

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

Društveni, lokacijski i okolišni kontekst u informacijskim i komunikacijskim uslugama

Tehnička dokumentacija

Verzija 1.0

Studentski tim: Jurica Babić

Ante Ivanković

Ana Jandras

Igor Sambolec

Krešimir Slugan

Igor Vlahek

Nastavnici: Ignac Lovrek, Vedran Podobnik

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

Sadržaj

1.	Opis razvijenog proizvoda	5
1.1	Sustav za ostvarivanje ad-hoc društvenog umrežavanja (AdNET)	6
1.2	Usluga za određivanje najbližih prijatelja na društvenoj mreži Facebook	8
1.3	Vizualizacija PowerTAC natjecanja	9
1.4	Sustav za upravljanje korisničkim profilima	10
1.5	Sustav za slanje velikog broja poruka u društvenoj mreži	12
2.	Tehničke značajke	14
2.1	Tehničke značajke sustava AdNET	14
2.2	Tehničke značajke usluge za određivanje najbližih prijatelja na društvenoj mreži Facebook	19
2.3	Tehničke značajke vizualizacije PowerTAC natjecanja	21
2.4	Tehničke značajke sustava za upravljanja korisničkim profilima	25
2.5	Tehničke značajke sustava za slanje velikog broja poruka u društvenoj mreži	27
3.	Upute za korištenje	34
3.1	Upute za korištenje sustava AdNET	34
3.2	Upute za korištenje usluge za određivanje najbližih prijatelja na društvenoj mreži Facebook	36
3.3	Upute za korištenje vizualizacije PowerTAC natjecanja	38
3.4	Upute za korištenje sustava za upravljanje korisničkim profilima	40
3.5	Upute za korištenje sustava za slanje velikog broja poruka u društvenoj mreži	40
4.	Opis korištenih tehnologija	42
4.1	Baza podataka Mongo	42
4.2	Sustav za <i>push</i> -poruke na operacijskom sustavu Android	43
4.3	Okvir Spring	43
4.4	Knjižnica Primefaces	44

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

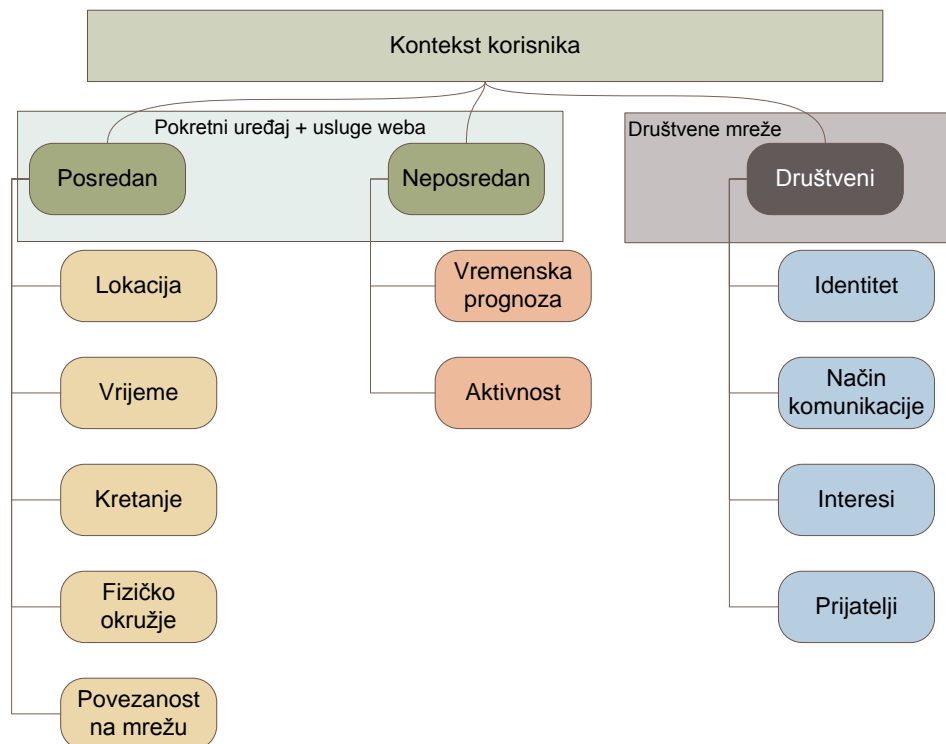
4.5	Tehnologije izgradnje semantičkih profila	44
5.	Literatura	47

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

1. Opis razvijenog proizvoda

U okviru projekta iskorištena su određena svojstva društvenih mreža i konteksta (okruženje usluge i korisnika) te su na temelju toga izvedeni modeli i implementacije odgovarajućih usluga. Kontekst promatran s aspekta korisnika prikazan je na slici (Slika 1). Elementi posrednog konteksta dobivaju se iz samog pokretnog uređaja dok se elementi neposrednog konteksta dobivaju kombiniranjem posrednog konteksta s uslugama weba. Društveni kontekst promatra odnose i karakteristike korisnika u nekoj društvenoj mreži (npr. *Facebook*).

Ostvareno je pet usluga koje koriste odgovarajući podskup elemenata konteksta (Slika 1). Navedene usluge će biti opisane u nastavku.

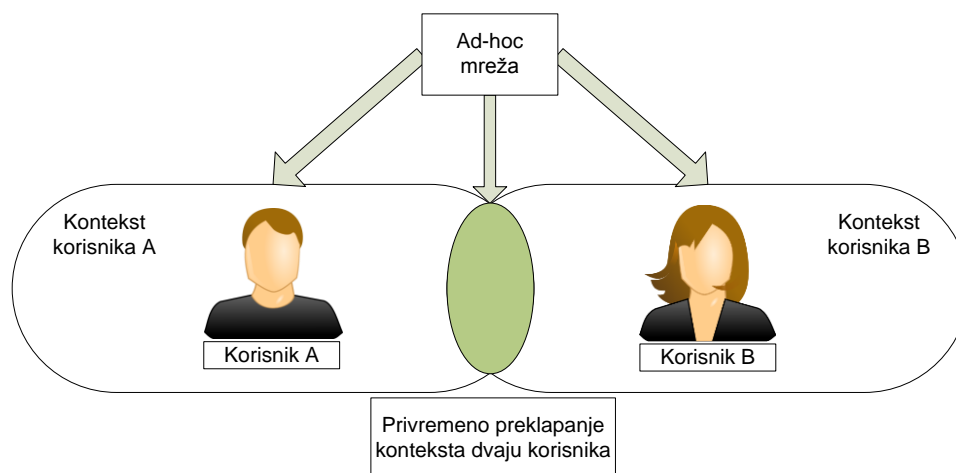


Slika 1. Kontekst korisnika

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

1.1 Sustav za ostvarivanje ad-hoc društvenog umrežavanja (AdNET)

Ad-hoc društvena mreža je privremena mreža korisnika koja se može ostvariti spontano na temelju konteksta korisnika i usluge. Za stvaranje ad-hoc mreže u obzir se mogu uzeti elementi iz svih kategorija konteksta. Primjerice, posredan kontekst može se opisati lokacijom i brzinom kretanja, neposredan kontekst vremenskom prognozom na temelju lokacije dok društveni kontekst interesima korisnika (npr. koje knjige voli, za koji klub navija i sl.). Na temelju tih informacija moguće je pružiti uslugu koja će umrežiti korisnike u trenutku u kojem se preklapaju određeni elementi konteksta korisnika (Slika 2). Kad preklapanje prestane (npr. korisnik je promijenio interese, lokaciju, brzinu i sl.).



Slika 2. Koncept ad-hoc društvenog umrežavanja

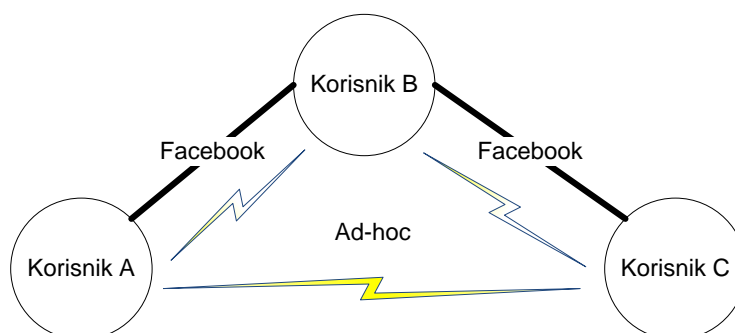
U sklopu ovog projekta realizirana je usluga koja omogućava ad-hoc umrežavanje korisnika uzimajući u obzir elemente konteksta koji su prikazani u tablici (Tablica 1). Elementi posrednog konteksta prikupljeni su iz pokretnog uređaja dok su elementi društvenog konteksta prikupljeni s društvene mreže Facebook. Sustav se naziva AdNET i temelji se na rješenju za ad-hoc umrežavanje *BFriend* [1] [2].

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

Tablica 1. Elementi konteksta korišteni u modeliranju usluge ad-hoc društvenog umrežavanja

Posredan	Društveni
Lokacija	Prijatelji
Vrijeme	Interesi
Brzina kretanja	Događaji
Povezanost na mrežu	

Usluga promatra pozive na događaje (engl. *event*) koji su upućeni korisniku i njegov odziv. Poziv na događaj (npr. koncert neke rock grupe) definiran je lokacijom i vremenom održavanja. Korisnik na događaj može odgovoriti potvrdno (prihvaća poziv) ili negativno (odbija poziv). Usluga omogućava umrežavanje korisnika ukoliko se preklape vrijeme i lokacija dvaju ili više korisnika s vremenom i lokacijom održavanja događaja na koji su ti korisnici potvrdno odgovorili. Dodatni uvjet je da su ti korisnici međusobno prijatelji ili da imaju zajedničkog prijatelja na društvenoj mreži Facebook što je prikazano na slici (Slika 3). *Korisnik A* i *korisnik C* imaju zajedničkog prijatelja *korisnika B* ali se oni međusobno ne poznaju. U slučaju da se korisnici koji imaju zajedničkog prijatelja tek upoznaju, o njima ovisi da li će stvoriti prijateljsku vezu na društvenoj mreži Facebook i to nije dio sustava AdNET. U nastavku slijedi ilustracija usluge jednim od mogućih scenarija.



Slika 3. Ilustracija odnosa veza mreže Facebook i ad-hoc.

Na društvenoj mreži Facebook, Ante i Marko imaju zajedničkog prijatelja Igora ali se Ante i

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

Marko međusobno ne poznaju i nisu prijatelji na toj mreži. Sva trojica koriste aplikaciju AdNET. U jednom trenutku na društvenoj mreži Facebook sva trojica su dobili pozivnicu na koncert grupe „Prljavo kazalište“ – događaj koji počinje u 22 sata 5.1.2012. u Zagrebu na stadionu Maksimir. Marko i Ante potvrđuju dolazak, a Igor ne. Na dan i u vrijeme koncerta Ante i Marko nalaze se na stadionu, sustav AdNET prepoznaje da ti korisnici imaju zajedničkog prijatelja, zajednički interes i lokacijski su bliski u vrijeme održavanja zajedničkog događaja stoga stvara privremenu ad-hoc vezu između Marka i Ante obavještavajući ih o tome preko njihovih pokretnih uređaja.

1.2 Usluga za određivanje najbližih prijatelja na društvenoj mreži Facebook

Pri korištenju društvenih mreža te izradi aplikacija za društvene mreže, često se nameće potreba za informacijom „koji su najbliži prijatelji nekog korisnika društvene mreže“. Iako se na prvi pogled čini da svaki korisnik ima svoj određen broj prijatelja te da su mu svi ti prijatelji jednako bliski, analizom interakcije na korisničkom profilu moguće je utvrditi koje su osobe korisniku „najbliskije“ i „najvažnije“.

Mnoge aplikacije na društvenim mrežama žele prosljediti određene informacije samo odabranom skupu prijatelja od korisnika koji je koristio aplikaciju. U tu svrhu, bilo bi idealno kada bi postojala neka postojana web-usluga koja bi mogla u nekoliko sekundi pouzdano izračunati i vratiti takvu listu korisnika društvene mreže.

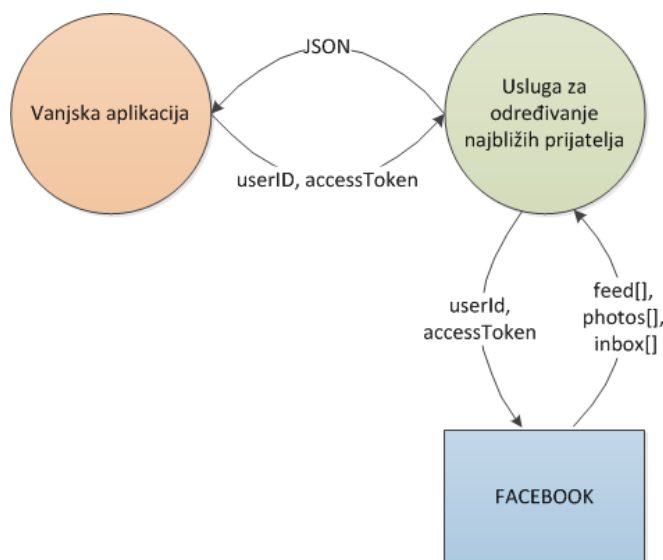
Usluga za određivanje najbližih prijatelja na društvenoj mreži Facebook napravljena je upravo s ciljem da pomogne u prethodno navedenim situacijama. Ova web-usluga namijenjena je razvijateljima aplikacija za društvene mreže, te im može uvelike pomoći da svoj rad koncentriraju oko funkcionalnosti vlastite aplikacije, a da sav izračun liste najbližih prijatelja prepuste ovoj web-usluži.

Osim što je ova usluga napravljena kao web-usluga, moguće ju je također koristiti u obliku zabavne i informativne Facebook aplikacije, gdje svaki korisnik može vidjeti s kojim prijateljima ima najveću interakciju, koji mu prijatelji šalju najviše poruka, koji mu najviše komentiraju i koji mu najviše pišu po „Wallu“.

Konačan sustav je razvijen u obliku web-usluge koji se nalazi kao dodatan alat da bi neka aplikacija na društvenoj mreži Facebook brže i lakše obavljala svoj „primarni“ posao. Slika 4 prikazuje upotrebu sustava od strane vanjske aplikacije. Dakle, vanjska aplikacija se spaja na web-uslugu i predaje joj *Access Token* korisnika koji koristi aplikaciju. Nakon toga, web-usluga za određivanje najbližih

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

prijatelja na društvenoj mreži Facebook izračunava podatke dohvaćajući korisnikove osobne podatke s njegovog profila na društvenoj mreži Facebook, te naposljetku vraća rezultat u obliku JSON objekta vanjskoj aplikaciji. Potom vanjska aplikacija može nastaviti sa svojim daljnjim radom, koristeći i usmjeravajući dobivene podatke.



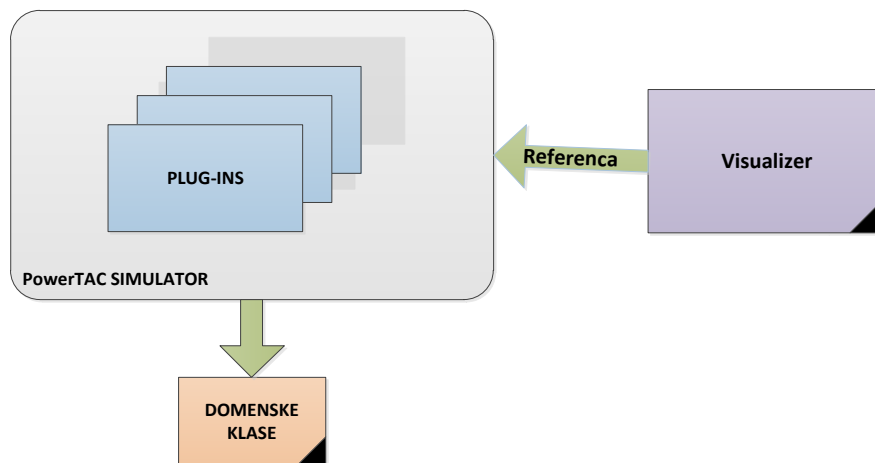
Slika 4. Položaj usluge za određivanje najbližih prijatelja na društvenoj mreži Facebook

1.3 Vizualizacija PowerTAC natjecanja

PowerTAC je primjer natjecanja agenata u kojem autonomni programski agenti djeluju kao maloprodajni posrednici unutar definirane regije u kojoj su prisutni obnovljivi izvori energije. Zadaci koje agenti pritom obavljaju uključuju kupovinu i prodaju električne energije. Energiju je moguće kupovati na dva osnovna načina: preko veleprodajnog tržišta ili posredstvom lokalnih izvora na maloprodajnom tržištu poput kućanstava, poslovnih objekata ili električnih automobila. Pribavljenu količinu energije posrednik isporučuje svojim lokalnim korisnicima dok se ostatak energije stavlja na trgovanje zainteresiranim stranama na veleprodajnom tržištu energije. Kako bi riješili problem lanca opskrbe električnom energijom, programski agenti moraju komunicirati s raznim entitetima poput veleprodajnog tržišta, energetske kompanije i tržišta energetske tarife.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

U sklopu ovog projekta realizirana je grafička komponenta *Visualizer* natjecanja PowerTAC u obliku web-aplikacije. Pojednostavljeni pregled sustava PowerTAC kao i način integracije *Visualizera* moguće je vidjeti na slici (Slika 5). Simulator je izgrađen od više projekata od kojih svaki opisuje po jedan entitet natjecanja (na slici označeno plavom bojom). Tako postoje projekti koji opisuju ponašanje maloprodajnih korisnika, veleprodajnih korisnika, entiteta za prijenos energije i drugo. Dodatno, simulator koristi projekt u kojemu se nalaze razredi koji opisuju domenu natjecanja. Kako agenti sa simulatorom moraju komunicirati koristeći standardni skup poruka, navedeni je projekt izdvojen iz simulatora te ga je moguće koristiti u zasebnim projektima.



Slika 5. Prikaz sustava PowerTAC.

Kako trenutno ne postoji upravljačko sučelje za administraciju natjecanja, za potrebu izrade *Visualizera*, dodatno je implementirano pojednostavljeno administracijsko sučelje za pokretanje i zaustavljanje instance natjecanja.

Konačan rezultat ovog projekta jest web-aplikacija *Visualizer* koja omogućuje bogat i intuitivan prikaz relevantnih informacija o aktivnostima pojedinim entitetima natjecanja PowerTAC. Kao sastavni dio paketa PowerTAC, *Visualizer* će poslužiti kao programsko pomagalo za validaciju uspješnosti natjecateljskih agenata.

1.4 Sustav za upravljanje korisničkim profilima

Korisnički profili danas bogati su podaci nastalim bilo izravnim bilo neizravnim posredovanjem

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

tablice (Tablica 2) što uključuje čitav društveni kontekst te lokaciju korisnika.

Tablica 2. Elementi konteksta korišteni pri upravljanju korisničkim profilima

Posredan	Društveni
Lokacija	Identitet
	Interesi
	Događaji
	Prijatelji

1.5 Sustav za slanje velikog broja poruka u društvenoj mreži

Mnoge aplikacije weba koriste neke od sustava koji omogućuju slanje raznih obavijesti putem e-maila. Takvi sustavi omogućuju razvijateljima aplikacija weba da se orijentiraju na razvijanje aplikacijske logike koristeći provjerene sustave za slanje obavijesti. Sustav za slanje velikog broja poruka u društvenoj mreži jest sustav namijenjen razvijateljima aplikacija za društvene mreže. Sustav omogućuje slanje obavijesti u obliku e-maila te slanje obavijesti u obliku specifičnim za društvene mreže.

Sustav je razvijen u obliku web-usluge koji se nalazi kao posrednik između aplikacije koja koristi sustav i korisnika kojem određena aplikacija šalje određene obavijesti. Slika 7 prikazuje upotrebu sustava od strane vanjske aplikacije. Vanjska aplikacija spaja se na sustav te šalje određene parametre u sustav. Na temelju parametara sustav šalje određenu poruku određenom korisniku. Kao što je prikazano na slici 7, sustav za slanje velikog broja poruka u društvenoj mreži služi kao posrednik između vanjske aplikacije i korisnika te aplikacije. Vanjska aplikacija ne sadrži potrebnu logiku potrebna za slanje različitih obavijesti korisnicima. Aplikacija koristi sustav u kojoj je sadržana logika za slanje različitih obavijesti. Vanjska aplikacija mora samo implementirati logiku za slanje poruka našem sustavu, koji brine da se određena obavijest sigurno i brzo dostavi korisniku kojem je ta poruka bila namijenjena. Za potrebe projekta korišteni su elementi konteksta iz tablice 3.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.



Slika 7. Interakcija aplikacije i korisnika

Tablica 3. Elementi konteksta korišteni u modeliranju sustava za slanje velikog broja poruka u društvenoj mreži

Posredan	Društveni
Povezanost na mrežu	Način komunikacije

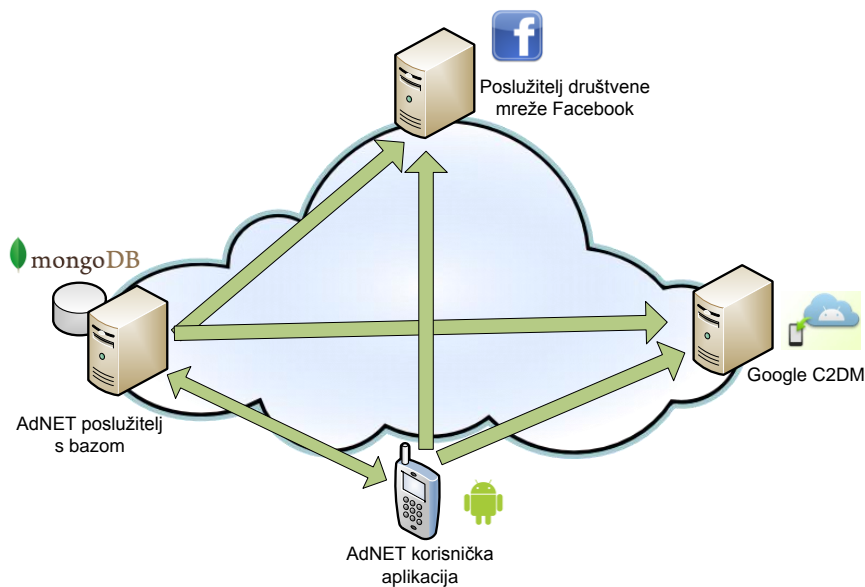
CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

2. Tehničke značajke

U nastavku su opisane tehničke značajke pojedine usluge.

2.1 Tehničke značajke sustava AdNET

Sustav AdNET temelji se na postojećem rješenju za ad-hoc umrežavanje *BFriend*. Sustav se sastoji od entiteta prikazanih na slici (Slika 8). Kako bi se omogućila bolja skalabilnost uvedena je baza podataka Mongo [3] i tehnologija C2DM (*Cloud To Device Messaging*) [4] koja je namijenjena za implementaciju *push*-poruka¹ na operacijskim sustavom Android. Navedene tehnologije opisane su u poglavlju 4.

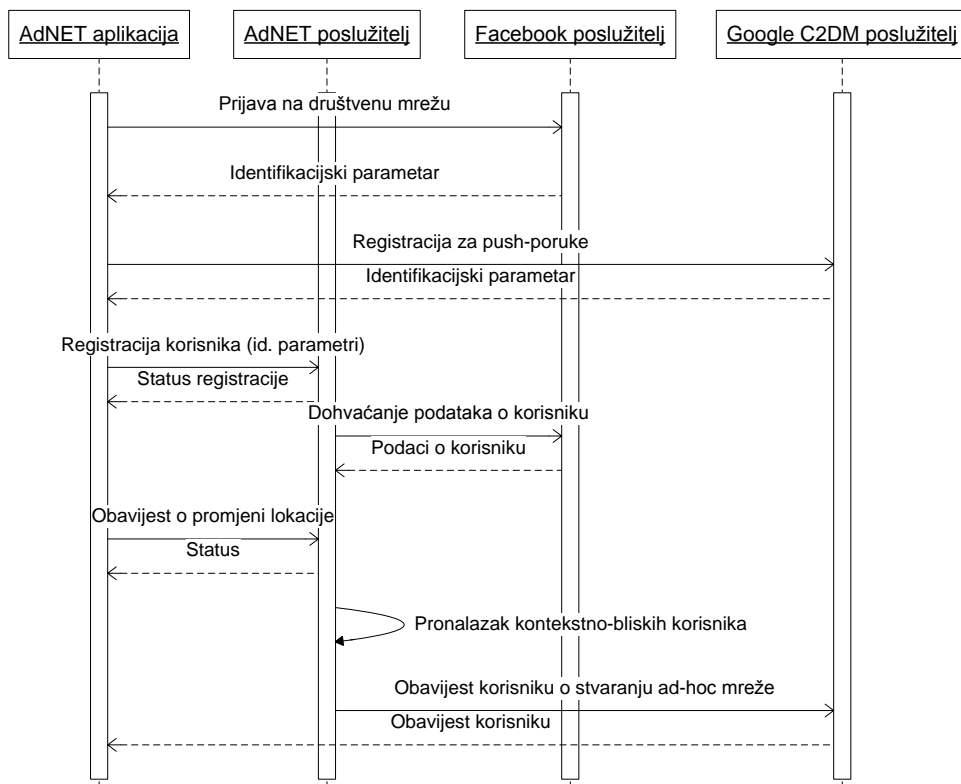


Slika 8. Skica AdNET sustava

Proces komunikacije između entiteta je prikazan na slici 9. Prilikom pokretanja aplikacije, uz potrebne podatke i dozvole korisnika, pristupa društvenoj mreži Facebook i preuzima identifikacijski parametar koji omogućava preuzimanje informacija s te društvene mreže bez interakcije s korisnikom. Potom se aplikacija se registrira na *push*-poslužitelj te preuzima potreban identifikacijski parametar. Identifikacijske parametre šalje na AdNET poslužitelj koji ih kasnije koristi za dohvaćanje podataka o

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

korisniku s društvene mreže Facebook i slanje informacija o stvorenim ad-hoc vezama prema korisnicima. Ukoliko se promjeni lokacija korisnika, o tome AdNET aplikacija obavještava AdNET poslužitelj koji potom pokušava pronaći korisnike s bliskim kontekstom. Ukoliko ih nađe, obavještava ih o stvaranju ad-hoc mreže između njih.



Slika 9. Slijedni dijagram sustava AdNET

Korisnička aplikacija sustava AdNET

Korisnička aplikacija je razvijena za pokretne uređaje s operacijskim sustavom Android. Pojednostavljeni dijagram razreda prikazan je na slici 10. Dva ključna razreda su `Main` i `MainService`. Razred `Main` je središnji dio koji korisniku omogućava sve aktivnosti kao što su registracija i pokretanje mehanizma praćenja lokacije i dr. Razred `MainService` omogućava omogućuje konstantno praćenje

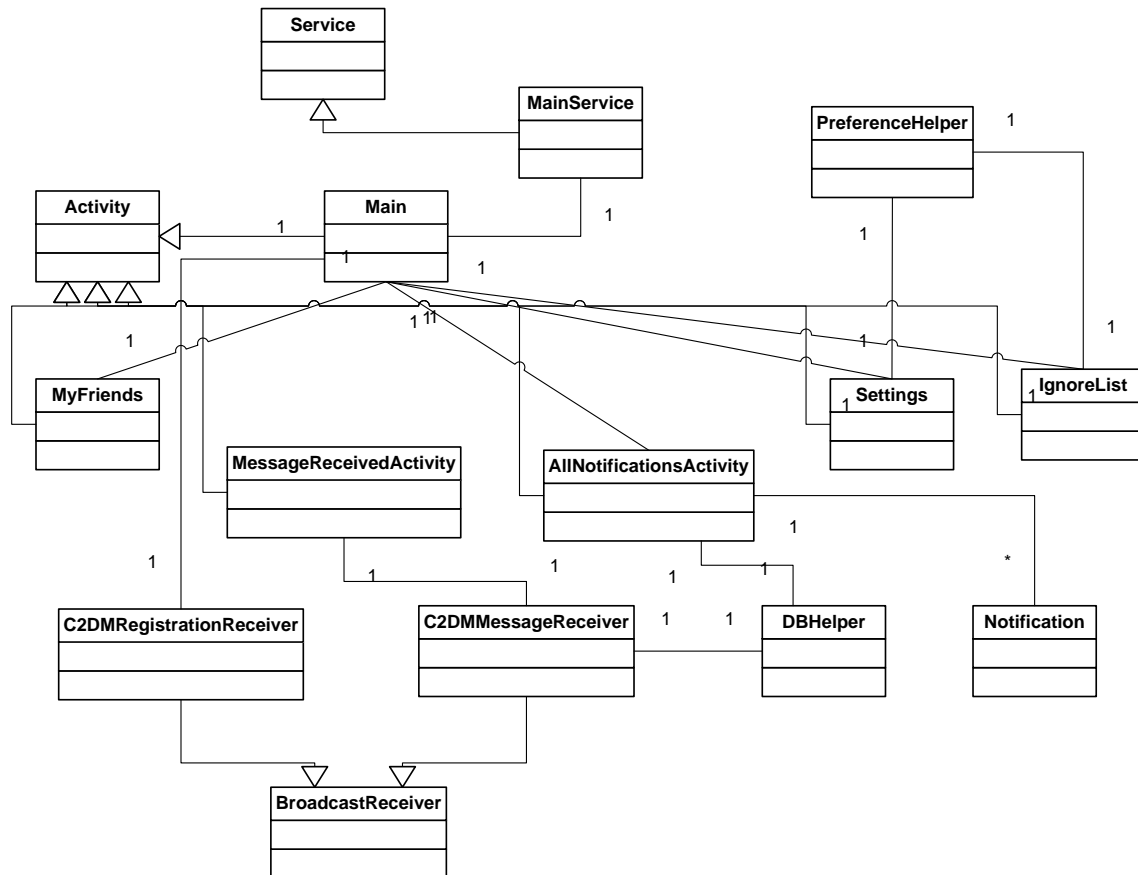
¹ *Push*-poruke su poruke koje su automatski prosljeđene do samog cilja (npr. e-mail, SMS-poruke i sl.)

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

lokacije korisnika i dojavljivanje poslužitelju sustava AdNET. Ostali razredi i njihove funkcije su sljedeće:

- `MyFriend` – omogućava pregled svih prijatelja s kojima je bila stvorena ad-hoc veza,
- `MessageReceivedActivity` – prikazuje korisniku poruku o stvaranju ad-hoc mreže,
- `AllNotificationsActivity` – prikazuje korisniku sve poruke o ad-hoc umrežavanju koje je primio,
- `Setting` – omogućava korisniku upravljanje postavkama aplikacije,
- `IgnoreList` – omogućava korisniku pregled svih korisnika s kojima ne želi uspostavljati ad-hoc mrežu,
- `C2DMRegistrationReceiver` – služi za registraciju korisničke aplikacije s C2DM poslužiteljem i dobivanje identifikacijskog parametra,
- `DBHelper`, `PreferenceHelper` – pomoćni razredi za pohranu podataka,
- `Notification` – pomoćni razred za spremanje obavijesti koje korisnik prima.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

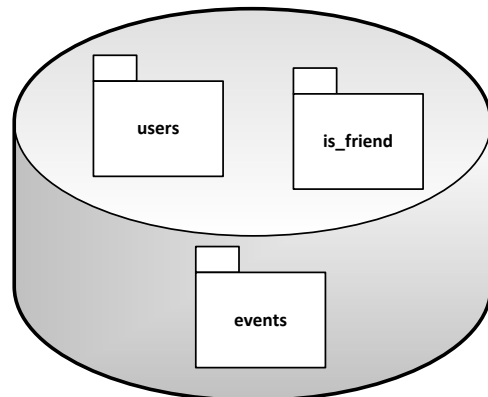


Slika 10. Dijagram razreda korisničke aplikacije sustava AdNET

Poslužiteljska aplikacija sustava AdNET

Poslužiteljski dio je razvijen kao nadogradnja na postojeći sustav *BFriend* uz određene preinake; iz inicijalnog projekta je uklonjena relacijska baza podataka MySQL [5] i uvedena je baza podataka Mongo radi njenih boljih performansi i mogućnosti skaliranja. Za razliku od baze podataka MySQL, Mongo ne sadrži relacijske tablice, već kolekcije dokumenata. Jedan dokument sadrži informacije na razini jednog korisnika. Struktura dokumenata nije strogo definirana i programer se brine o integraciji podataka. Detaljniji opis baze podataka Mongo dostupan je u 4. poglavlju. Na slici (Slika 11) vidljive su kolekcije koje se nalaze u bazi podataka ovog sustava.

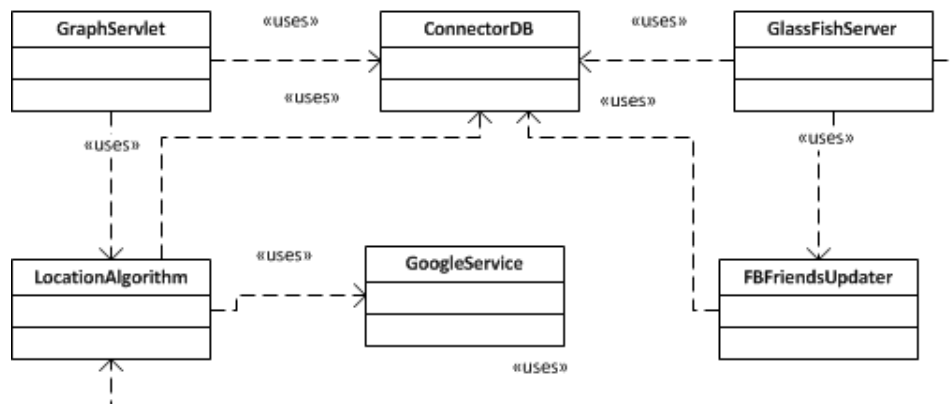
CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.



Slika 11 Kolekcije u bazi podataka Mongo

Unutar kolekcije *users* se nalaze podaci o korisniku te podaci potrebni za spajanje na usluge koje koristi ovaj sustav; društvenu mrežu Facebook i poslužitelj za uslugu slanja *push*-poruka. Kolekcija *is_friend* pohranjuje veze između korisnika odnosno prijateljstva. Kolekcija *events* sadrži podatke o događajima kojima korisnici prisustvuju.

Funkcionalni dio sustava izvršava se na poslužitelju Glassfish [6] i razvijena je u programskom jeziku Java. Dijagram razreda na slici 12. prikazuje odnos između razreda i međusobne odnose.



Slika 12 Dijagram razreda poslužiteljske aplikacije

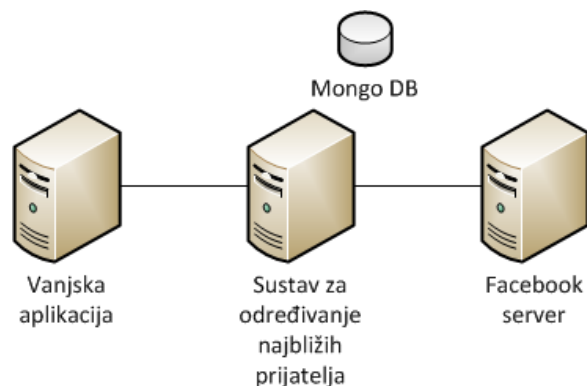
Razred `ConnectorDB` ostvaruje vezu sa bazom Mongo i zato je koriste svi ostali razredi kod kojih je prisutno pohranjivanje ili čitanje iz baze. Klijent kada se spaja na sustav pristupa preko razreda `GlassFishServer` koji nasljeđuje razred `HTTPServlet`. Unutar sustava se pokreće periodičko praćenje lokacije korisnika te osvježavanje liste njegovih prijatelja sa društvene mreže Facebook. Poslužitelj uz to

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

prati događaje (engl. *event*) na kojima je korisnik odlučio prisustvovati. Bilježi se opis i vrijeme događaja. Ukoliko je netko od korisnikovih prijatelja također objavio da će prisustvovati tom događaju sustav počinje pratiti njihove aktivnosti. `LocationAlgorithm` treba izračunati koliko su korisnici blizu jedan drugome na temelju njihovih koordinata. Ako se u vrijeme događaja nađu međusobno blizu, oboje su obaviješteni o prisutnosti i aktivnosti onog drugog preko *push*-poruke.

2.2 Tehničke značajke usluge za određivanje najbližih prijatelja na društvenoj mreži Facebook

Sustav za određivanje najbližih prijatelja na društvenoj mreži Facebook prikazan je na slici 13. Sustav se sastoji od poslužitelja weba Apache [7] te nerelacijske baze podataka (Mongo). Slika (Slika 13) prikazuje sustav za određivanje najbližih prijatelja u društvenoj mreži u interakciji s vanjskim komponentama za čije je korištenje sustav i inicijalno napravljen. Sustav za određivanje najbližih prijatelja nalazi se u sredini zainteresiranih strana: vanjske aplikacije weba kojoj je cilj imati informaciju o najbližim prijateljima svog originalnog korisnika, te društvene mreže Facebook koja putem svog API-ja (*Application Programming Interface*) pruža kvalitetnu potporu dohvat i obradi svih informacija svakog korisničkog profila.



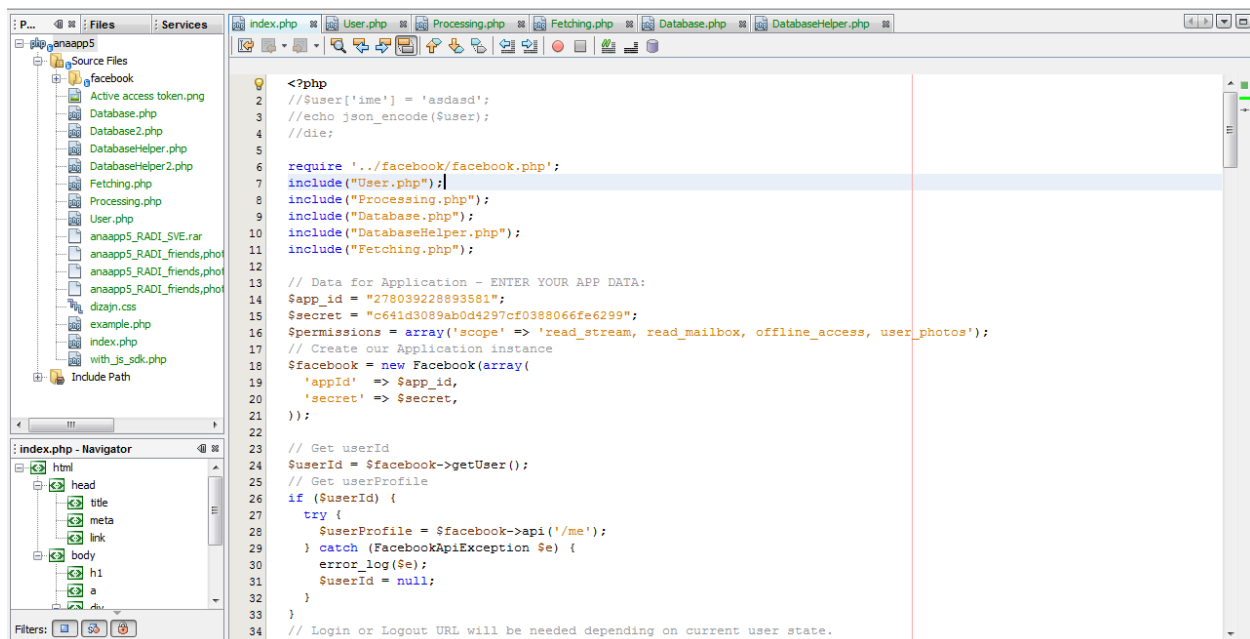
Slika 13. Arhitektura sustava za određivanje najbližih prijatelja

Kada se u vanjskoj aplikaciji pojavi potreba za listom najbližih prijatelja svog korisnika, ona pošalje upit sustavu za određivanje najbližih prijatelja te nakon primitka rezultata može sa pouzdanošću nastaviti raditi svoj primarni zadatak orijentiran ka svojim ciljanim korisnicima.

Sustav za određivanje najbližih prijatelja određenog korisnika u društvenoj mreži Facebook

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

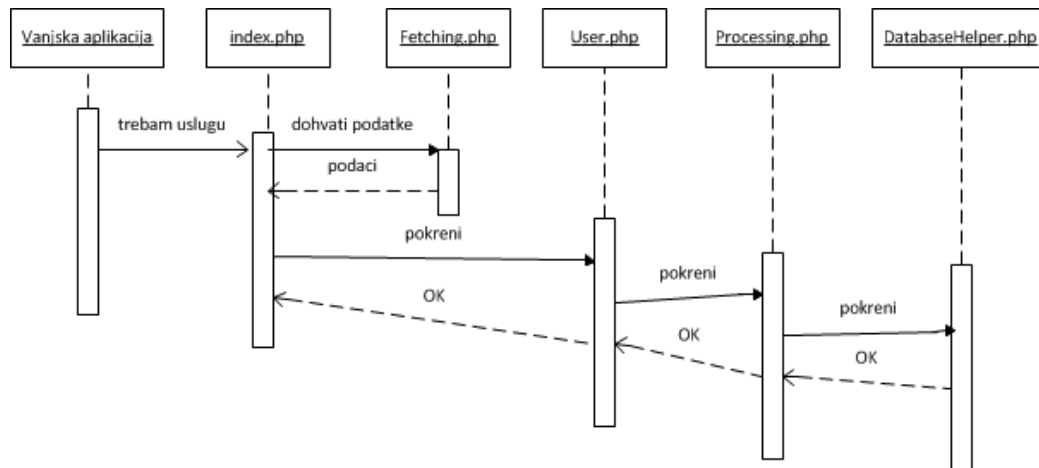
razvijen je kao web-aplikacija u programskom jeziku PHP. Arhitektura sustava jest objektno-orijentirana, te je raspoređena u 4 razreda: `User.php`, `Processing.php`, `Fetching.php` i `DatabaseHelper.php`. Također, postoji i apstraktni razred `Database.php` koji je zajednički sustavu za određivanje najbližih prijatelja te sustavu za slanje velikog broja poruka u društvenoj mreži. Razlog tome je što su oba sustava razvijena nad nerelacijskom bazom podataka Mongo, te su metode za pristup bazi i manipulacijom sadržajem baze identične. Glavna skripta koja se pokreće nazvana je `index.php`, te može vraćati ili JSON objekt ukoliko se usluga koristi kao web-usluga ili HTML (*HyperText Markup Language*) kod ukoliko se usluga koristi kao jednostavna Facebook aplikacija. Izgled radnog prozora u alatu Netbeans prikazan je na slici 14.



Slika 14. Radni prozor u Netbeansu i raspodjela razreda

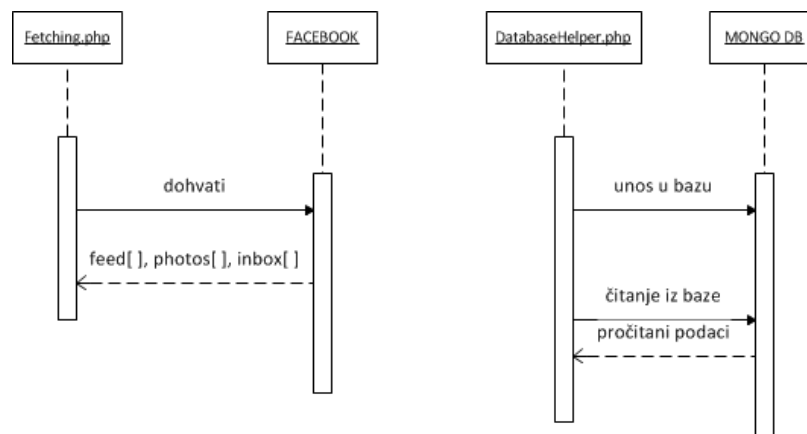
Proces komunikacije između sustava i vanjskih komponenti prikazan je na slici 4 u obliku slijednog dijagrama aktivnosti.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.



Slika 15. Slijedni dijagram aktivnosti sustava

Proces dohвата podataka s mreže Facebook i unošenja podataka u bazu podataka Mongo prikazan je na slici 16 u obliku slijednog dijagrama.



Slika 16. Slijedni dijagram dohвата podataka

2.3 Tehničke značajke vizualizacije PowerTAC natjecanja

Grafička komponenta *Visualizer* natjecanja PowerTAC jest bogata web-aplikacija temeljena na Java tehnologijama. Kako je ostatak paketa PowerTAC temeljen na okviru Spring [8], u svrhu jednostavne integracije *Visualizera* s postojećim paketom, poslovna logika *Visualizera* također je izrađena u Springu. Kako je jedan od glavnih zahtjeva bio bogat i intuitivan grafički dizajn, za implementaciju prezentacijskog dijela *Visualizera* korištena je knjižnica *Primefaces* [9] temeljena na JSF (*JavaServer*

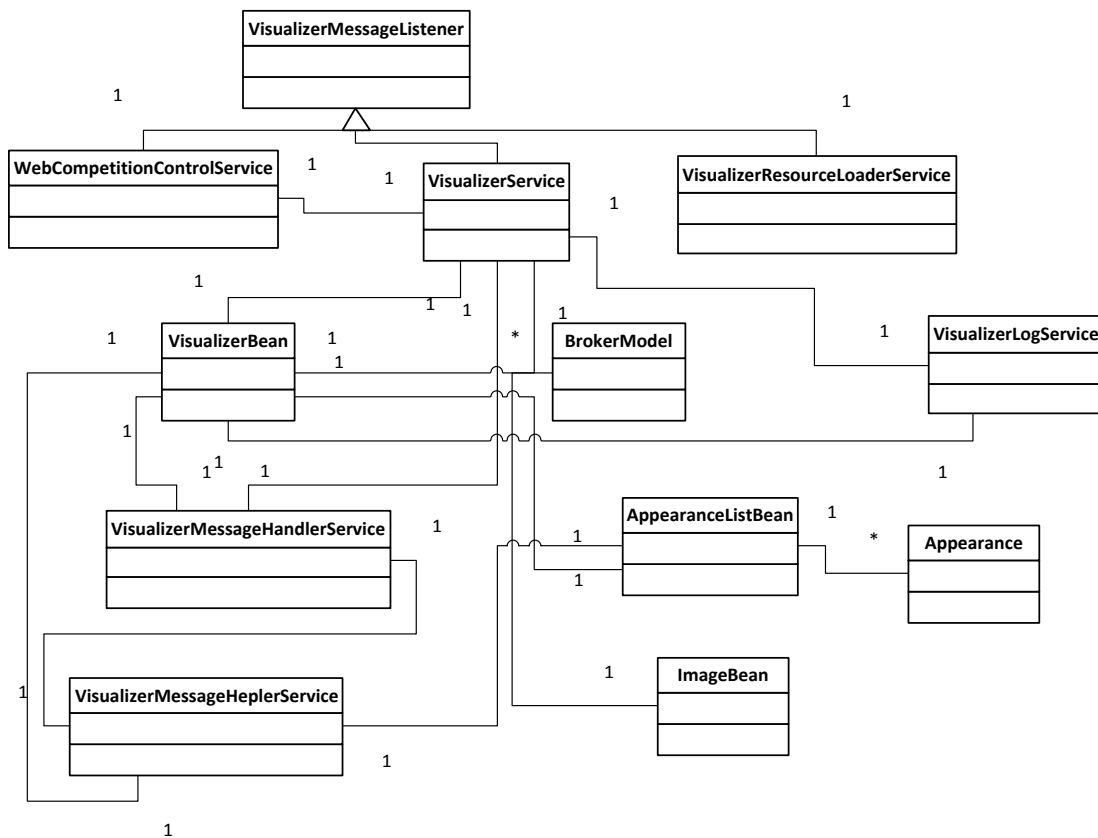
CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

Faces) [10] tehnologiji.

Dijagram razreda moguće je vidjeti na slici (Slika 17). Razred `WebCompetitionControlService` opisuje uslugu administracije instance natjecanja. Zahvaljujući njoj, korisnik je u stanju preko internetske stranice poslati zahtjev za pokretanje novog natjecanja ili prekidanje postojećeg. Razred `VisualizerService` implementira sučelje `VisualizerMessageListener` te omogućava primanje poruka iz simulatora. Po primitku poruke, ovisno o njezinom tipu, poruka se prosljeđuje odgovarajućoj metodi u objektu razreda `VisualizerMessageHandler`. Ostali razredi i njihove glavne funkcije su:

- `VisualizerResourceLoaderService` – nastaje *singleton* objekt koji omogućava učitavanje datotečnih resursa koji se nalaze unutar aplikacijskog *classpatha*,
- `VisualizerBean` – razred koja opisuje stanje *Visualizera*. Nastali *singleton* objekt koristi se za dohvat informacija u prezentacijski sloj,
- `BrokerModel` – domenski razred koji sadrži informacije o agentu,
- `VisualizerLogService` – omogućava dinamičko postavljanje *logger*a,
- `VisualizerMessageHandlerService` – nastali *singleton* objekt obrađuje primljene poruke te održava stanje *Visualizera*,
- `VisualizerMessageHelperService` – sastoji se od skupa pomoćnih metoda koje pomažu prilikom obrade primljenih poruka,
- `AppearanceListBean` – sastoji se od liste objekata koji opisuju vanjski izgled posrednika. Nastali objekt dodatno vodi računa da svaki posrednik dobije jedinstveni vanjski izgled,
- `ImageBean` – sadrži često korištene grafičke resurse,
- `Appearance` – opisuje vanjski izgled posrednika,

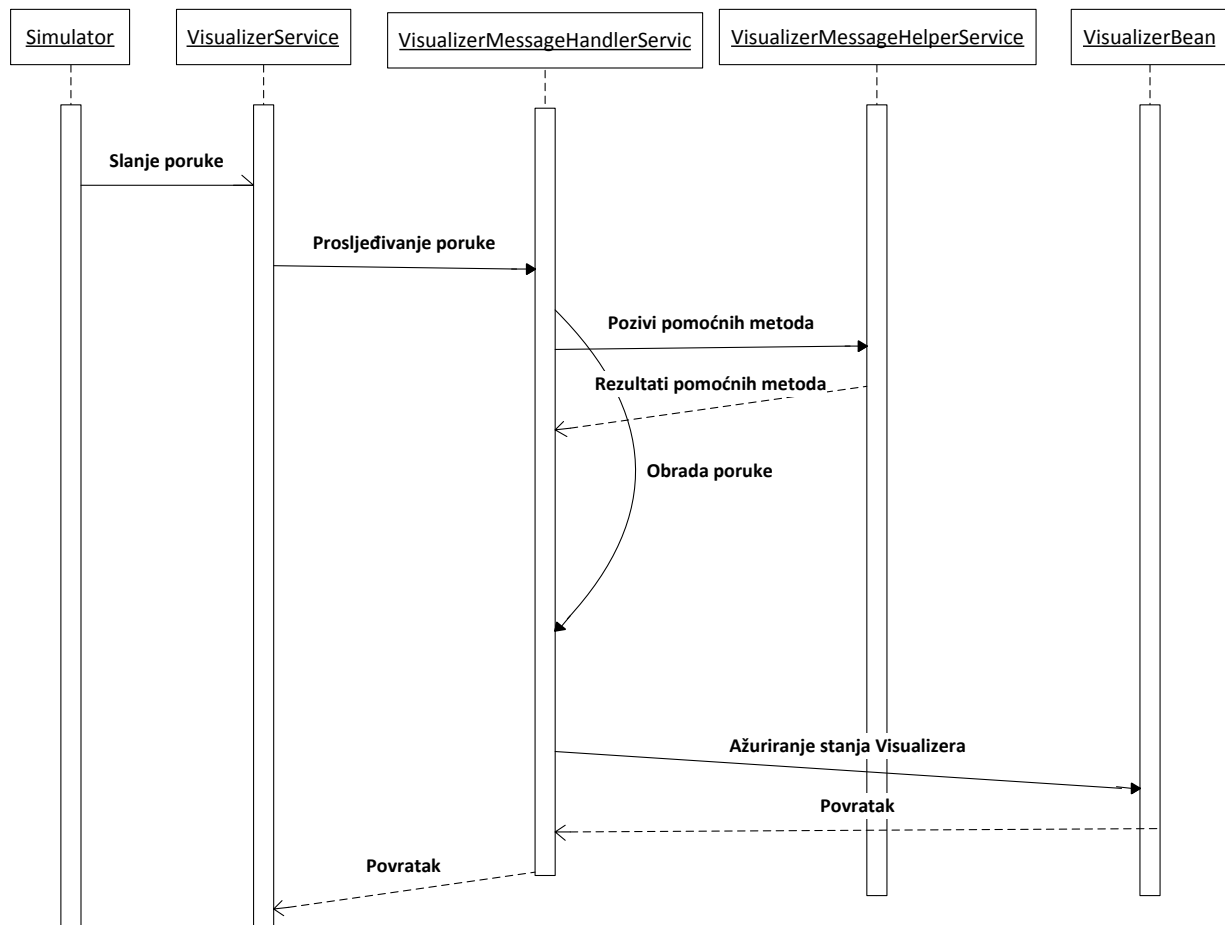
CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.



Slika 17. Pojednostavljeni dijagram razreda web-aplikacije *Visualizer*.

Slijedni dijagram koji prikazuje tipičan ciklus obrade primljene poruke može se vidjeti na slici (Slika 18).

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.



Slika 18: Sekvencijski dijagram obrade primljene poruke.

Prvi korak jest primanje poruke iz Simulatora. *VisualizerService* na temelju tipa poruke odabire odgovarajuću metodu za obradu poruke te ju prosljeđuje objektu razreda *VisualizerMessageHandlerService*. Obrada poruke sastoji se od izvlačenja korisnih informacija iz primljene poruke. Pritom su mogući sinkroni pozivi pomoćnih metoda koji se nalaze u razredu *VisualizerMessageHelperService*. Nakon što su pripremljene korisne informacije iz poruka, potrebno je ažurirati stanje *Visualizera* u *singleton* objektu razreda *VisualizerBean*. Nakon ažuriranja stanja, kontrola toka se vraća sve do objekta razreda *VisualizerService* te se čeka na primitak nove poruke.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

2.4 Tehničke značajke sustava za upravljanja korisničkim profilima

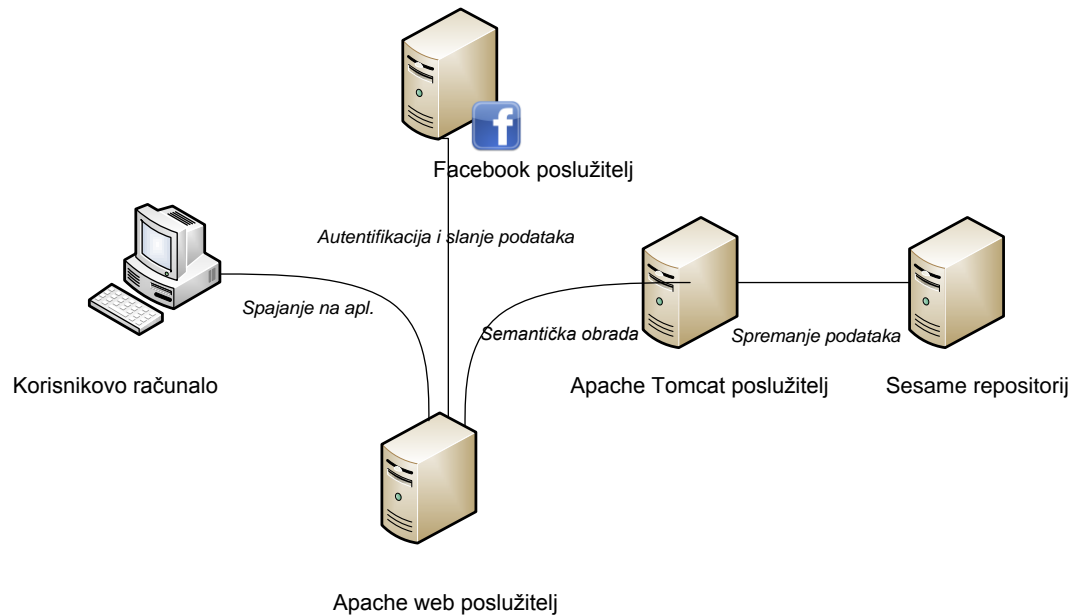
Sustav za upravljanje korisničkim profilima sastoji se od nekoliko dijelova od kojih su najvažniji (detaljnija arhitektura dana je na slici 19):

- web-poslužitelj PHP [11],
- Apache Tomcat poslužitelj [12] i,
- Sesame 2 repozitorij [13].

Sa tehničke strane rad sustava se može opisati u nekoliko koraka:

- korisnik se spaja na sustav (Apache web-poslužitelj) koji od njega traži da se autentificira sa svojim Facebook podacima,
- nakon autentifikacije korisnik se ponovno spaja na sustav koji sada može dobiti šifru za preuzimanje korisničkih podataka (*token*),
- podaci se preuzimaju sa Facebook poslužitelja i prosljeđuju dalje Apache Tomcat poslužitelju koji ih zapisuje u prikladnom RDF (*Resource Description Framework*) formatu te takve šalje poslužitelju Sesame 2 odnosno sprema ih u semantički repozitorij,
- nakon toga podaci su spremni za obradu (kreiranje upita nad semantičkom bazom, proširivanje profila i sl.).

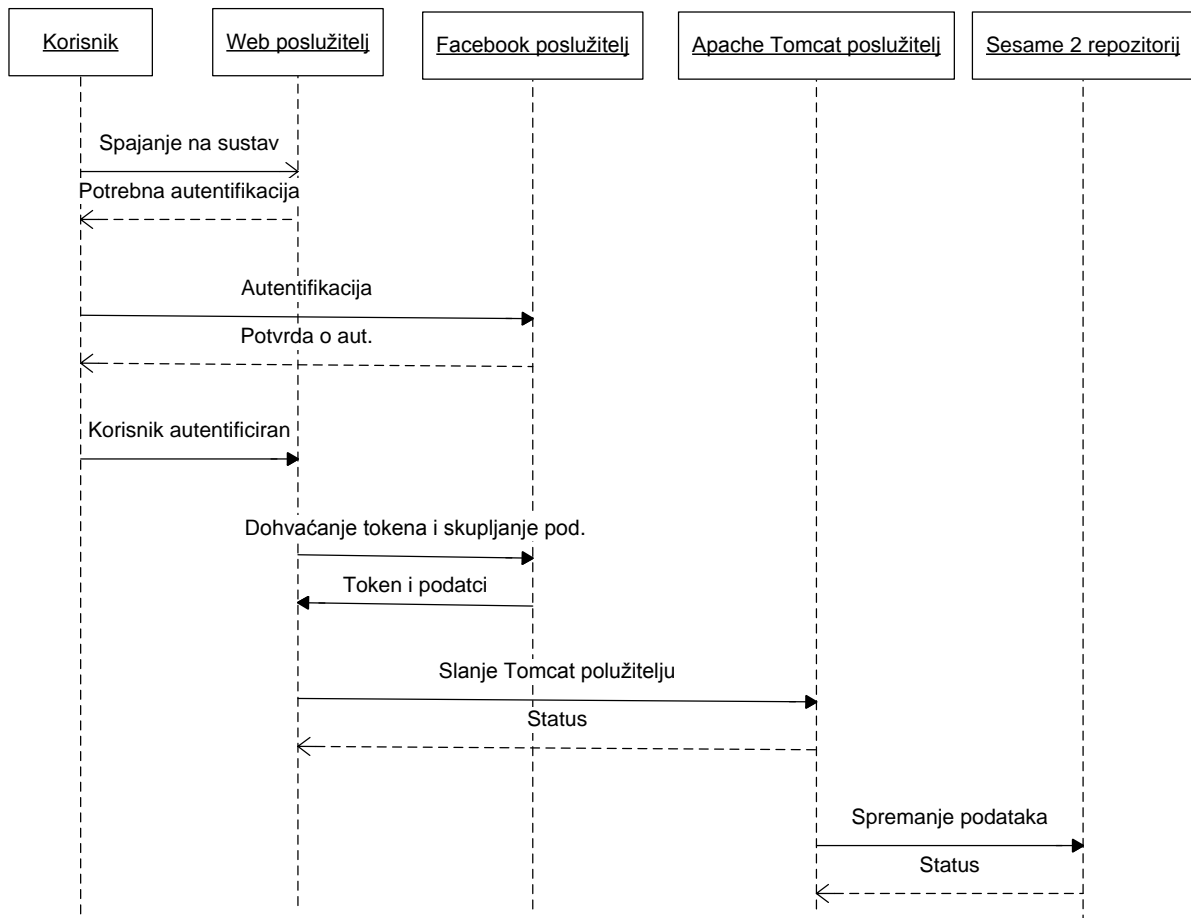
CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.



Slika 19. Arhitektura sustava za upravljanje profilima

Rad sustava detaljno je opisan sljedećim dijagramom na slici 20. Bitan dio sustava je *Java servlet* aplikacija koja se vrti na poslužitelju Apache Tomcat. Upravo tu je sadržana sva logika semantičkog rasuđivanja i jezgra funkcionalnosti aplikacije. Za to su korištene sljedeće tehnologije i jezici: RPID format podataka (engl. *Rich Presence Information Data format*) kao okvir za kreiranje profila, RDF kao način zapisa profila, OWL (*Web Ontology Language*) jezik za zapisivanje ontologija, Sesame 2 repozitorij za spremanje, dohvaćanje i upravljanje semantičkim profilima. Dio navedenih tehnologija detaljnije je opisan u poglavlju 4.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

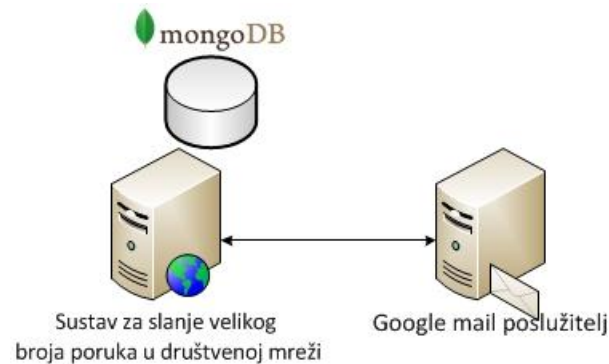


Slika 20. Sekvencijski dijagram sustava za upravljanje profilima

2.5 Tehničke značajke sustava za slanje velikog broja poruka u društvenoj mreži

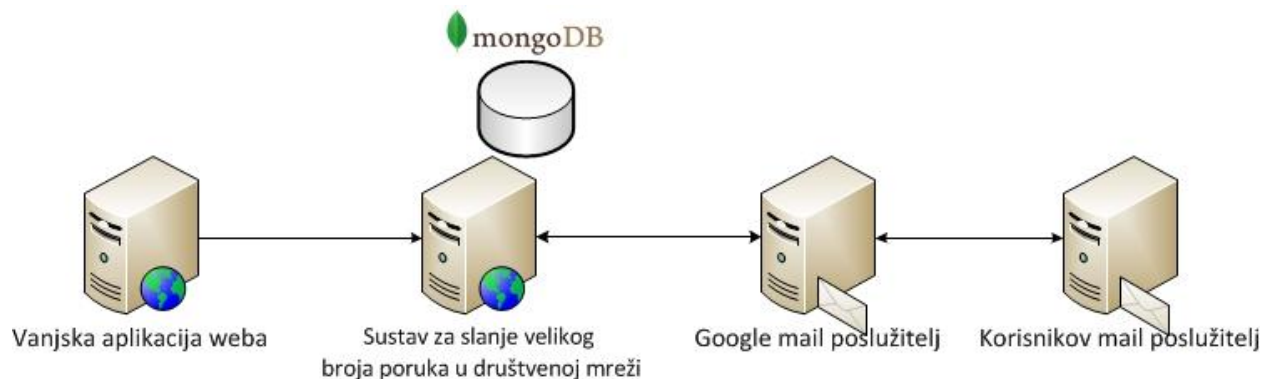
Sustav za slanje velikog broja poruka u društvenoj mreži prikazana je na slici 21. Sustav se sastoji od poslužitelja Apache weba te nerelacijske baze podataka. Sustav koristi Google mail poslužitelj za slanje e-mailova korisnicima.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.



Slika 21 Arhitektura sustava

Slika 22 prikazuje sustav za slanje velikog broja poruka u društvenoj mreži u interakciji sa vanjskim komponentama za čije je korištenje sustav i inicijalno napravljen. Sustav za slanje velikog broja poruka u društvenoj mreži nalazi se u sredini zainteresiranih strana: vanjske aplikacije weba kojoj je cilj poslati obavijesti te korisnika aplikacije weba koji očekuje primanje različitih obavijesti od vanjske aplikacije. Sustav preko računa otvorenog na Gmailu šalje obavijesti korisnicima. Sustav se temelji na tehnologijama weba čime je omogućeno korištenje sustava neovisno o programskom jeziku u kojem su pisane vanjske aplikacije weba.



Slika 22 Sustav i vanjske komponente koje sudjeluju u komunikaciji

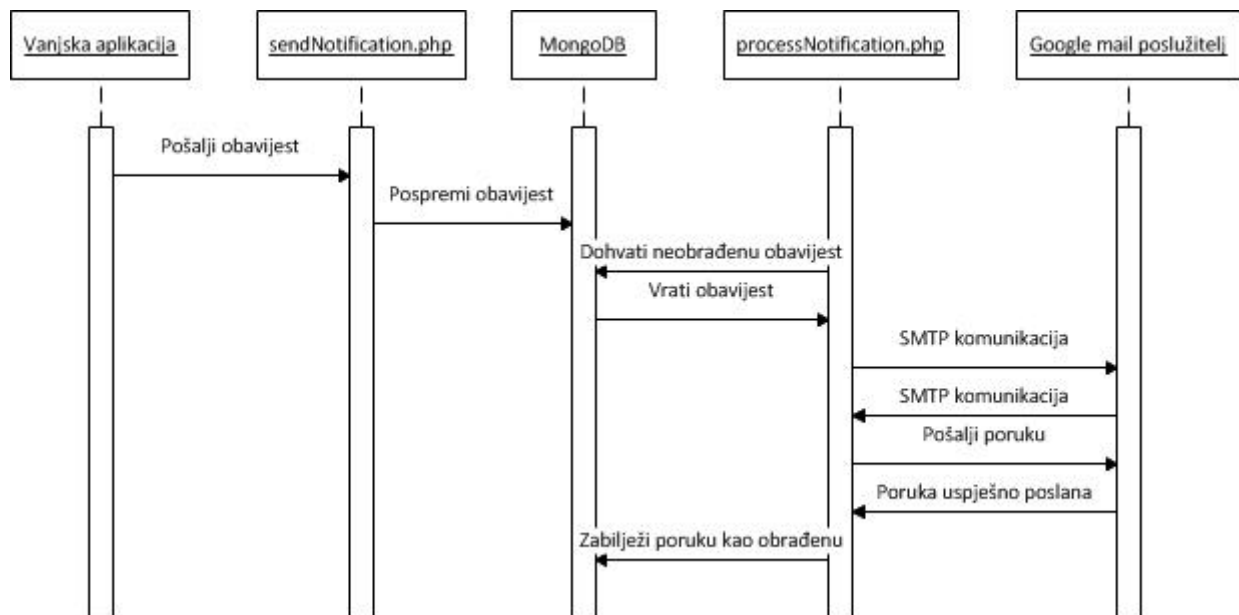
Sustav za slanje velikog broja poruka u društvenoj mreži sastoji se dvije komponente koje sudjeluju u komunikaciji sa vanjskim komponentama:

- skripte `sendNotification.php` koja služi kao vanjsko sučelje preko kojeg vanjske aplikacije šalju obavijesti u sustav i,

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

- skripte `processNotification.php` koja omogućuje obradu obavijesti te slanje obavijesti korisnicima putem e-maila.

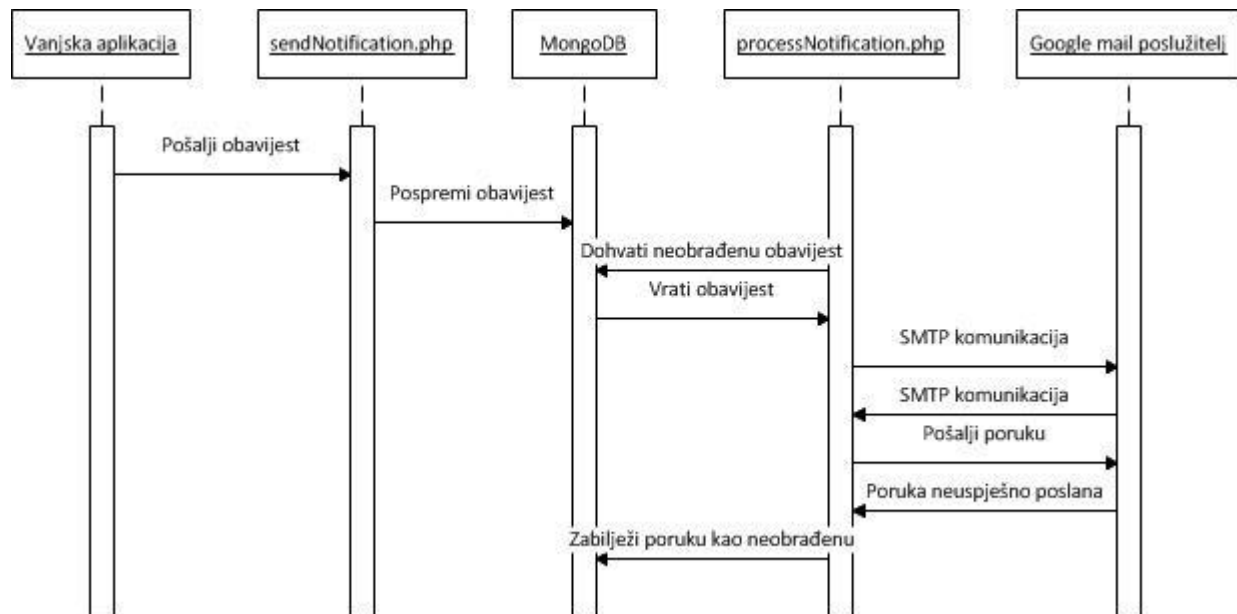
Proces komunikacije između sustava i vanjskih komponenti koji sudjeluju u komunikaciji prikazani su na slikama 23 i 24. Na slici 23 prikazan je tijek komunikacije između entiteta koji sudjeluju u komunikaciji prilikom uspješnog slanja obavijesti korisniku. Vanjska aplikacija koja koristi sustav šalje obavijest u sustav pozivajući skriptu `sendNotification.php` koja sprema obavijest u bazu podataka. Skripta `processNotification.php` svakih nekoliko sekundi pristupa bazi podataka te s upitom za neobrađenim porukama. Ukoliko obavijest postoji, skripta se spaja na Googleov mail poslužitelj koristeći SMTP protokol te šalje poruku. Googleov mail poslužitelj u ovom primjeru vraća sustavu poruku da je poruka uspješno poslana, sustav to bilježi u bazi te nastavlja sa obradom druge neobrađene obavijesti ukoliko se ona nalazi u sustavu.



Slika 23 Uspješno slanje obavijesti

Na slici 24 prikazan je tijek komunikacije između entiteta koji sudjeluju u komunikaciji prilikom neuspješnog slanja obavijesti korisniku. Tijek komunikacije je isti kao onaj prikazan na slici 23 osim zadnjeg poruka. Kada sustav dobije poruku da je poruka neuspješno poslana, sustav označuje poruku kao neobrađenu te ju obrađuje nakon što prođe pol sada od njenog spremanja u sustav. Sustav nastavlja sa dohvaćanjem neobrađenih poruka te njihovom obradom.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.



Slika 24 Neuspješno slanje obavijesti

Iz prethodna dva dijagrama vidljivo je da je ključna komponenta sustava baza podataka Mongo koja služi kao svojevrsan posrednik između skripti koje omogućuju komunikaciju sa vanjskim komponentama. Ovakva arhitektura sustava je napravljena kako bi se omogućila neovisnost sustava zaduženog za primitak obavijesti od sustava zaduženog za obradu poruka. Mongo je nerelacijska baza te je zbog arhitekture sustava ona bila odabrana zbog svojih dobrih performansi koje omogućuju brzo dohvaćanje i spremanje dokumenata u bazu podataka.

Programska implementacija sustava

Sustav za slanje velikog broja poruka u društvenoj mreži napisan je u programskom jeziku PHP te je postavljen na Apache poslužitelj weba. Sustav koristi bazu podatak Mongo za spremanje podataka. Za potrebe izrade sustava u bazi podataka dizajnirane su 4 kolekcije (analogno tablicama u relacijskim bazama podataka):

- `Notification` – služi za spremanje obavijesti koje dođu u sustav
- `UnprocessedNotifications` – služi za spremanje ključeva neobrađenih obavijesti pohranjenih u kolekciji `Notification`
- `ProcessedNotifications` – služi za spremanje ključeva obrađenih obavijesti

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

pohranjenih u kolekciji `Notification`

- `ErroredNotifications` – služi za spremanje ključeva obavijesti pohranjenih u kolekciji `Notification` koje nisu mogle biti poslane.

Ispis 1 prikazuje strukturu dokumenta pohranjenih u kolekciji `Notification`. Ovo je prikaz standardnog dokumenta koji je u svojoj obradi doživio grešku pri slanju. Iz naziva polja vidljivo njihovo značenje: polje `notificationType` označava tip obavijesti, `attempts` broj pokušaja slanja obavijesti, `recipient` primatelja itd.

```
[_id] => ObjectId Object (
  [$id] => 4ee5f5608ead0e8c1800000e
)
[attempts] => 6
[errorMessage] => Greška kod slanja e-maila
[from] => igor.vlahek@fer.hr
[message] => Bok, ovo je probni e-mail
[notificationType] => eMail
[recipient] => igorvlahek1@gmail.com
[storedTime] => 12-12-2011 13:36:48
[subject] => probni subject
[timeAsInt] => 1323693408
```

Ispis 1. Struktura dokumenta pohranjenih u kolekciji `Notification`

Ispis 2 prikazuje strukturu dokumenta pohranjenog u kolekcijama `UnprocessedNotification`, `ProcessedNotification`, `ErroredNotification`. Ove kolekcije sadrže samo jedan dokument koji se prilikom svakog dolaska, obrade poruke mijenja. Kolekcije sadrže samo polje ključeva (`MongoId Object`) obavijesti koje su došle u sustav. Prilikom dolaska obavijesti u sustav one se stavljaju u kolekciju `Notification` dok se ključ obavijesti stavlja u dokument kolekcije `UnprocessedNotification`. Nakon završetka obrade ključ dokumenta se prebacuje u dokument kolekcije `ProcessedNotification` ili u dokument `ErroredNotification`. Ovakvom strukturom kolekcija u bazi podataka omogućili smo red čekanja pomoću baze podataka Mongo te smanjenu ovisnost komponenti sustava.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

```

[_id] => MongoId Object (
  [$id] => 4ee5e6ed8ead0e9018000003
)
[id] => Array (
  [0] => MongoId Object (
    [$id] => 4ee5e8638ead0e8c18000001
  )
  [1] => MongoId Object (
    [$id] => 4ee5e8658ead0e8c18000003
  )
  [2] => MongoId Object (
    [$id] => 4ee5e7ba8ead0e9018000009
  )
)
[name] => processed

```

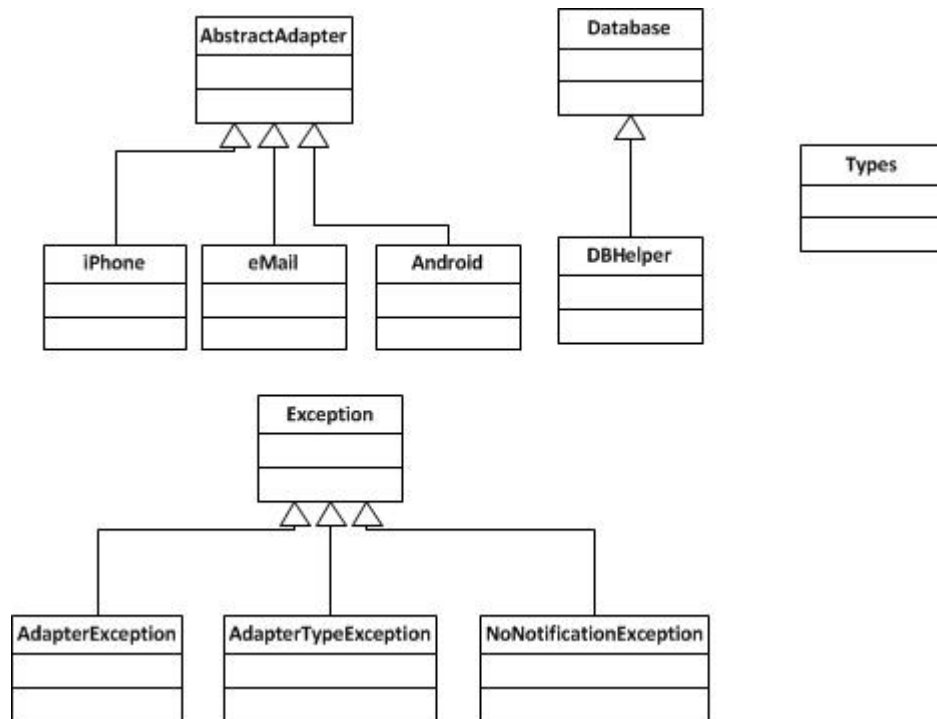
Ispis 2. Struktura dokumenta pohranjenog u kolekcijama `UnprocessedNotification`, `ProcessedNotification`, `ErroredNotification`

Kao što je ranije napomenuto, sustav je napisan u programskom jeziku PHP. Sustav se sastoji od dvije komponente koje sudjeluju u komunikaciji sa vanjskim komponentama:

- skripte `sendNotification.php` koja služi kao vanjsko sučelje preko kojeg vanjske aplikacije šalju obavijesti u sustav i,
- skripte `processNotification.php` koja omogućuje obradu obavijesti te slanje obavijesti korisnicima putem e-maila.

Sustav je oblikovan prateći načela objektnog oblikovanja. Skladno tome dvije prethodno navedene skripte koriste objekte prilikom rada sustava. Slika 14 prikazuje dijagram razreda korištenih za izradu sustava. U razredu `eMail` nalazi se logika za manipulaciju obavijesti koje su tipa e-mail. Razred implementira logiku za spremanje obavijesti u bazu podataka te za komunikaciju sa mail serverom. Razred `iPhone` i `Android` služe za simulaciju slanja obavijesti na iPhone i Android uređaje. Razred implementira logiku za spremanje obavijesti u bazu podataka, ali i ne logiku za stvarno slanje obavijesti. Razred `DBHelper` naslijeđuje razred `Database`. Razred `Database` sadrži osnovne metode za manipulaciju podataka u bazi podataka, dok razred `DBHelper` sadrži specifične metode koje omogućuju manipulaciju podatka specifičnih za ovaj sustav. Razred `Types` sadrži konstante koje predstavljaju tip obavijesti. Razredi `AdapterException`, `AdapterTypeException`, `NoNotificationException` naslijeđuju razred `Exception`. Prije navedeni razredi su iznimke koje sustav „baca“ prilikom određene nepoželjne situacije u sustavu izazvane vanjskim utjecajima.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.



Slika 25 Dijagram razreda sustava

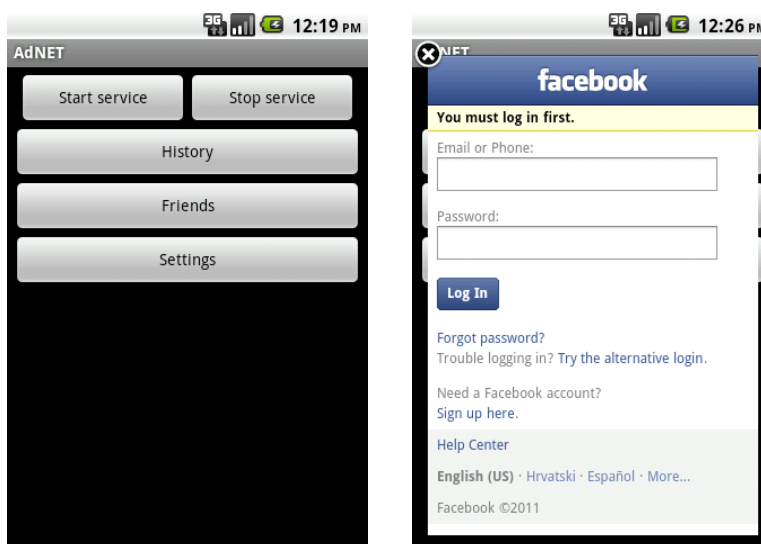
CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

3. Upute za korištenje

U nastavku slijede upute za korištenje pojedine usluge.

3.1 Upute za korištenje sustava AdNET

Nakon instalacije, korisnik pokreće aplikaciju s izbornika. Početni ekran prikazan je na slici 26. Prije korištenja korisnik mora pokrenuti uslugu i prijaviti se na sustav. To čini odabirom opcije „*Start service*“. Obrazac za registraciju prikazan je na slici 26.



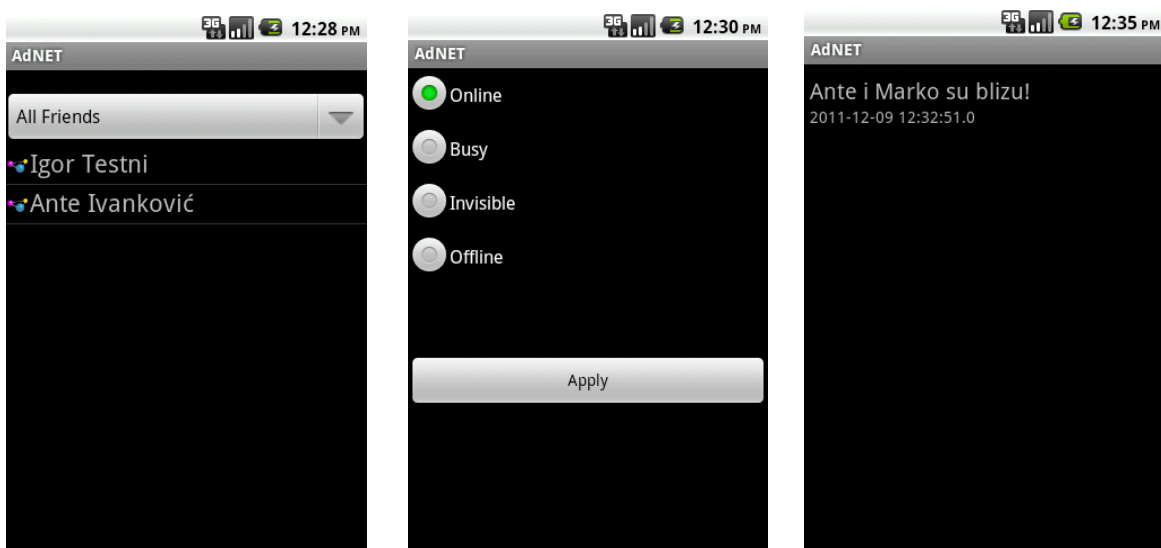
Slika 26. Početni ekran (lijevo), ekran za registraciju korisnika (desno)

Početni ekran omogućuje korisniku pregled svih dobivenih poruka za stvaranje ad-hoc mreže (opcija *History*), pregled svih prijatelja (opcija *Friends*) i postavljanje vlastitog statusa (opcija *Settings*). Ukoliko korisnik primjeti da je napajanje pokretnog uređaja na niskoj razini ili iz bilo kojeg razloga želi zaustaviti uslugu AdNET, može odabrati opciju *Stop service* koja zaustavlja pozadinske procese.

Na slici 27 prikazani su ekrani koji se prikazu korisniku odabirom svake pojedine opcije. Pri pregledu liste prijatelja, korisnik može odabrati nekog od korisnika i ignorirati ga (engl. *Ignore*). Na taj način, označeni korisnik se ne uzima u obzir prilikom formiranja ad-hoc društvenih veza u odnosu na korisnika koji ga je ignorirao. Korisnik također može izabrati ukoliko želi privremeno sakriti svoju dostupnost i ne primati obavijesti od AdNET sustava. Može izabrati između 4 opcije:

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

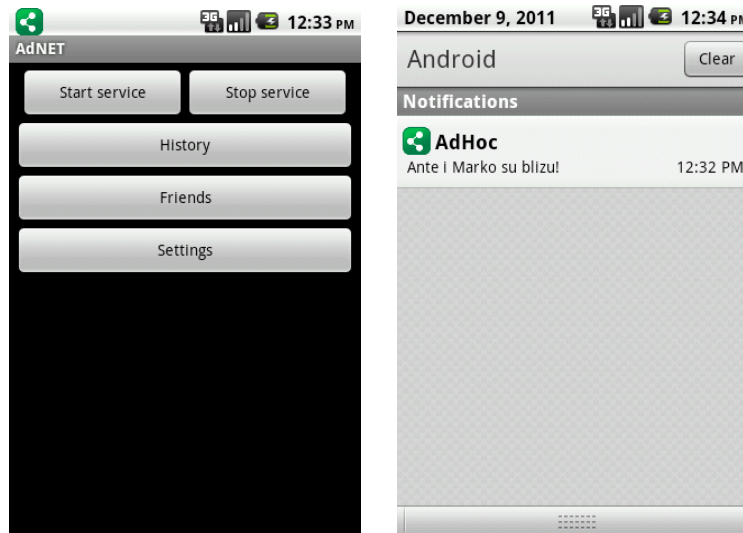
1. *Online* – korisnik je dostupan na mreži,
2. *Busy* – korisnik je zaposlen i ne želi da ga se ometa u radu stoga će biti vidljiv drugim korisnicima ali neće primiti poruke o njima,
3. *Invisible* – korisnik „vidi“ druge korisnike ali oni ne vide njega,
4. *Offline* – korisnik je isključen iz mreže.



Slika 27. Ekran za prikaz prijatelja (lijevo), postavke (sredina) i pregled svih poruka (desno)

Da bi korisnik primao obavijesti o ad-hoc društvenim vezama u odgovarajućem trenutku i na odgovarajućoj lokaciji, nužno je da su odgovarajući korisnici potvrdili svoj dolazak na neki događaj preko društvene mreže Facebook. Kada se poklopi vrijeme i lokacija korisnika, oni će dobiti obavijest na način koji je prikazan na slici 28 bez obzira da li je upaljena aplikacija AdNET ili ne.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.



Slika 28. Primanje obavijesti o ad-hoc mreži

3.2 Upute za korištenje usluge za određivanje najbližih prijatelja na društvenoj mreži Facebook

Sustavu za određivanje najbližih prijatelja na društvenoj mreži Facebook može se pristupiti na dva načina:

- 1) kao samostalnoj Facebook aplikaciji;
- 2) kao web-usluzi.

U skladu s time, u nastavku su objašnjena oba načina pristupanja.

1) Pristupanje kao samostalnoj Facebook aplikaciji:

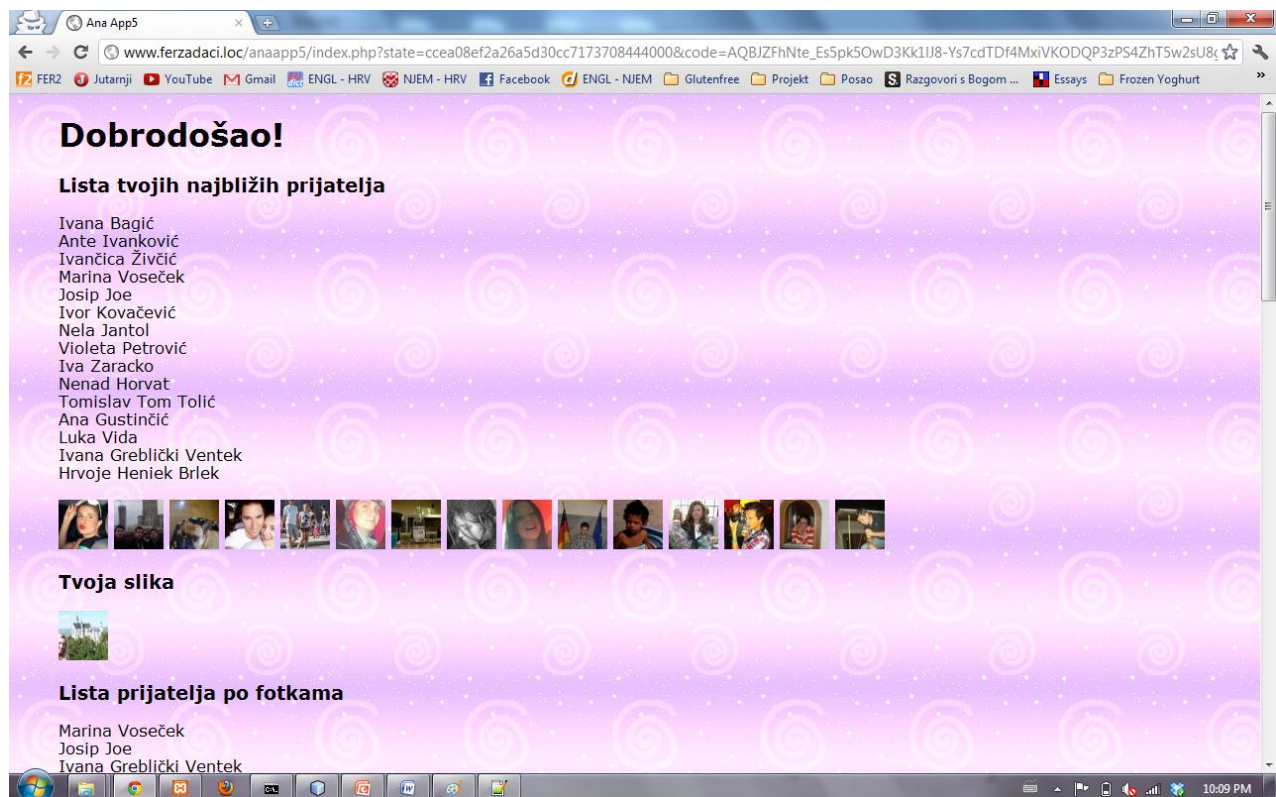
- učiniti *Login* na društvenu mrežu Facebook,
- u glavnoj traci za pretraživanje pronaći aplikaciju pod nazivom Ana App5,
- pristupiti aplikaciji,
- kliknuti *Allow* na sve tražene dozvole (engl. *permission*) ili samo na dozvole po želji,
- pričekati vrijeme potrebno za obradu podataka i pripremu prikaza,

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

- vidjeti prikaz aplikacije i uživati u njemu s najdražim prijateljima.

2) Pristupanje kao web-usluzi:

- pristupiti web-usluzi na adresi <http://www.ferzadaci.loc/anaapp5/> te u URL-u (*Uniform Resource Locator*) prenijeti podatke: `UserId` i `Access Token`,
- kliknuti *Enter*,
- vidjeti prikaz rezultata u obliku JSON objekta.



Slika: Pristupanje usluzi za određivanje najbližih prijatelja kao Facebook aplikaciji

Usluga za određivanje najbližih prijatelja na društvenoj mreži Facebook vrlo je jednostavna za korištenje, te može biti korisna pri razvijanju bilo koje složenije i moderne Facebook aplikacije. Izazovi pri razvoju ove usluge sastojali su se ponajviše u vremenu potrebnom za dohvat i obradu podataka s društvene mreže Facebook. Ovaj se izazov može riješiti pokretanjem paralelnih upita nad društvenom

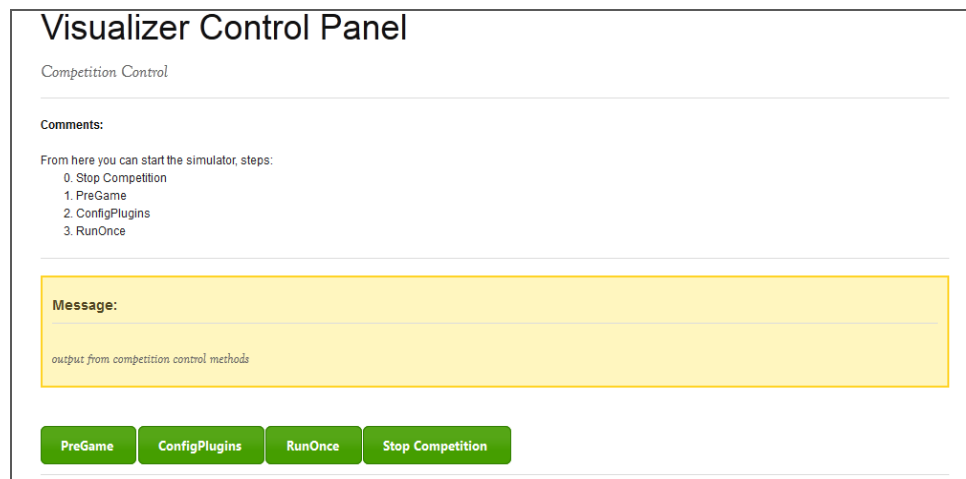
CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

mrežom Facebook te efikasnijom obradom podataka u već postojećoj i stabilnoj strukturi kolekcija u bazi podataka Mongo.

3.3 Upute za korištenje vizualizacije PowerTAC natjecanja

Visualizer je web-aplikacija zbog čega je namijenjen pokretanju iz internetskog preglednika. Nakon što se *Visualizer* postavi i pokrene na odgovarajućem poslužitelju poput *Tomcata*, *Glassfisha* ili drugi, pomoću internetskog preglednika valja otvoriti poveznicu na kojoj se nalazi *Visualizer*.

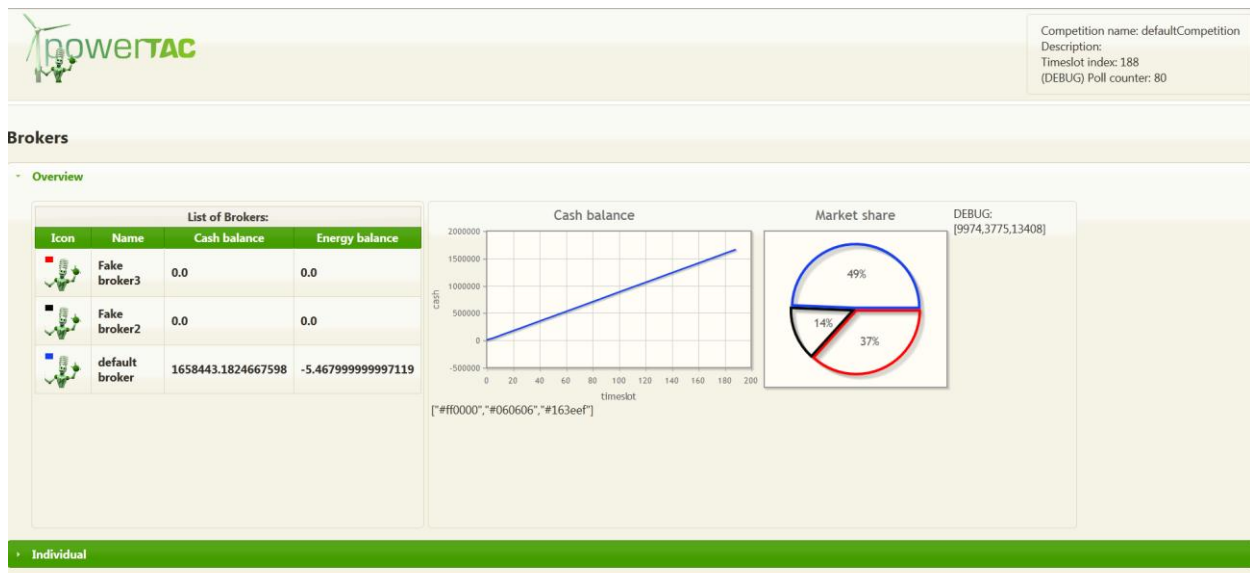
Konkretno, *Visualizer* nudi dvije osnovne funkcionalnosti: administraciju natjecanja i vizualizaciju natjecanja. Administraciju natjecanja moguće je vidjeti na slici (Slika 29). Klikom na *PreGame* gumb web-aplikacija stvara novu instancu natjecanja. *ConfigPlugins* omogućava konfiguraciju simulatora. Konačno, za pokretanje simulatora dovoljno je pritisnuti *RunOnce*. Zahtjev za zaustavljanje natjecanja šalje se klikom na *Stop Competition*.



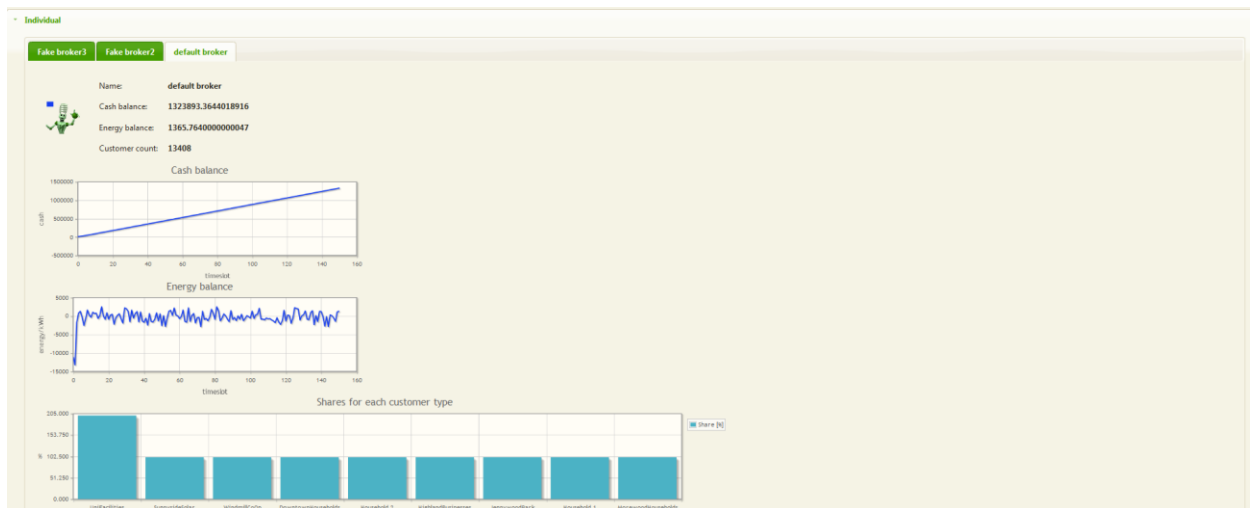
Slika 29: Administracija natjecanja

Na slikama Slika 30 i Slika 31 prikazana je vizualizacija natjecanja. Preko nje korisnik može ovisno o vlastitim željama konfigurirati *Visualizer* kako bi dobio najvažnije informacije.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.



Slika 30: Općeniti pregled svih agenata.



Slika 31: Detaljniji pregled odabranog brokera.

U skup podataka koji Visualizer prikazuje spadaju informacije o pojedinom agentu, podaci o potrošačima energije, podaci o vremenskoj prognozi, financijski dijagrami i druge korisne informacije.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

3.4 Upute za korištenje sustava za upravljanje korisničkim profilima

Obzirom da sustav u trenutnoj fazi nije namijenjen radu sa korisnicima nije potrebno davati značajnije upute. Jedino što se od korisnika zahtijeva je da se pri posjeti web-stanice usluge spoji na Facebook društvenu mrežu (Slika 31) da bi sustav mogao dohvatiti potrebne podatke sa semantičku obradu (Slika 32).



Slika 32. Traženje od korisnika da se spoji na mrežu Facebook



Slika 33. Sustav nakon spajanja na mrežu Facebook

Na slici (Slika 32) prikazan je izgled sustava nakon spajanja korisnika na Facebook. Spajanje se vrši preko standardnog sučelja društvene mreže Facebook.

3.5 Upute za korištenje sustava za slanje velikog broja poruka u društvenoj mreži

Sustavu za slanje velikog broja poruka u društvenoj mreži pristupa se pozivanjem skripte `sendNotification.php`. Skripta prima putem URL-a parametre potrebne za slanje obavijesti na obradu u sustav. Parametri potrebni za slanje obavijesti u obliku e-maila jesu:

- `notificationType` – tip obavijesti, ukoliko želimo poslati obavijest u obliku e-maila,

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

parametar `notificationType` mora biti postavljen `eMail`, inače će se dogoditi greška,

- `subject` – subjekt koji će biti specificiran prilikom slanja obavijesti na korisnikov e-mail,
- `message` – poruka koja se šalje u e-mailu,
- `recipient` – primatelj obavijesti,
- `from` – specificira pošiljatelja e-maila.

Primjer URL-a koji vanjska aplikacija weba poziva jest:

```
http://www.ferzadaci.loc/igor/sendNotification.php?notificationType=eMail&subject=probni%20subject&message=Bok%20kaj%20ima%20kak%20si&recipient=igorvlahek1@gmail.com&from=igor.vlahek@fer.hr.
```

Pozivanjem ovog URL-a poslat će se obavijest u obliku e-maila na mail igorvlahek1@gmail.com, pošiljatelj će biti igor.vlahek@fer.hr, subjekt e-maila će biti „*probni subject*“, poruka koju prenosimo u e-maila će biti „*Bok kaj ima kak si*“. Pokretanjem skripte `processNotification.php` obradit će se sve neobrađene obavijesti u sustavu.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

4. Opis korištenih tehnologija

U nastavku su opisane bitne karakteristike korištenih tehnologija prilikom realizacije projekta.

4.1 Baza podataka Mongo

Baza podataka Mongo jest dokumentno orijentirana baza podataka napisana u C++. Bazu podataka karakteriziraju:

- dokumentno orijentirano spremanje podataka u obliku JSON-a čije dinamičke sheme omogućuju jednostavnost,
- podrška za indeksiranje čime se ubrzava pretraživanje podataka,
- brzo pretraživanje podataka,
- brza manipulacija podataka,
- agregacija podataka.

Za razliku od relacijskih baza podataka, baza podataka Mongo nema striktnu strukturu podataka. Umjesto tablica, u Mongu postoje kolekcije. U kolekcije se spremaju dokumenti čija struktura međusobno ne mora biti ista. Mongo također omogućuje dinamičko mijenjanje strukture dokumenata unutar kolekcije. Bitno je da svaki dokument ima jedinstveni ključ koji će ga razlikovati od ostalih. Ovakvim načinom pohrane podataka ubrzava se pretraživanje, dok se problem različitih struktura podataka mora rješavati na aplikacijskom sloju jer, kao što je napomenuto, struktura dokumenata unutar jedne kolekcije ne mora biti jednaka. Bitno je napomenuti da Mongo ne koristi SQL za potrebe manipulacije podataka u bazi podataka, već koristi Mongo dokumentno-orijentirani jezik za manipulaciju podataka.

Za omogućavanje pristupa bazi Mongo putem PHP-a potrebno je (pretpostavka je se koristi programski paket XAMPP):

1. skinuti bazu podataka Mongo sa adrese: <http://www.mongodb.org/downloads>,
2. otpakirati Mongo u mapu prema izboru (npr. *D:\work\programs\mongo*),
3. Napraviti mapu u koji će se spremati podaci (npr. *D:\work\db*),

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

4. Skinuti PHP adapter za Mongo s <https://github.com/mongodb/mongo-php-driver/downloads> i spremiti u `D\xampp\php\ext` (mapa u kojoj se nalazi XAMPP),
5. U `php.ini` (`D:\xampp\php\php.ini`) dodati `extension=php_mongo.dll`,
6. Ponovno pokrenuti *Apache*,
7. Napraviti skriptu `start_mongo.bat` kojom ćemo pokretati bazu podataka Mongo. „`D:\work\programs\mongo\bin\mongod.exe --dbpath D:\work\db`“. (prvi parametar jest put do direktorija gdje ste unzipali bazu podataka Mongo, a drugi parametar jest mapa koja će se koristiti kao spremište podataka).

4.2 Sustav za *push*-poruke na operacijskom sustavu Android

Push-poruke na operacijskom sustavu Android realiziraju se pomoću C2DM (*Cloud to Device Messaging*) tehnologije. To je implementacija koja omogućava malu potrošnju energije, skalabilnost i brzinu takve usluge. Da bi se usluga mogla koristiti potrebno je:

1. Registrirati aplikaciju za ovu uslugu na web-stranici <http://code.google.com/android/c2dm/signup.html>. Potrebno je čekati do dva dana da administratori provedu novu registraciju,
2. U manifestu aplikacije za Android potrebno je registrirati razrede koji će biti zaslužni za prihvaćanje takve poruke (<http://code.google.com/android/c2dm/#manifest>),
3. Implementirati razrede za primanje poruka i razrede za manipulaciju registracijom.

Registracijska e-mail adresa zahtjeva se i u kodu klijenta i u kodu poslužitelja. U oba slučaja ona mora biti ista kao i e-mail adresa koja je korištena u procesu registracije aplikacije (točka 1).

4.3 Okvir Spring

Okvir Spring jest aplikacijski okvir otvorenog koda za Java platformu. Sadrži funkcionalnosti koje su podijeljene u otprilike 20 modula. Svaki je modul grupiran u grupe *Core Container*, *Data Access/Integration*, *Web*, *AOP (Aspect Oriented Programming)*, *Instrumentation* i *Test*.

Neke od karakterističnih funkcionalnosti su:

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

- Kontejner Inverzije Upravljanja (engl. *Inversion of Control container*) – omogućava konfiguraciju komponenti aplikacije i upravljanje životnim ciklusom Java objekata.
- Aspektno orijentirano programiranje (engl. *Aspect-oriented programming*) – inherentno ugrađena podrška za aspektno orijentirano programiranje.
- MVC (model-view-controller) okvir – temelji se na tehnologiji Servlet te omogućava proširenje i modificiranje za web-aplikacije i *RESTful* web-usluge.
- Autentikacija i autorizacija – Spring nudi sigurnosne mehanizme koji podržavaju skup standarda i alata preko *Spring Security* potprojekta.

Standardne funkcije okvira Spring moguće je koristiti u bilo kojoj Java aplikaciji, međutim, postoje određena proširenja za izgradnju web-aplikacija korištenjem platforme Java EE. Iako okvir Spring nameće poseban programski model, sve se češće koristi kao alternativa ili dodatak EJB (Enterprise JavaBean) modelu.

4.4 Knjižnica Primefaces

Primefaces jest JSF knjižnica koja se sastoji od niza bogatih komponenti koje je moguće iskoristiti za grafički dizajn u XHTML-u. Neke od pogodnosti koje ova knjižnica nudi su: bogat skup komponenata, ugrađena podrška za Ajax, dobra podrška za stiliziranje komponenata te osnovna podrška za Ajax Push.

Za korištenje ove knjižnice, u postojeći JSF projekt potrebno je dodati JAR koji se nalazi na sljedećoj poveznici: <http://www.primefaces.org/downloads.html>.

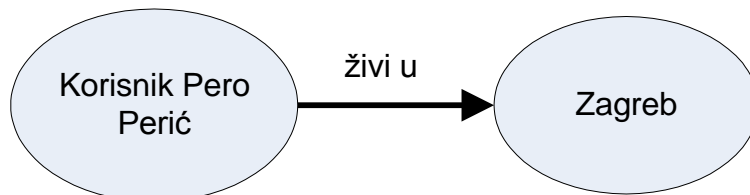
4.5 Tehnologije izgradnje semantičkih profila

Za semantičko profiliranje koriste se sljedeće tehnologije: RDF, OWL i Sesame.

RDF (engl. *Resource Description Framework*) je W3C specifikacija za modeliranje informacija i njihov zapis pomoću različitih sintaksnih formata. Omogućava zapisivanje informacija u računalu razumljivom

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

formatu odnosno u obliku izjava koje se sastoje od subjekata, predikata i objekta, a zajedno čine usmjerenih graf. Elementi RDF zapisa su najčešće resursi reprezentirani svojim URI-em. Jedna RDF konstrukcija je prikazana na slici 34..



Slika 34. RDF konstrukcija

OWL (engl. *Web Ontology Language*) jezik je za izgradnju ontologija nad RDF podatcima. Sadrži značajan vokabular za opisivanje klasa, njihovih odnosa, atributa i svojstava. Sa službene stranice je moguće preuzeti tri različite verzije OWL jezika, ovisno o potrebama i složenosti odnosa među čvorovima: *OWL Lite*, *OWL DL* i *OWL Full*. Za potrebe ovog projekta korišten je *OWL Lite*, njegova najjednostavnija varijanta. Namijenjena je korisnicima kojima je potrebna izgradnja klasifikacijske hijerarhije te ograničenja kardinalnosti ograničena na vrijednosti 0 ili 1.

Primjer jedne izjave u OWL jeziku:

```

<owl:Class rdf:ID="Croatia">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Europe"/>
  </rdfs:subClassOf>
</owl:Class>
  
```

Ovime se tvrdi da je resurs imena Croatia podklasa resursa imena Europa što konkretno znači da se upitom u semantičku bazu podataka može na temelju informacije da je netko iz Hrvatske zaključiti da je i iz Europe.

Sesame predstavlja skup klasa tj. biblioteka (tzv. *framework*) u Javi za pohranu i dohvaćanje RDF podataka [13]. Za tu svrhu podržava SPARQL i SeRQL jezike za stvaranje upita nad semantičkom bazom, sadrži i vlastiti skup funkcija (engl. *API*) za izvođenje navedenih funkcionalnosti.

Za instalaciju i zajednički rad navedenih tehnologija potrebno je instalirati *Apache Tomcat* poslužitelj te

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

u *webapps* datoteku otpakirati datoteke za Sesame 2 repozitorij i *OWL Lite* nadogradnju. Profile se može stvarati preko web sučelja, ali i putem dostupnog API-a programskim rješenjima u Javi.

CProject	Verzija: 1.0
Tehnička dokumentacija	Datum: 14.12.2011.

5. Literatura

- [1] HEVČUK I.: Ad-hoc društvena umreženost lokacijski bliskih korisnika. *Diplomski rad*. Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva, Zagreb (Hrvatska),
- [2] SMAILOVIĆ, V.: Komunikacija u ad-hoc društvenim mrežama. *Diplomski rad*. Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva, Zagreb (Hrvatska), 2011.
- [3] Baza podataka Mongo, <http://www.mongodb.org/>, [posjećeno 5.10.2011.]
- [4] VOGELLA L.: Android Cloud to Device Messaging (C2DM) - Tutorial, <http://www.vogella.de/articles/AndroidCloudToDeviceMessaging/article.html>, [posjećeno 12.10.2011.]
- [5] Baza podataka MySQL, <http://www.mysql.com/>, [posjećeno 15.9.2011.]
- [6] Aplikacijski poslužitelj Glassfish, <http://glassfish.java.net/>, [posjećeno 23.9.2011.]
- [7] Aplikacijski poslužitelj Apache, <http://httpd.apache.org>, [posjećeno 20.9.2011.]
- [8] Platforma Spring, <http://www.springsource.org>, [posjećeno 25.9.2011.]
- [9] Knjižnica Primefaces, <http://www.primefaces.org/>, [posjećeno 25.9.2011.]
- [10] JSF Technology, <http://www.oracle.com/technetwork/java/javaee/javaserverfaces-139869.html>, [posjećeno 5.10.2011.]
- [11] PHP: Hypertext Preprocessor, <http://www.php.net/>, [posjećeno 26.9.2011.]
- [12] Apache Tomcat, <http://tomcat.apache.org>, [posjećeno 25.9.2011.]
- [13] Sesame 2, <http://www.openrdf.org/>, [posjećeno 3.10.2011.]