

## Projekt: NWTiS\_2010\_2011

Sustav se treba sastojati od sljedećih aplikacija:

1. ant aplikacija ({korisnicko\_ime}\_aplikacija\_1.xml) za pripremu pojedine baze podataka (MySQL i JavaDB) koje se koriste u web aplikacijama 2 i 3. Postupak treba obuhvatiti pojedinu akciju vezanu uz rad s izabranu bazu podataka (kreiranje baze podataka, kreiranje korisničkog imena (nwtis\_{korisnickoime}) s pripadajućom lozinkom, dodjelu dozvola za rad s bazom podataka, kreiranje tablica i inicijalno punjenje podacima), sve akcije zajedno u predviđenom slijedu ili brisanje baze podataka. Za svaku bazu podataka treba parametarski definirati konfiguracijske podatke. Ova aplikacije se zadnja prikazuje na prezentaciji projekta.
2. web aplikacija ({korisnicko\_ime}\_aplikacija\_2) u pozadinskom modu tj. servletu s automatskim startanjem pokreće dretvu (konfiguracijom se određuje vremenski interval preuzimanja podataka, npr. 30 sec, 100 sec, 2 min, 10 min, 30 min, 60 min, ...) preuzima važeće meteorološke podatke (minimalno 7 meteoroloških podataka) od web servisa za izabrani skup gradova iz SAD (njihovi zip kodovi nalaze se u tablici mycities). Važeći meteorološki podaci preuzimaju se putem operacije `GetLiveWeatherByUSZipCode` s WeatherBugWebServices web servisa <http://api.wxbug.net/weatherservice.asmx> čiji je WSDL <http://api.wxbug.net/weatherservice.asmx?WSDL>. Potrebno je obaviti registraciju da bi se mogao koristiti web servis. Adresa <http://weather.weatherbug.com/desktop-weather/api.html>. Drugi zadatak web aplikacije je pružanje web servisa za prikupljene podatke kao što su: popis svih zip kodova za koje se prikupljaju podaci, x posljednjih podataka za izabrani zip code, podaci za zip code u nekom vremenskom intervalu i sl. Treći zadatak je vidljivi dio web aplikacije odnosno korisnički dio koji treba sadržavati pregled prikupljenih podataka putem JSP i displaytag-a uz filtriranje (izabrani zip kod, vremenski interval u kojem su prikupljeni podaci i sl.) i straničenje. Korisnički dio web aplikacije treba biti zaštićen putem web/servlet kontejnera na bazi obrasca/forme za prijavljivanje uz pomoć JDBC pristupa do baze podataka te osiguranja sigurnog kanala (SSL uz certifikat s imenom i prezimenom studenta). Četvrti zadatak je vođenje dnevnika zahtjeva na bazi filtera uz zapis trajanje obrade zahtjeva u bazu podataka. U tablici prikupljenih meteoroloških podataka trebaju biti podaci za minimalno 20 gradova i za svaki od njih minimalno 100 preuzetih podataka u intervalu od većem od 6 sati. Web aplikacija izvršava se na Tomcat-u.
3. web aplikacija ({korisnicko\_ime}\_aplikacija\_3) koja u pozadinskom modu tj. servletu s automatskim startanjem pokreće dretvu (konfiguracijom se određuje vremenski interval rada, npr. 5 sec, 20 sec, 100 sec, ...) koja provjerava u poštanskom sandučiću (adresa poslužitelja, korisničko ime i lozinka definiraju se

u konfiguracijskoj datoteci) pristiglu poštu. Od pristiglih email poruka u obzir dolaze samo one koje imaju Subject: NWTiS i MIME tip „text/plain“. Te poruke nazivamo NWTiS porukama. U sadržaju NWTiS poruke mogu se nalaziti komande:

Komanda	Opis	Vrsta
newUser('pero', 'Pero', 'Kos', '123456');	kreiranje korisnika pero, s imenom Pero, prezimenom Kos, lozinkom 123456	0
user('pero', '123456');	prijavljivanje korisnika pero s lozinkom 123456	1
data();	slanje posljednjih meteoroških podataka gradova prijavljenog korisnika putem emaila	2
data (10451, 7);	slanje posljednjih 7 meteoroških podataka za grad sa zip code-om 10451 prijavljenom korisniku putem emaila	3
data ('2010-02-01 00:00:00', '2010-02-01 08:00:00');	slanje meteoroških podataka za gradove prijavljenog korisnika zadani interval putem emaila	4
forecast (10451, 5);	slanje podataka meteoroške prognoze za sljedeći dan za grad prijavljenog korisnika. Poruke se šalju sljedećih 5 dana putem JMS poruke. Poruka se šalje jednom dnevnom u prvom ciklusu koji je nakon vremena zaprimanje zahtjeva. Svako slanje poruke se naplaćuje pa treba provjeriti da li postoji dovoljno sredstava za ispunjenje cijelog zahtjeva. To je posao 2. zadatka.	5
stopforecast (10451);	prestanak slanje podataka meteoroške prognoze za grad prijavljenog korisnika	6

Kod kreiranje korisničkog računa pamti se i email adresa pošiljatelja koji je to zatražio. U email poruci smije biti samo jedna komanda vrste 0. U ostalim slučajevima email poruka treba sadržavati dvije komande, od kojih je prva vrste 1, a zatim neka od komandi vrste 2 do 6. Kod komande vrste 1 aplikacija treba provjeriti autentičnost korisnika te da li email adresa pošiljatelja odgovara zaspisanoj adresi. Ukoliko sadržaj NWTiS poruke nije ispravan odnosno provjera autentičnosti nije uspješna, aplikacija šalje korisniku email poruku s opisom problema. NWTiS poruke koje su ispravne mogu se obrađivati. Za njih aplikacija šalje određeni zahtjev web servisu iz druge aplikacije. Nakon prijema podataka, aplikacija šalje u email poruci tražene podatke u text/html formatu. Za svaku obrađenu NWTiS poruku tj. zahtjev korisnika, potrebno je upisati podatke u bazu podataka. Obrađene NWTiS poruke treba spremi u tablicu baze podataka, s time da se može razlikovati vrsta poruke (NWTiS poruka, ostale poruke), a zatim obrisati iz sandučića. Ostale ne-NWTiS poruke treba označiti da su pročitane. Za svaku iteraciju obrade email poruka treba zapisati u dnevnik rada podatke o vremenu početka i završetka rada, broju pročitanih poruka, broju NWTiS poruka, broju ispravnih NWTiS poruka i sl.

Drugi zadatak web aplikacije odvija se u pozadinskom modu tj. servletu s automatskim startanjem pokreće dretvu (konfiguracijom se određuje vremenski interval rada, npr. 5 sec, 20 sec, 100 sec, ...) koja šalje JMS poruke za zahtjeve za prognozom vremena.

Treći zadatak je vidljivi dio web aplikacije odnosno korisnički dio, koji treba realizirati putem JSF (facelets) uz minimalno dvojezičnu varijantu (hrvatski i engleski jezik). To znači da svi statički tekstovi u pogledima trebaju biti označeni kao „labele“ i dobiti jezične prijevode. Jezik se odabire na početnoj stranici aplikacije. Prijavljeni korisnik planira putovanje te određuje redoslijed gradova koji će biti na njegovom putovanju i za koje će primiti podatke iz skupa gradova za koje se prikupljaju podaci (njihov popis dobije se putem web servisa iz druge aplikacije). Kod pregleda izabranih gradova za putovanje i njihovih podataka potrebno je koristiti Google Maps JavaScript API V3.

Google Maps JavaScript API V3 - <http://code.google.com/intl/hr-HR/apis/maps/documentation/javascript/>

<http://code.google.com/intl/hr-HR/apis/maps/documentation/javascript/tutorial.html> (tutorial)

<http://code.google.com/intl/hr-HR/apis/maps/documentation/javascript/examples/index.html> (primjeri)

<http://code.google.com/intl/hr-HR/apis/maps/documentation/javascript/examples/infowindow-simple.html>  
(informacija o zip kodu i trenutnoj meteorološkim podacima izabranog zip koda uz AJAX osvježavanje podataka s intervalom od x sekundi)

<http://code.google.com/intl/hr-HR/apis/maps/documentation/javascript/examples/circle-simple.html>  
(informacija o trenutnim prognozi temperaturama )

<http://code.google.com/intl/hr-HR/apis/maps/documentation/javascript/examples/polyline-simple.html>  
(informacija o putovanju po dijelu izabranih zip kodova)

Korisnik može pregledavati povijesne meteorološke podatke za svoje izabrane gradove uz filtriranje (interval, grad i sl) i straničenje. Korisnik može pregledavati podatke meteorološke prognoze za svoje izabrane gradove (operacija `GetForecastByUSZipCode` s WeatherBugWebServices web servisa). Korisnik može pregledavati svoje email zahtjeve uz filtriranje (interval, vrsta komande i sl.) i straničenje. Pregledi trebaju biti međusobno povezani vezama tako da se iz jednog pregleda izborom (npr. klik na vezu za prikazani grad) može prelaziti na drugi pregled. Korisnički dio web aplikacije treba biti zaštićen filterom tako da ga mogu koristiti samo prijavljeni korisnici.

Četvrti zadatak je pružanje popisa aktivnih (trenutno prijavljenih) Web korisnika, aktivnosti korisnika u intervalu (potrebna autentikacija) putem RESTful web servisa.

Pristup do podataka u bazi podataka treba biti realiziran putem ORM-a tj. putem session i entity bean-ova. Web aplikacija izvršava se na GrassFish-u i treba se temeljiti na EE6 osobinama (filteri, slušači, startanje servleta i sl.).

4. web aplikacija ({korisnicko\_ime}\_aplikacija\_4) s JSF (facelets) koja preuzima JMS poruke (kao MessageDriven Bean) za pregled podataka meteorološke prognoze. Primljene poruke se spremaju u kontekst aplikacije. Ako aplikacija prestaje s radom, potrebno je poruke serializirati na vanjski spremnik (izabrana datoteka u WEB-INF direktoriju). Kada se aplikacija podiže potrebno je učitati serializirane poruke u kontekst aplikacije. Korisnik se treba prijaviti te nakon toga može pregledati vlastite poruke. Korisnik može brisati sve svoje poruke. Korisnički dio web aplikacije treba biti zaštićen filterom tako da ga mogu koristiti samo prijavljeni korisnici. Web aplikacija izvršava se na GrassFish-u i treba se temeljiti na EE6 osobinama (filteri, slušači, startanje servleta i sl.).
5. stolna aplikacija ({korisnicko\_ime}\_aplikacija\_5) poziva RESTful Web servis za ispis aktivnih korisnika treće aplikacije, za ispis vlastitih aktivnosti u intervalu. Aplikacija mora biti potpisana digitalnim potpisom autora, pripremljena u obliku izvršne .jar datoteke te se mora izvršavati uz uključivanje SecurityManager-a i pridruženu datoteku sigurnosne politike (ne smije se koristiti dozvola tipa AllPermissions). Aplikacija se mora izvršavati s komandne linije.

Sustav se temelji na dvije baze podataka sa sljedećim osobinama:

1. MySQL baza podataka (nwtis\_{korisnickoime}\_bp\_1) za drugu aplikaciju, treba sadržavati tablice: mycities i ostale koje su potrebne za rad.
2. Java DB – Derby baza podataka (nwtis\_{{korisnickoime}}\_bp\_2) za treću i **četrtu** aplikaciju, treba sadržavati tablice: cities, counties, states, zip\_codes i ostale koje su potrebne za rad.

Projektna dokumentacija (dokumentacija.html koja je dostupna iz obje web aplikacije) sadrži opis projektnog zadatka i glavne odrednice projektnog rješenja. Potrebna je Java doc dokumentacija. Nije potrebna korisnička dokumentacija