application

If your application qualifies as more than one type, create a new application for each one.

* Application Use: Select One...

What best describes your application?

* Live Status: Development

While in development, your network updates will only go to the developers you choose. When live, they will go to your connections.

Application Developers:

Start typing the name of a connection

Network updates you send will appear only for developers you list.

☒ Include yourself as a developer for this application

Slika 10. Registracija aplikacije na družabnem omrežju linkedin.com. [35]

Pri vseh registracijah aplikacij družabnih omrežij je treba podati ime aplikacije, ki ne sme vsebovati imena družabnega omrežja na katerem kreiramo aplikacijo, podati je potrebno tudi kratek opis in spletni naslov, na katerem se bo aplikacija nahajala. Pri nekaterih registracijah je potrebno podati še naslov, kamor naj družabno omrežje preusmeri uporabnika po izvedeni avtentikaciji, pri drugih pa je to rešeno znotraj knjižnice, ki komunicira z vmesnikom, kjer povezavo za preusmeritev nazaj na stran podamo kot parameter. Google ima za registriranje

aplikacij eno stran, ki pokriva vse njegove storitve. Tako lahko preko ene strani spremljaš vse statute njihovih storitev, spreminjaš nastavitve, itd.

	Google Cloud Storage		☐ OFF	Pricing
	Google Maps API v2		☐ OFF	Courtesy limit: 25,000 queries/day • Pricing
	Google Maps API v3		☐ OFF	Courtesy limit: 25,000 queries/day • Pricing
	Google+ API		☒ ON	Courtesy limit: 1,000 queries/day
	Google+ Hangouts API		☐ OFF	
	Groups Settings API		Request access...	Courtesy limit: 100,000 queries/day
	Identity Toolkit API		☐ OFF	Courtesy limit: 1,000,000 queries/day

Slika 11. Google konzola za pregled vseh Googlovih storitev. [36]

Twitter poleg standardnih polj pri registraciji aplikacije dodaja še polje, ki opisuje nivoje dostopa aplikacij. Ločujejo tri nivoje: samo branje, branje in pisanje ter vse pravice, kjer aplikacija lahko bere tudi zasebna sporočila. LinkedIn in Facebook pa pri registraciji dodajata še vrsto integracije registrirane aplikacije. Torej ali gre za mobilno aplikacijo, namizno aplikacijo, aplikacijo za iOS ali Android, ali je to spletna aplikacija ali le aplikacija, ki bo del družabnega omrežja. Ko registracijo aplikacij zaključimo, nam pri vseh družabnih omrežjih v podobni obliki podajo informacije o možnih dostopih. Tako pridobimo potrebne podatke, ki jih bomo uporabljali za avtentikacijo na osnovi OAuth protokola.

Za avtentikacijo smo izdelali dve knjižnici. Prva je namenjena verziji 1.0 protokola OAuth, ki ga uporabljata LinkedIn in Twitter, druga pa verziji 2.0 in je torej namenjena omrežjema Facebook in Google+.

Ko je uporabnik avtentificiran lahko začne uporabljati funkcionalnosti vmesnika API določenega družabnega omrežja. Za vsako omrežje smo izdelali knjižnico, ki komunicira z vmesnikom API.

Facebookov vmesnik ima največ funkcionalnosti, hkrati pa je tudi najbolj zapleten, ker ima še tako imenovane »razširjene pravice«, ki pokrivajo različne sklope podatkov. V knjižnici, ki smo jo razvili, so podprte naslednje funkcionalnosti:

- Pridobitev podatkov profila uporabnika.
- Pridobitev liste prijateljev.
- Pridobitev podatkov profila prijatelja.
- Pridobitev obvestil.
- Pridobitev zasebnih sporočil.
- Pridobitev prošenj.
- Objava novega statusa.

Za zgornje funkcionalnosti je bilo potrebno uporabiti naslednje razširjene pravice.

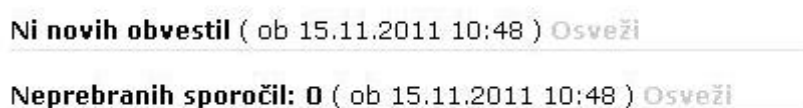
Tip razširjene pravice	Opis
Read_mailbox	Omogoča branje uporabnikovega nabiralnika sporočil.
Manage_notifications	Omogoča aplikaciji branje obvestil in označevanje, da so sporočila prebrana.
Read_request	Omogoča branje novih prošenj za prijateljstvo.
Publish_stream	Omogoča aplikaciji, da ustvarja nove statuse na zidu, komentira ali uporabi funkcijo »všeč mi je«.
Read_stream	Omogoča branje vseh objav uporabnika, ki jih je objavil na svojem zidu.

Tabela 2. Pregled pravic za dostop do podatkov pri Facebook vmesniku API. [33]

Knjižnica, ki skrbi za komunikacijo z Googlovim vmesnikom API, trenutno podpira le funkcionalnost pridobivanja podatkov javnih profilov uporabnikov. Hkrati pa ima tudi najbolj napreden sistem za avtentikacijo, saj ima poleg navadnega žetona, ki se uporablja za dostop do vmesnika, še žeton za osveževanje in parameter, ki nam pove, koliko časa bo žeton veljaven. Po preteku žetona izvedemo funkcijo, ki pošlje vmesniku žeton za osveževanje, kot odgovor dobimo nov žeton, ki ga potem uporabljamo naprej za dostop do podatkov. Knjižnica, zadolžena za LinkedIn vmesnik, lahko pridobi podatke profila uporabnika, vse povezave s katerimi je profil povezan in podroben profil teh povezav. Omogoča tudi objavljanje statusov naravnost z naše aplikacije. Knjižnica za klice Twitterjevega vmesnika omogoča pridobivanje zadnjih objav uporabnika, listo prijateljev, listo sledilcev, podatkov

profila uporabnika in njegovih prijateljev in sledilcev, neposredno preko aplikacije lahko objavimo tudi nov »tweet«.

Vsi pridobljeni podatki se shranijo v bazo. Razloga za to pa sta dva. Prvi razlog je zmanjšanje prometa, saj ni potrebe, da bi ob zelo pogostem osveževanju strani aplikacija znova iz znova pridobivala podatke profila uporabnika oziroma sezname prijateljev. Drugi pa je povezan z omejitvami števila klicev vmesnikov API. Dodana pa je seveda tudi funkcionalnost, ki ponovno pobere podatke preko vmesnika. Tako se na primer zasebna sporočila in obvestila iz Facebooka preverijo na vsako uro, oziroma jih lahko s klikom na povezavo sami osvežimo, prav tako lahko osvežimo liste prijateljev, sledilcev oziroma povezav in lastnih profilov.



Slika 12. Povezave za osveževanje obvestil in sporočil.

Vse objekte, ki jih vrne vmesnik API, razen spiska prijateljev oziroma povezav, serializiramo, kar pomeni, da objekte pretvorimo v tok podatkov, ki ga potem shranimo v bazo. S tem dosežemo, da nam ne glede na število podatkov v profilu, strukture tabel ni potrebno spreminjati, ker se vse zapiše v eno samo polje. Ob branju podatkov iz baze se ta tok podatkov rekonstruira nazaj v prvotni objekt.

Ko uporabnik doda omrežja, se mu odpre nova stran, ki prikaže povzetek podatkov, ki so bili pobrani z določenega omrežja. Tako na strani namenjeni omrežju Facebook vidimo trenutno sliko profila, osnovne podatke (ime, priimek, datum rojstva, spol in kraj), pod njim pa zadnja neprebrana obvestila in sporočila ter podatek, kdaj je bilo zadnje preverjanje.

Facebook

Profil



Pojdi na fb

Ime:
Rojstni dan:
Spol:
Kraj:

Jure
06/06/1987
Moški
Ljubljana

Ni novih obvestil (ob 10.02.2012 18:18) [Osveži](#)

Neprebranih sporočil: 0 (ob 10.02.2012 18:18) [Osveži](#)

Objavi nov status:

St. znakov: 0

Objavi

Prijatelji

Osveži povezave



Ales
02/07/1983



Maja
12/05/1985



Viktorija
02/06



Eva
07/04



Grega
07/01



Gašper

Slika 13. Podstran namenjena Facebooku.

Na desni strani je prikazana lista prijateljev. Vsak zapis je sestavljen iz slike prijatelja, imena in priimka, rojstnega datuma in kraja, od koder prihaja. Dodana pa je tudi direktna povezava do njegovega profila. Če rojstnega datuma in kraja prijatelj ni objavil v svojem profilu, ga posledično tudi tu ni. Za boljšo preglednost se v začetku prikaže le prvih 10 prijateljev, za več prikazov pa kliknemo na namenski gumb, ki prikaže naslednjih 10 zapisov.

```

/**
 * Funkcija za prikaz dodatnega števila prijateljev
 */
$('#more-friends').click( function() {

    $('.page').each(function(index) {

        if( $(this).css('display') == 'none' ){
            $(this).css('display','block');

            if( $('.page').length == ( index +1 ) )
                $('#more-friends').remove();

            return false;
        }
    });

});

```

Slika 14. JavaScript koda za prikaz dodatnih prijateljev pri pregledu.

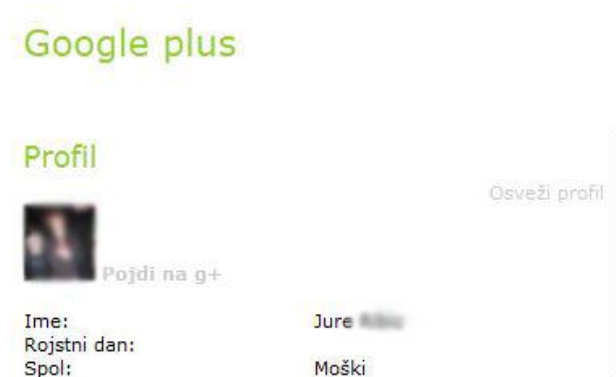
Zapisi prijateljev so razdeljeni na strani. Vsaka stran vsebuje 10 zapisov. Tako se najprej prikaže prva stran, vse ostale pa so skrite v ozadju. Ob kliku na gumb »Več« pa se prikaže naslednja stran. Ko se uporabniku prikažejo vse strani, gumb izgine, tako uporabnika obvestimo, da je dosegel zadnjo stran. Pri omrežju Facebook je pomembno tudi to, da se je vedno znova potrebno prijaviti, ko se pridobiva podatke iz omrežja, če seja poteče. Tako smo izdelali funkcijo, ki ob ponovni prijavi preveri, če gre za istega uporabnika. Preverja torej id profila uporabnika, ki je zapisan v bazi s ključem, ki ga vrne vmesnik Facebook. V primeru da se ključa razlikujeta, nas o tem obvesti in poda dve možnosti. Če račun, s katerim smo prijavljeni v Facebook račun, ni naš, se odjavimo in ponovno prijavimo z našim uporabniškim imenom in geslom. V nasprotnem primeru, če smo torej spremenili Facebook račun, pa to potrdimo s klikom na gumb »Potrdi«. Izvede se funkcija, ki v bazi posodobi profil uporabnika, nastavi nov id ter doda morebitne nove prijatelje, stare pa obdrži.

Uporabnik s katerim ste trenutno prijavljeni na družabnem omrežju Facebook ni enak tistemu katerega ste dodali v aplikacijo.
Potrdite spremembo ali se odjavite in prijavite z drugim uporabniškim imenom.

Potrdi ali Odjavi iz facebooka in Osveži

Slika 15. Prikaz obvestila v primeru, da se nov profil razlikuje od tistega, s katerim se je uporabnik najprej predstavil.

Google+ podstran je med vsemi podstranmi namenjenimi družabnim omrežjem najbolj skopa, saj so prikazani le osnovni podatki uporabnika z njegovo sliko profila in povezavo.



Slika 16. Google+ podstran.

Podstran za Twitter prav tako prikazuje osnovne podatke profila (ime, število sledilcev, datum ustvarjenega računa), spisek oseb, ki uporabniku sledijo in osebe, ki jim sledi uporabnik. Spisek je sestavljen podobno kot pri omrežju Facebook, le da je tu še malo manj podatkov, saj Twitter sam po sebi ne omogoča vpisovanje podatkov, kot so datum rojstva, od kod prihajamo, itd. Posamezen zapis je sestavljen iz slike profila, imena in povezave na profil Twitterja.

Twitter

Št. dostopov: 350/348

Profil



Pojdi na tweeter

Ime: **Jure (@jures)**
 Št. sledilcev: **3**
 Račun ustvarjen: **28.08.2011 17:14:54**
 Št. statusov: **9**

Objavi nov status:

St. znakov: 0

Omejitev je 140 znakov.

Objavi

Osebe s katerimi si povezan

Osveži povezave



Janez ~~Novakovič~~
 Povezava do profila



john ~~Smith~~
 Povezava do profila

Zadnji twitti

Zadnjič osveženo: 10.02.2012 18:10

Osveži status

Objavil uporabnik: **Jure** (05.12.2011 10:27:53)

~~Tweet...~~

Objavil uporabnik: **Jure** (01.12.2011 10:26:22)

~~Tweet...~~

Objavil uporabnik: **Jure** (25.11.2011 14:08:12)

~~Spet se je pojavila priložnost za to čudovito stvar... Tweet...~~

Objavil uporabnik: **Jure** (25.11.2011 13:50:42)

~~Tweet...~~

Objavil uporabnik: **Jure** (25.11.2011 13:26:22)

~~Spet se je pojavila priložnost za to čudovito stvar... Tweet...~~

Objavil uporabnik: **Jure** (11.10.2011 12:51:39)

~~Neke vrste followerjev~~

Objavil uporabnik: **Jure** (28.09.2011 15:13:30)

~~Moja Twitter računa zdaj ne bo več...~~

Objavil uporabnik: **Jure** (28.09.2011 15:09:27)

~~Diplomska naloga napreduje!!!~~

Objavil uporabnik: **Jure** (08.09.2011 22:57:13)

~~Twitter je priložnost za to...~~

Objavil uporabnik: **Jure** (08.09.2011 22:54:35)

~~Thu, 08 Sep 11 22:54:35 +0000~~

Slika 17. Podstran za družabno omrežje Twitter.

Možno pa je dodajanje novih objav naravnost z naše aplikacije. Preprosto se v polje vpiše željeno besedilo in s pritiskom na gumb »Objavi« se preko vmesnika API pošlje objava na Twitter. Dodana je JavaScript funkcija, ki deluje kot števec znakov, da ima uporabnik predstavlo, koliko je že napisal, saj so »tweeti« omejeni na dolžino 140-ih znakov. V primeru da uporabnik preseže omejitev, PHP poskrbi za krajšanje teksta, preden ga pošlje vmesniku API, prav tako poskrbi, da odstrani morebitne HTML značke.

```
// pridobi trenutno število znakov
$('#counter').html( 'St. znakov: ' + $('#status').val().length);

$('#status').keyup(function()
{
    // nastavi novo število znakov
    $('#counter').html( 'St. znakov: ' + $(this).val().length);
});
```

Slika 18. JavaScript funkcija, ki je zadolžena za štetje in prikaz znakov.

Desni del podstrani Twitterja je namenjen prikazu zadnjih objav z datumom, kdaj je bila osvežitev zadnjič izvedena. Avtomatsko se preverja objave vsako uro, uporabnik pa lahko uporabi gumb »osveži statute«, ki izvede funkcijo ponovne pridobitve statusa preko vmesnika API. Pri klicu vmesnika API je treba upoštevati, da je v eni uri možnih 350 dostopov.

LinkedIn podstran prikazuje podatke profila (ime, priimek, datum rojstva, kraj), na desni strani pa so prikazane povezave uporabnika z istim načinom prikaza kot pri podstrani omrežja Facebook. Dodana sta še gumba za osveževanje profila in povezav, ki sta namenjena ročnemu osveževanju vsebin. Na levi strani pod podatki profila je obrazec za objavo statusov na družabnem omrežju. Prav tako ga sestavlja JavaScript funkcija, ki šteje vpisane znake, na strani strežnika pa je še dodatno poskrbljeno, da se omeji dolžino na 700 znakov ter da se sestavi XML, ki se potem pošlje vmesniku LinkedIn. XML mora obvezno vsebovati polje »comment«, kjer je zapisan nov status, ter polje »visibility«, ki pove, kdo vse lahko vidi objavljen status.

LinkedIn

Profil

Pojdi na linkedin

Ime:

Jure [redacted]
Student at Univerza v Ljubljani

Rojstni datum:

[redacted]

Objavi nov status:

St. znakov: 0

Omejitev je 700 znakov.

Objavi

Osveži profil

Prijatelji

Osveži povezave



Marko Jurešič

[redacted]

Slika 19. Podstran namenjena omrežju LinkedIn.

Ob vsaki osvežitvi strani se v ozadju izvede javascript funkcija, ki s tehnologijo AJAX preverja, če so družabna omrežja dosegljiva.

```

$('#networks_buttons').hide();

$.ajax({
    url: '/login/set_access',

    complete: function(){
        $('#networks_buttons').show();
    }
});

```

Slika 20. Koda, ki pokliče AJAX funkcijo za preverjanje dosegljivosti omrežij.

Preverjanje se izvede tako, da se za vsako omrežje poskuša naložiti podstran, ki je namenjena za določeno omrežje.

```

/**
 * Funkcija, ki preveri dosegljivost povezave
 * @param string $url
 * @return boolean
 */
protected function check_url($url) {

    $ch = curl_init();
    curl_setopt($ch, CURLOPT_USERAGENT,
        "Mozilla/5.0 (Windows; U; Windows NT 5.1; rv:1.7.3) Gecko/20041001 Firefox/0.10.1" );
    curl_setopt($ch, CURLOPT_URL, $url);
    curl_setopt($ch, CURLOPT_HEADER, true);
    curl_setopt($ch, CURLOPT_NOBODY, true);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    $data = curl_exec($ch);
    curl_close($ch);
    $tmp = explode(' ', $data);

    return $tmp[1] == '200' ? TRUE : FALSE;
}

```

Slika 21. Koda, ki se izvede na strani strežnika in je zadolžena za preverjanje dosegljivosti povezave.

Preverjanje se izvede s pomočjo knjižnice cURL. Ta izvede klic povezave, ki jo dobi kot parameter, in vrne niz podatkov o glavi, ki jo dobi kot odgovor na klic.

V primeru, da kot rezultat za omrežje dobimo HTML kodo 200, to pomeni, da je omrežje dosegljivo. Posledično se izvede še ena funkcija, ki postavi sejno spremenljivko, ki je namenjena za preverjanje dosegljivosti omrežja na vrednost »TRUE«. Privzeto so vse spremenljivke, ki so namenjene za preverjanje dosegljivosti postavljene na »FALSE«, tako da če testiranje dosegljivosti ne da odgovora 200, spremenljivke ohranijo svoje privzete vrednosti. Če je uporabnik omrežja že dodal v aplikacijo, vendar v tistem trenutku omrežje ni

dosegljivo, se prikažejo zadnji shranjeni podatki, povezav za osveževanje ni, prav tako pa so odstranjeni vsi obrazci namenjeni za objavljanje.

4.5.2. Prijatelji

Podstran »prijatelji« je namenjena prikazu vseh uporabnikovih prijateljev, ki so zapisani v bazi. Najprej so prikazani vsi združeni zapisi, pod njimi pa sledijo nepovezani zapisi vseh dodanih družabnih omrežij. Pri vsakem zapisu je zelo nazorno prikazano, katere profile zajema, oziroma če ni združen zapis, kateremu družabnemu omrežju pripada. Med podatki profilov povezanih zapisov se premikamo tako, da kliknemo na ikono družabnega omrežja. Izbrano omrežje je predstavljeno z netransparentno ikono. Spreminjanje prikazov podatkov je narejeno z JavaScriptom. Ob spremembi se na desni strani spremenijo podatki, na levi pa se prikaže slika profila izbranega družabnega omrežja.

```

$( '.buttons a' ).click( function() {

    //odstrani stil aktivnemu gumbu
    $(this).siblings().removeClass('active');

    //dodaj stil izbranemu gumbu
    $(this).addClass('active');

    //katero omrežje bomo aktivirali
    predhodniki = $(this).prevAll().length

    //pridobi podatke vseh profilov
    datas = $(this).parent().next().children();

    //pridobi vse slike
    images = $(this).parent().prev().children();

    //skrij profile in slike vseh omrežij
    $(datas).addClass('none');
    $(images).addClass('none');

    //prikaži profil izbranega omrežja
    $(datas[predhodniki]).removeClass('none');
    $(images[predhodniki]).removeClass('none');

});

```

Slika 22. Javascript koda, ki je odgovorna za preklapljanje med profili povezanega profila.

Za lažje iskanje profilov smo izdelali še JavaScript funkcijo, ki izvede filtriranje glede na vnesen iskani niz. Pri vsakem profilu smo dodali skrito vnosno v polje, v katerem sta zapisana ime in priimek prijatelja, ta niz je pred izpisom spremenjen v male črke. Pri povezanih

profilih je za vsako družabno omrežje, ki je v povezanem profilu, eno skrito vnosno polje. Tako je iskanje možno po vseh variantah imena in priimka, s katerimi se je uporabnik predstavil v družabnih omrežjih. Ko uporabnik začne vpisovati niz znakov v vnosno polje, se ob vsakem pritisku na tipkovnico izvede filter, ki prikaže le ujemajoče se profile. Vsi ostali profili pa so, s pomočjo tehnike CSS, skriti v ozadju.

```

/*
 * Funkcija za filtriranje prijateljev
 */
$( 'input[name="search_key"]' ).keyup( function() {

    //ce je dolžina niza večja od 0
    if( $(this).val( ).length > 0 ){

        //pridobi iskani niz in ga spremeni v male črke
        var my_search_key = $(this).val( ).toLowerCase();

        //skrijemo vse zapise
        $( "input:hidden" ).parent().hide();

        //prikažemo ujemajoče zapise
        $( "input:hidden[value*='"+ my_search_key +"']" ).parent().show();
    }
});

```

Slika 23. JavaScript koda za filtriranje zadetkov pri iskanju prijateljev.

Za primer če želimo ponastaviti iskani niz, smo desno od vnosnega polja dodali gumb »Ponastavi«. Ob kliku na ta gumb se ponastavi filtriranje, pobriše se vpisan iskani niz in ponovno se prikažejo vsi profili.

Za avtomatsko združevanje profilov je bila izdelana posebna funkcija. Poženemo jo tako, da v podmeniju izberemo »Avtomatsko iskanje ujemanj«. To deluje tako, da se najprej izvede preverjanje že obstoječih ujemanj. Za vsak profil uporabnikovega prijatelja se v bazi preveri ali je bil že uporabljen. Če se najde ujemanje, se preveri, s katerim profilom je povezan in v katero družabno omrežje spada. V polje »match« ujemajočega se profila se doda identifikacijski ključ profila, s katerim se ujema, še prej se preveri, če mogoče ta ključ ni bil že dodan, saj se lahko v bazi neka povezava profilov pojavi večkrat. Na koncu se preveri, če je med povezanimi profili mogoče takšen profil, ki ga uporabnik še nima med prijatelji. Morebiten nov profil se doda med uporabnikove prijatelje, hkrati pa se iz baze prebere še podrobnosti tega profila, ki bodo uporabljene kasneje pri izpisu. Po končanem preverjanju morebitnih že obstoječih povezav sledi pripravljanje podatkov na morebitno dodatno združevanje profilov, ki še niso bili povezani. V tej funkciji se za vsako omrežje izdela niz

objektov (ang. array). Hkrati pa se izloča profile, ki so že bili dodani v združen profil. Tako tukaj preprečimo možnost, da bi bil en profil dodan v dva različna povezana profila.

Vsak objekt niza je sestavljen iz naslednjih polj :

Polje	Opis
Id	Identifikacija profila
Network	Ime omrežja, ki mu profil pripada.
First	Ime
Last name	Priimek
Gender	Spol
Birthday	Datum rojstva
Home	Domači kraj
Matches	Niz podatkov, ki vsebuje identifikacijske ključe profilov, s katerimi je ta profil povezan.

Tabela 3. Polja, ki jih vsebuje niz podatkov, ki predstavlja prijatelje.

Na vseh črkovnih poljih se izvede še funkcija, ki odstrani podvojene, začetne in končne presledke, nato se spremeni vse znake v male črke, nazadnje se vse šumnike spremeni v sičnike. Razlog za slednje je ta, da nekateri v profilih uporabijo enkrat šumnike drugič sičnike, saj kakšno omrežje nima dobre podpore za šumnike in se jih zato izogibajo. Prav tako pa se bo poenostavilo primerjanje zapisov pri iskanju ujemanj. Pri datumu rojstva se poenoti format, dobimo niz sestavljen iz dveh oziroma treh polj, odvisno od tega, ali je objavljeno tudi leto rojstva ali samo dan in mesec, obstaja pa seveda tudi možnost, da polje ostane prazno. Ko se podatki pripravljene, lahko začnemo s primerjanjem profilov. Primerjava se izvede med vsemi omrežji. V dveh ugnezenih zankah se pridobiva podatke dveh družabnih omrežij, katerih podatki so podani kot parametra funkciji za iskanje ujemanj. V funkciji se primerja med seboj vse profile obeh podanih omrežij. Ujemanje je deljeno na dva dela. Prvi del ja namenjen primerjavam, kjer je eno izmed omrežij Twitter, drugo pa je za vse ostale primerjave. To deljenje smo naredili zato, ker pri družabnem omrežju Twitter ni na razpolago toliko podatkov, ki bi nam pomagali pri primerjavi. Tako se pri slednjem ujemanje izvede le na poljih »name«, ki naj bi bilo namenjeno imenu uporabnika računa Twitter, in »screen_name«, ki je vzdevek uporabnika. Vsa ostala omrežja pa za ta namen uporabljajo polji ime in priimek. Izdelali smo šest možnih kombinacij primerjanj v povezavi z družabnim omrežjem Twitter:

- Ali sta ime in priimek profila prvega omrežja enaka imenu ali vzdevku profila uporabnika Twitterja? (Ime in priimek združimo s presledkom pred primerjavo.)
- Ali je ime profila prvega omrežja enako imenu ali vzdevku profila uporabnika Twitterja?
- Ali je priimek profila prvega omrežja enak imenu ali vzdevku profila uporabnika Twitterja?

Pri iskanju ujemanj ostalih omrežij smo imeli na razpolago več parametrov za primerjanje. Poleg imena in priimka smo uporabili še domači kraj in datum rojstva. Tako se primerjanje najprej izvede med imenom in priimkom obeh profilov, če ni ujemanj, se preveri, če za oba profila obstaja podatek o datumu rojstva. Če podatek obstaja, se preventivno preveri domači kraj profilov, če je to mogoče, saj ob morebitnem različnem kraju lahko izločimo ujemanje. V primeru, da se domača kraja ujemata, oziroma da teh podatkov ni, pa se izvede primerjanje rojstnega datuma.

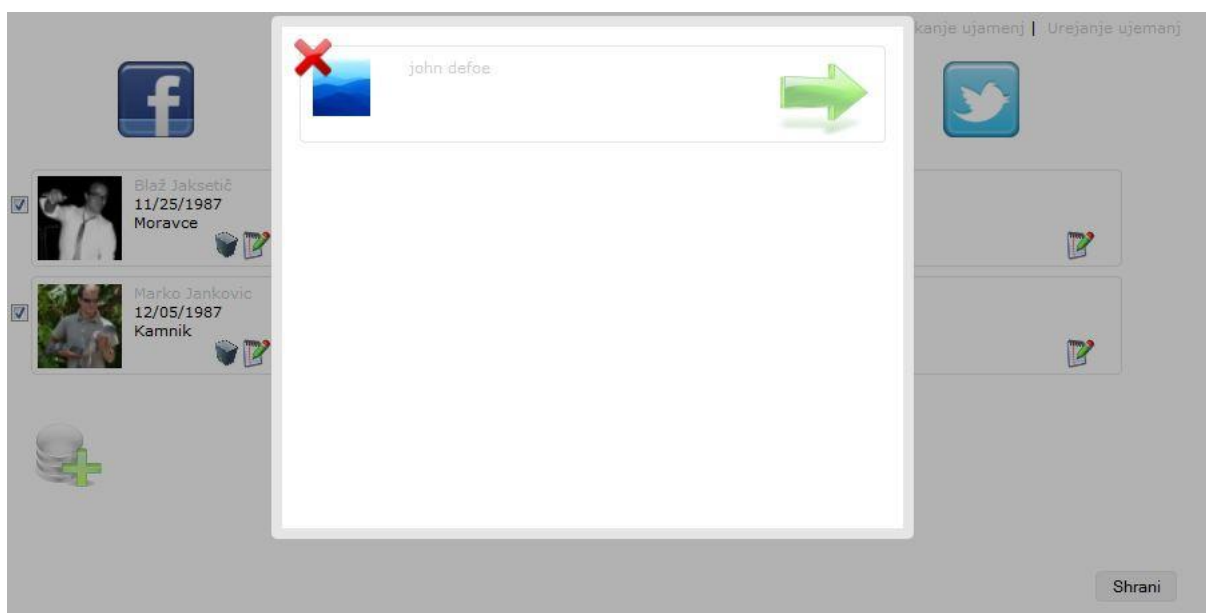
Ob najdenem ujemanju se v oba objekta, ki opisujeta profila, ki ju primerjamo, v polje »match« doda identifikacijski ključ ujemajočega se profila in omrežje, ki mu ta ključ pripada. To nam je v pomoč kasneje, pri izpisu najdenih povezanih omrežij. Ko se algoritem za preverjanje ujemanj zaključi, pa je na vrsti še prikaz rezultatov. Pri prikazu podatkov je potrebno preverjati le morebitna večkratna ujemanja, torej to, če je bilo mogoče določen profil družabnega omrežja povezati z več različnimi profili drugega družabnega omrežja na podlagi parametrov, ki smo jih uporabljali za primerjavo. Takšne profile izločimo in niso prikazani med rezultati. Primer rezultata je prikazan na sliki.



Slika 24. Prikaz rezultata najdenega povezanega profila.

V zgornji vrstici so s simboli prikazana družabna omrežja, ki predstavljajo profile stolpcev. Če zapis pod določenim družabnim omrežjem manjka, to pomeni, da ni bilo najdenega ujemanja za tisto omrežje. Ikona, ki predstavlja koš, je namenjena odstranjevanju profila v primeru, da avtomatsko ujemanje ni bilo pravilno. Klik na brisanje izvede JavaScript funkcijo, ki je zadolžena za odstranjevanje profila, tako se dinamično brez nepotrebnega osveževanja strani odstrani želeni profil. Vsak profil ima še povezavo do družabnega omrežja, kjer se nahaja, tako da uporabnik lahko še dodatno preveri, če je prišlo do napak. V primeru da

kakšnega izmed povezanih profilov ne želimo shraniti, smo na začetek vsakega dodali potrditveno polje, ki označuje kateri povezan profil se bo shranil in kateri ne. Če polje ni izbrano, se tisti zapis ne bo shranil v bazo. Ko smo s pregledovanjem rezultatov avtomatskega združevanja zaključili, pritisnemo gumb shrani. Izvede se funkcija, ki preveri, če sta pri vsakem združenem zapisu še vedno vsaj dve različni omrežji, saj bi uporabnik lahko med pregledovanjem odstranil profil katerega izmed omrežij. Če je pogoj zadoščen, se povezan profil shrani v podatkovno bazo. Po zaključku se pojavi še obvestilo, ki nam pove, koliko povezanih profilov je bilo shranjenih. Možno je tudi naknadno urejati oziroma ročno dodajati povezane profile. To storimo tako, da v meniju izberemo »Urejanje ujemanj«. Prikažejo se vsi povezani profili v skoraj identični obliki kot pri avtomatskem iskanju ujemanj. Dodani sta ikona za spreminjanje profila znotraj ujemanja ter ikona za dodajanje novega ujemanja. Prva deluje tako, da se ob kliku nanjo odpre pojavno okno (ang. popup) z vsemi profili izbranega družabnega omrežja, ki še niso bili uporabljeni v povezanih profilih. JavaScript poskrbi, da se dinamično prikaže okno, s tehnologijo AJAX pridobimo vse profile, na koncu pa se izvede še JavaScript funkcija, ki izloči morebitne že uporabljene profile pri urejanju, saj jih na strani streznika nismo mogli izločiti, če še ni bilo izvedeno shranjevanje. Oknu smo dodali tudi dogodek (ang. event), ki mu dinamično spreminja pozicijo vedno, ko se pomaknemo po strani navzdol oziroma navzgor.



Slika 25. Pojavno okno v katerem menjamo oziroma dodajamo profil v povezan profil.

V pojavnem oknu potem izberemo ustrezen nov profil in s klikom na gumb se zamenja oziroma nastavi nov profil. Pojavno okno se avtomatsko zapre in lahko nadaljujemo s pregledovanjem oziroma urejanjem povezanih profilov. V primeru da smo pojavno okno odprli, vendar profila ne želimo zamenjati oziroma dodati novega, lahko v desnem zgornjem

kotu pritisnemo na »rdeč x«. Ta izvede klic JavaScript funkcije, ki poskrbi, da se pojavno okno zapre in da ni bila narejena nobena sprememba. Drugi gumb, ki se nahaja levo spodaj in je namenjen ročnemu dodajanju povezanih profilov, deluje tako, da se ob kliku nanj s pomočjo JavaScripta kreira novo polje za vsako družabno omrežje, na vsakem polju pa se pojavi ikona za dodajanje profila. Uporabnik potem izbere želen profil za posamezno družabno omrežje na isti način, kot poteka spreminjanje profila. Če uporabnik v celoti želi odstraniti povezan profil, enostavno odstrani kljukico v izbirnem polju, ki se nahaja na začetku povezanega profila. Ko je uporabnik zaključil z urejanjem povezanih profilov, pritisne gumb »Shrani«. Izvede se funkcija preverjanja in shranjevanje v bazo. Pri preverjanju za vsak povezan profil preverimo, če je izbirno polje izbrano in če vsebuje vsaj dva profila družabnih omrežij. Če sta izpolnjena oba pogoja, se nov profil shrani v bazo. Po končanem shranjevanju dobimo še obvestilo o številu shranjenih povezanih profilov, nato pa se lahko vrnemo nazaj na pregled zapisov.

4.5.3. Prva stran

Prva stran je namenjena povzetku uporabnika. Omenili smo že ikone, ki prikazujejo razpoložljiva družabna omrežja, s transparentnostjo le-teh smo ponazorili, ali je določeno omrežje uporabnik že dodal k aplikaciji ali ne. Pod ikonami pa je postavljena povezava na pregled prijateljev. Na levi strani so prikazani osnovni podatki profila posameznega družabnega omrežja ter število vseh povezav, ki jih je zabeležila aplikacija. Na desni strani je postavljen obrazec, ki omogoča objavljanje statusa v vseh dodanih omrežjih hkrati. Pri dostopanju do prve strani so ikone in obrazec najprej skriti, to je posledica AJAX tehnologije, ki izvede preverjanje dosegljivosti družabnih omrežij. V primeru, da kakšno družabno omrežje ni dosegljivo, se ne prikaže kot opcija pri obrazcu za objavo statusa.

The image shows a web interface for posting a status. At the top, it says "Objavi nov status:". Below this, there are three checkboxes for social media networks: "Facebook" (checked), "LinkedIn" (checked), and "Twitter" (checked). To the right of these checkboxes, it says "Št. znakov: 60". Below the checkboxes is a large text area with the placeholder text "Nov status, ki se bo pojavil na vseh družabnih omrežjih ...". At the bottom left of the text area is a button labeled "Objavi".

Slika 26. Vnosno polje za objavo statusa v več družabnih omrežjih hkrati.

Vsako družabno omrežje je predstavljeno s potrditvenim poljem, tako izberemo omrežja, v katerih želimo, da se nov status objavi, v tekstovno polje vpišemo želeni status in pritisnemo gumb »objavi«. Izvede se funkcija, kjer se najprej v sejno spremenljivko shrani sporočilo in izbrana družabna omrežja, kjer bo status objavljen. Nato pa se vzame prvo izbrano omrežje, se ga odstrani iz sejne spremenljivke in preusmeri na povezavo, ki je zadolžena za objavo statusa. Tako se izvede ista funkcija za objavo statusa, kot če bi šli na podstran nekega družabnega omrežja in objavili status tam. Razlika je samo v tem, da se tam prebere vsebino sporočila iz POST (ena izmed mnogih metod z zahtevki, ki jih podpira http protokol), v primeru, da se objavlja statuse s prve strani, pa se vsebino pridobi iz sejne spremenljivke. Ko je sporočilo na določeno omrežje poslano, se izvede preusmeritev nazaj na glavno funkcijo. Tam se preveri, če sejna spremenljivka vsebuje sporočilo in če je bil status objavljen že na vseh omrežjih. Če ni bil, se spet izvede preusmeritev na funkcijo, ki je zadolžena za objavo statusov v tistem družabnem omrežju, če pa je bilo sporočilo objavljeno na vseh omrežjih, se odstrani sejna spremenljivka, ki so bile pri tej funkciji uporabljene, ter se preusmeri nazaj na prvo stran, kjer se prikaže tudi obvestilo o uspešnosti objave.

5. Ugotovitve in možnosti za razširitve

Metode ter pristope smo predstavili v predhodnih poglavjih, v tem poglavju pa bomo predstavili rezultate in ugotovitve, do katerih smo prišli med razvojem aplikacije.

Uspelo nam je sestaviti aplikacijo, ki povezuje različna družabna omrežja in prikazuje podatke na enem mestu. Med razvojem aplikacije smo bili pozorni na to, da bo aplikacija zgrajena tako, da jo bo mogoče na enostaven način razširiti. Na ta način bi brez večjih težav lahko naknadno dodali še kakšno dodatno družabno omrežje v sklop teh štirih, s katerimi trenutno aplikacija operira. Lahko bi se spustili tudi na nivo blogov, kjer bi uporabnik neposredno z aplikacije lahko napisal novo objavo. Zelo zanimiv del aplikacije pa je algoritem za povezovanje profilov prijateljev različnih omrežij, ki avtomatsko povezuje profile na podlagi štirih parametrov: ime, priimek, datum rojstva in domači kraj. Takšno število parametrov se je izkazalo za zanesljivo, saj nismo zasledili nobene napačne povezave profilov. Pojavili pa so se profili, ki jih algoritem ni znal povezati, predvsem je tu družabno omrežje Twitter, ki ne razpolaga z večino teh parametrov, ki so uporabljeni v algoritmu. Tu smo uporabili polje »vzdevek« za primerjavo s polji »ime« in »priimek«. Učinkovitost algoritma smo izboljšali z uporabo že povezanih profilov, ki so shranjeni v bazi, tako se pri novem iskanju ujemanj uporabi shranjene zapise. S tem smo pridobili na času ter avtomatsko povezali tiste zapise, ki so jih uporabniki ročno dodali, ker jih algoritem ni znal povezati. S tem pa smo dosegli še več, ker algoritem uporablja shranjene zapise, se ves čas izpopolnjuje. Zanimiva sta dva scenarija: pri prvem uporabnik doda svoje stike dveh družabnih omrežij, ko se aktivira algoritem za povezovanje profilov, ugotovi, da pri enem ali več prijateljih obstaja profil tudi v tretjem družabnem omrežju, ki ga ta uporabnik ne uporablja ali pa tam s tem prijateljem ni povezan, algoritem potem v to povezavo doda še nov profil in uporabnik dobi dostop do podatkov, ki jih je prijatelj objavil v tem novem družabnem omrežju. Drugi scenarij je podoben. Ko se aktivira algoritem za povezovanje profilov, se v eni ali več povezavah najde nov profil, ki v bazi še ni zapisan, po potrditvi se v bazo zapiše nova povezava. Uporabniki, ki imajo ta povezan profil, bodo ob naslednjem pregledu stikov videli spremembo in lahko dostopali do podatkov, ki jih vsebuje nov profil. S tem algoritmom smo dosegli, da se na legalen način širijo podatki, do katerih prej uporabniki niso imeli dostopa.

Ugotovili smo, da je kljub temu, da se preko vmesnikov API da pridobiti veliko število podatkov, še vedno kar nekaj omejitev. Prav nobeno družabno omrežje ne dopušča

pridobivanja elektronskih naslovov prijateljev, prav tako ne moremo pridobiti nobene telefonske številke. Ta dva parametra bi še izboljšala delovanje algoritma za iskanje ujemanj, saj v večini uporabniki za družabna omrežja uporabljajo iste elektronske naslove, izjema so lahko profesionalna družabna omrežja, kot na primer LinkedIn, kjer uporabniki ponavadi uporabljajo drugačne elektronske naslove. Ko se ustvari profil v naši aplikaciji in se prenesejo podatki iz družabnih omrežij, bi bilo možno dodati pridobivanje podatkov tudi s kakšnim drugim pristopom [5]. Sicer naj bi bil pristop pridobivanja podatkov preko »third party« aplikacij najbolj zanesljiv pri zbiranju podrobnih podatkov, saj po raziskavah večina uporabnikov da aplikaciji popoln dostop do njihovega profila [9], vendar pa ker se nekaterih podatkov ne da pridobiti preko vmesnikov API, bi bilo zanimivo uporabiti še tehniko razčlenjevanja profila. S tem bi pridobili kakovostnejše slike profilov in morebitne dodatne podatke profilov, če prevzamemo, da je še vedno veliko število profilov privzeto javnih [2], vendar se na ta pristop ne moremo preveč zanašati, saj danes vse več strani uporablja dinamično nalaganje vsebine, občasno pa spreminjajo tudi strukture strani, kar še oteži pridobivanje podatkov [8]. Pri uporabnikih, ki niso delili svojih rojstnih letnic in domačih krajev, bi bilo zanimivo uporabiti algoritem za ugotavljanje manjkajočih atributov [7].

S pridobivanjem podatkov preko vmesnika API se pojavijo tudi nekatere težave. Poleg že omenjenega omejevanja pridobivanja nekaterih podatkov profila, je tu še osveževanje podatkov, ki lahko včasih traja zelo dolgo. Druga težava so omejitve števila dostopov, saj imajo prav vsa družabna omrežja omejitve do vmesnika API, bodisi število dostopov v uri bodisi dnevno. Zato smo morali razviti sistem predpomnenja, kjer se podatki osvežujejo na eno uro, oziroma uporabnik lahko vsili osveževanje že prej, če še ni dosegel limita števila dostopov, s tem pa smo tudi močno pospešili našo aplikacijo in hkrati nismo preveč zmanjšali ažurnosti podatkov.

Dodatna registracija v našo aplikacijo ima eno dobro stran, in sicer to, da dodatno zbiramo elektronske naslove uporabnikov, ki bi jih kasneje lahko uporabili tudi pri prikazu uporabniških kontaktov ali združevanju profilov. Lahko pa bi sestavili drugačno logiko za prijavo v sistem, kjer bi se uporabilo sisteme, ki jih ponujajo družabna omrežja, npr. »Facebook connect«, prijava s Twitterjem ali z »Google Friend Connect«. S tem bi odpravili dodatno registracijo, bi pa odprli dodatne težave, kot so na primer, kako vemo, da gre za istega uporabnika, če se enkrat prijavi s sistemom Facebook drugič s sistemom Google, v naši aplikaciji pa teh računov še ni združil. Potrebno bi bilo izdelati neko funkcionalnost, ki bi v takem primeru ponujala združevanje računov.

Nazadnje se ozrimo še malo v prihodnost in razmislimo, kaj vse bi lahko vmesniki API ponujali. Pri Facebooku so poleti 2011 zagnali storitev Skype, kjer uporabniki uvozijo vse svoje stike iz Skype računa, ki jih potem hrani Facebook. Ta bi lahko s časom začel ponujati

pridobivanje teh stikov preko vmesnika API. Na drugi strani bi lahko Google podatke pridobljene s sinhronizacijo z mobilnih telefonov, z Gmail računom povezal z njihovim družabnim omrežjem Google+, kjer bi kasneje tudi ponujal to preko svojega vmesnika API. To bi lahko s pridom izkoriščali in podatke na podoben način, kot to dela naša aplikacija, pridobivali ter jih potem delili tudi z uporabniki, ki sicer sploh nimajo dostopa do njih. Uporabnikova zasebnost bi bila na tak način močno ogrožena, saj bi nekdo posedoval njegove osebne podatke. Te podatke bi lahko uporabil v svojo korist in celo izsiljeval uporabnika.

6. Sklep

Družabna omrežja so danes postala nepogrešljiv del komunikacije modernega časa. Uporabniki, ki se vključujejo v ta omrežja, bi morali biti zelo previdni glede podatkov, ki jih bodo delili z ostalimi, saj s tem širijo izpostavljenost sebe in svojih bližnjih. S povečanjem števila uporabnikov na družabnih omrežjih raste tudi problem varstva osebnih podatkov. S tem ko so družabna omrežja ponudila še svoje vmesnike API, pa je problem varovanja podatkov postal še večji, saj se pojavlja vse več strani, ki s pridom izkoriščajo zbiranje podatkov. Kljub temu da so vmesniki API prinesli veliko dobrih stvari, od združevanja podatkov na enem mestu in lažjega pregleda vseh novosti, do prijav v druge sistema s pomočjo družabnih omrežjih, ima na drugi strani zbiranje osebnih podatkov tudi zelo slabe strani, od uporabe podatkov za oblikovanje oglasov glede na interese uporabnika do škodovanja uporabniku na tak in drugačen način s tem, ko je nek drug uporabnik prišel do njegovih zasebnih podatkov.

Menimo, da so družabna omrežja odličen način za ohranjanje stikov in komunikacijo ter obenem opozarjamo, da tudi če smo poskrbeli za varnost svojega profila, to še ne pomeni, da podatki ne morejo odtehati ven, ker se bo med prijatelji zagotovo našel nekdo, ki ima javen profil in bi podatki lahko preko njega prišli ven. Zavedati se moramo, da smo največkrat krivci za odtekanje podatkov kar uporabniki sami, ker smo premalo osveščeni, kje je meja med zasebnostjo in javnostjo.

Kazalo slik

Slika 1. Namizna aplikacija Yoono.	6
Slika 2. Pregled razvoja družabnih omrežij.	9
Slika 3. Facebookov podatkovni center. [13]	11
Slika 4. Prikaz toka avtorizacije s protokolom OAuth 1.0 [35].	16
Slika 5. Podatkovni model spletne aplikacije.	25
Slika 6. Forma za registracijo.	27
Slika 7. Forma za prijavo v aplikacijo.	28
Slika 8. Ikone za dostop do podstrani družabnih omrežij.	29
Slika 9. Prošnja za dovoljenje dostopa aplikacije do podatkov uporabnika pri omrežju Facebook. [41].	29
Slika 10. Registracija aplikacije na družabnem omrežju linkedin.com. [35]	30
Slika 11. Google konzola za pregled vseh Googlovih storitev. [36]	31
Slika 12. Povezave za osveževanje obvestil in sporočil.	33
Slika 13. Podstran namenjena Facebooku.	34
Slika 14. JavaScript koda za prikaz dodatnih prijateljev pri pregledu.	35
Slika 15. Prikaz obvestila v primeru, da se nov profil razlikuje od tistega, s katerim se je uporabnik najprej predstavil.	35
Slika 16. Google+ podstran.	36
Slika 17. Podstrani za družabno omrežje Twitter.	37
Slika 18. JavaScript funkcija, ki je zadolžena za štetje in prikaz znakov.	37
Slika 19. Podstran namenjena omrežju LinkedIn.	38
Slika 20. Koda, ki pokliče AJAX funkcijo za preverjanje dosegljivosti omrežij.	39
Slika 21. Koda, ki se izvede na strani strežnika in je zadolžena za preverjanje dosegljivosti povezave.	39
Slika 22. Javascript koda, ki je odgovorna za preklapljanje med profili povezanega profila.	40
Slika 23. JavaScript koda za filtriranje zadetkov pri iskanju prijateljev.	41
Slika 24. Prikaz rezultata najdenega povezanega profila.	43
Slika 25. Pojavno okno v katerem menjamo oziroma dodajamo profil v povezan profil.	44
Slika 26. Vnosno polje za objavo statusa v več družabnih omrežjih hkrati.	45

Kazalo tabel

Tabela 1. Število unikatnih mesečnih obiskovalcev družabnih omrežij. [40].....	10
Tabela 2. Pregled pravic za dostop do podatkov pri Facebook vmesniku API. [33]	32
Tabela 3. Polja, ki jih vsebuje niz podatkov, ki predstavlja prijatelje.	42

Literatura

- [1] A. Narayanan and V. Shmatikov, *De-anonymizing Social Networks*, marec 2009.
- [2] R. Gross, A. Acquisti, and H. Heinz, *Information revelation and privacy in online social networks*, In WPES, 2005.
- [3] A. Mislove, B. Viswanath, K. P. Gummadi, P. Drushel, *You are who you know: Inferreing User Profiles in online social network*, Februar 2009.
- [4] (2007) C. Anderson. Social networking is a feature, not a destination. Dostopno na: http://www.longtail.com/the_long_tail/2007/09/social-networki.html
- [5] J. Bonneau, J. Anderson, G. Denezis, *Prying data out of a social network*, 2009.
- [6] D. H. Chau, S. Pandit, S. Wang, and C. Faloutsos, *Parallel Crawling for Online Social Networks*, v WWW '07: Proceedings of the 16th international conference on World Wide Web, str. 1283–1284, 2007.
- [7] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, *Measurement and Analysis of Online Social Networks*, v IMC '07: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement, , str. 29–42, 2007
- [8] S. Ye, J. Lang, F. Wu, *Crawling online social graphs*, Januar 2010.
- [9] A. Felt and D. Evans, *Privacy Protection for Social Networking Platforms*, Workshop on Web 2.0 Security and Privacy, 2008.
- [10] Moo Nam Ko, G. P. Cheeck, M. Shebab, R. Sandhu, *Social-Networks connect services*, IEEE, 2010.
- [11] (2011) What is Facebook's architecture? Dostopno na: <http://www.quora.com/What-is-Facebooks-architecture>
- [12] (2010) Inside Facebook's Open Source Infrastructure. Dostopno na: <http://www.developer.com/open/article.php/3894566/Inside-Facebooks-Open-Source-Infrastructure.htm>
- [13] (2010) The Facebook Data Center FAQ. Dostopno na: <http://www.datacenterknowledge.com/the-facebook-data-center-faq/>
- [14] (2011) Hiphop. Dostopno na: <https://github.com/facebook/hiphop-php/wiki/>
- [15] (2009) Scaling Twitter: Making Twitter 10000 Percent Faster. Dostopno na: <http://highscalability.com/scaling-twitter-making-twitter-10000-percent-faster>

- [16] (2012) Ruby (programming language). Dostopno na: [http://en.wikipedia.org/wiki/Ruby_\(programming_language\)](http://en.wikipedia.org/wiki/Ruby_(programming_language))
- [17] (2010) LinkedIn's Data Infrastructure. Dostopno na: <http://www.infoq.com/news/2010/08/linkedin-data-infrastructure>
- [18] (2008) Web Scalability Practices: Bumper Sticker on Rails. Dostopno na: <http://blog.linkedin.com/2008/06/23/web-scalability-practices-bumper-sticker-on-rails/>
- [19] (2012) LinkedIn: About Us. Dostopno na: <http://press.linkedin.com/about>
- [20] (2011) Google+ Technological Details. Dostopno na: <http://www.infoq.com/news/2011/07/Google-Plus>
- [21] (2012) I'm a technical lead on the Google+ team. Ask me anything. Dostopno na: <http://anyasq.com/79-im-a-technical-lead-on-the-google+-team>
- [22] (2007) Inside MySpace: The Story. Dostopno na: <http://www.baselinemag.com/c/a/Projects-Networks-and-Storage/Inside-MySpacecom/>
- [23] (2009) MySpace Architecture. Dostopno na: <http://highscalability.com/myspace-architecture>
- [24] J. Dobnikar, *Nova generacija internetnih podjetij in njihovi izzivi pri oblikovanju poslovnih elementov*, 2009, pogl. 2.
- [25] D. M. Boyd, N. B. Ellison, *Social Network Sites: Definition, History, and Scholarship*, 2007, pogl. 2.
- [26] (2011) Top 15 Most Popular Social Networking Sites. Dostopno na: <http://www.ebizmba.com/articles/social-networking-websites>
- [27] (2012) OAuth. Dostopno na: <http://oauth.net/>
- [28] (2010) Introducing OAuth 2.0. Dostopno na: <http://hueniverse.com/2010/05/introducing-oauth-2-0/>
- [29] (2008) Če nimaš svojega profila, skoraj ne obstajaš. Dostopno na: <http://www.delo.si/clanek/70196>
- [30] (2010) How To Connect Your Social Media Profiles. Dostopno na: <http://zemalf.com/1107/connect-social-media-profiles/>
- [31] (2012) About FriendFeed. Dostopno na: <http://friendfeed.com/about/>
- [32] (2011) 8 Best Desktop Instant Messaging Software With Multiple Account Functions. Dostopno na: <http://ginva.com/2011/03/best-desktop-instant-messaging-software/>
- [33] (2012) Facebook developers. Dostopno na: <https://developers.facebook.com/docs/>
- [34] (2012) Twitter developer. Dostopno na: <https://dev.twitter.com/docs>
- [35] (2012) LinkedIn developers. Dostopno na: <http://developer.linkedin.com/apis>

- [36] (2012) Google+ developers. Dostopno na: <https://developers.google.com/+/api/>
- [37] (2011) How many websites are there in the. Dostopno na: <http://www.webdevelopersnotes.com/blog/how-many-websites-number/>
- [38] (2001) What Types of Social Networks Exist. Dostopno na: http://socialnetworking.lovetoknow.com/What_Types_of_Social_Networks_Exist
- [39] (2008) Poročilo in smernice za varstvo zasebnosti v spletnih družabnih omrežjih Dostopno na: <https://www.ip-rs.si/novice/detajl/porocilo-in-smernice-za-varstvo-zasebnosti-v-spletnih-druzabnih-omrezjih/>
- [40] (2012) The eBusiness Knowledgebase. Dostopno na: <http://www.ebizmba.com/>
- [41] (2012) Facebook. Dostopno na: <http://www.facebook.com>
- [42] (2012) Youtube. Dostopno na: <http://www.youtube.com>
- [43] (2012) Twitter. Dostopno na: <http://www.twitter.com>
- [44] (2012) Google+. Dostopno na: <http://plus.google.com>
- [45] (2012) Netlog. Dostopno na: <http://sl.netlog.com>
- [46] (2012) MySpace. Dostopno na: <http://www.myspace.com>
- [47] (2012) LinkedIn. Dostopno na: <http://www.linkedin.com>
- [48] A. T. Fiore and J. S. Donath. *Homophily in online dating: when do you like someone like yourself?*, 2005.
- [49] (2012) Do-It-Yourself Community. Dostopna na: <http://www.diychatroom.com/>
- [50] (2012) Oh My Bloom. Dostopno na: <http://www.ohmybloom.com/>
- [51] (2012) The Student Room. Dostopno na: <http://www.thestudentroom.co.uk/>
- [52] (2012) Academia. Dostopno na: <http://academia.edu>
- [53] B. Krishnamurthy and C. E. Wills, *Characterizing Privacy in Online Social Networks*, v WOSN: Workshop on Online Social Networks, 2008.
- [54] (2012) XML. Dostopno na: <http://sl.wikipedia.org/wiki/XML>