

Upute za 1. laboratorijske vježbe

Fakultet elektrotehnike i računarstva, Sveučilište u Zagrebu
Računarstvo zasnovano na uslugama

3. studenog 2014.

Zadatak

Vaš zadatak je osmisлити, oblikovati i ostvariti jednostavnu web uslugu koja izlaže sučelje koristeći REST načela. Web uslugu možete sami osmisлити ili možete kopirati neku od popularnih web usluga ili aplikacija koje često koristite (npr. Twitter¹ ili Flickr²). Bitno je istaknuti da naglasak ove laboratorijske vježbe nije na količini mogućnosti izgrađene web usluge! Jedine obavezne komponente su REST sučelje, autentifikacija korisnika, automatsko ispitivanje i generiranje dnevnika. Nije potrebno ostvariti bogato korisničko sučelje, bazu podataka (iako je neki oblik perzistencije podataka koristan, npr. SQLite) i administracijsko sučelje.

Izgrađena usluga mora imati barem dvije grupe sredstava koji se preslikavaju na URL-ove. Npr. web usluga za prikazivanje fotografija mora imati (barem) jedan URL za fotografije (npr. `/photos/`) i URL za korisnike (npr. `/users/`). Dodatno, usluga mora omogućiti dohvat svih fotografija određenog korisnika (npr. `/users/1/photos`). Za svako izloženo sredstvo ostvariti **sve** CRUD metode! Programsko sučelje mora biti dostupno kao poddirektorij (npr. `http://localhost/api`). Navedeni korijenski direktorij mora sadržavati dokumentaciju s popisom svih CRUD metoda za sve metode programskog sučelja! Sadržaj i izgled dokumentacije su proizvoljni.

Za potrebe izgradnje web usluge dozvoljeno je koristiti bilo koji programski jezik i platformu. Neki od popularnijih radnih okvira za izgradnju web usluga/aplikacija su Django/Python, Ruby On Rails/Ruby, CodeIgniter/PHP, node.js/Javascript itd. Nije dozvoljeno korištenje naprednih sustava za upravljanje sadržajem (CMS) - cilj je samostalno ostvarenje web usluge!

¹<http://twitter.com>

²<http://flickr.com>

Automatsko ispitivanje funkcionalnosti

Izgrađena web usluga mora biti funkcionalna! Uslugu ispitivati koristeći neki od alata za automatsko ispitivanje ugrađenih u radni okvir koji koristite (proučite samostalno dokumentaciju). Alternativno, moguće je koristiti Selenium (<http://seleniumhq.org/>) ili curl (<http://curl.haxx.se/>).

Napomena: ispitivanje mora biti **automatizirano**! Ako koristite neki od radnih okvira za testiranje, automatiziranje je podržano i može se relativno jednostavno ostvariti. Curl naredbe možete objediniti u skriptu (npr. `bash` skripta).

Za svako izloženo sredstvo napisati barem jedan ispitni slučaj koji omogućava provjeru ispravnosti usluge nakon npr. mijenjanja programskog koda i nakon postavljanja usluge na novi poslužitelj. Ukupan broj slučajeva i oblik pojedinog ispitnog slučaja jest proizvoljan i ovisi o programskom (curl) i grafičkom (Selenium) sučelju vaše usluge. Dobro bi bilo napraviti nekoliko zasebnih grupa ispitnih slučajeva koje demonstriraju (ispituju) funkcionalnost nekog segmenta web usluge, npr. stvaranje, dohvaćanje, brisanje i pretraživanje objekata. Primjeri korištenja:

curl

1. Ispitivanje slanja statusa na društvenu mrežu:

```
curl -u username:password -d status="test" http://test-web-usluga/status
```

2. Ispitivanje web obrasca:

```
curl http://test-web-usluga/obrazac -F \polje1=a" -F \polje2=b"
curl http://test-web-usluga/trazi/&polje1=a
```

Selenium

1. Upis subjekta kroz web obrazac:

```
sel.click("link=Upis")
sel.wait_for_page_to_load("30000")
sel.type("id_number", "123456")
```

```
sel.type("id_name", "test")
sel.type("id_description", "opis test")
sel.click("save_button")
sel.wait_for_page_to_load("30000")
sel.select("id_region", "label=Splitko-dalmatinska")
sel.type("id_city", "Split")
sel.type("id_postal_number", "21000")
sel.type("id_address", "test adresa 1000")
sel.type("id_phone_1", "020/111-222")
sel.type("id_email_1", "a@a.com")
sel.click("save_button")
sel.wait_for_page_to_load("30000")
sel.click("end_button")
```

Dnevnik korištenja

Dnevnik korištenja (engl. *log*) jednostavna je tekstualna datoteka u koju vaša usluga bilježi pristup svakom URL-u. Web-usluga bilježi u tekstualni dnevnik zapise oblika: `url browser`, gdje je `url` dio URL-a koji opisuje sredstvo, a `browser` identifikator web-preglednika koji je pristupio sredstvu. Primjer dnevnika prikazuje sljedeći isječak tekstualne datoteke:

```
/photo/23  firefox
/photo/23  chrome
/photo/24  chrome
/user/1    ie9
/user/2    ie9
/photo/67  chrome
```

Za sada je potrebno samo generirati dnevnik. Izgrađeni dnevnik će se koristiti u 2. laboratorijskoj vježbi.

Što je potrebno znati na labosima?

1. izgraditi, pokrenuti i opisati kako radi izgrađena web usluga i, ako bude potrebe, uvesti manje modifikacije u programski kod
2. pokrenuti automatske ispitne slučajeve i pokazati da je izgrađena usluga funkcionalna

3. po završetku automatskog ispitivanja prikazati generirane dnevnik
korištenja