# Iskustva u formiranju baze podataka Univerziteta u Beogradu na osnovu elektronskih obrazaca prikupljenih iz više izvora

Irena Odžić, *Univerzitet u Beogradu,* Jelica Protić, *Elektrotehnički fakultet*

***Sadržaj —*** Univerzitet u Beogradu, kao kompleksna ustanova sa više od 40 institucija-članica i preko 5000 zaposlenih, započeo je proces integracije podataka radi akreditacije Univerziteta, formiranja Škole doktorskih studija, pripreme podataka za registar visokoškolskih ustanova i publikovanja relevantnih informacija na zajedničkom sajtu. Proces je započet preciznim definisanjem više elektronskih formulara sa sistemom validacije u Excel-u, koji su popunjavani i prikupljani na institucijama-članicama i dostavljani Rektoratu. Sistemskom analizom obrazaca definisan je model podataka na osnovu koga je implementirana baza podataka Univerziteta u Beogradu. Pomoću parsera podaci iz formulara su dodatno verifikovani i uneti u bazu. Za pisanje parsera korišćena je Java platforma i Apache POI 3.1. biblioteka.

***Ključne reči* — formular, parsiranje, model podataka, baza podataka, informacioni sistem, Java, Excel.**

## I. Uvod

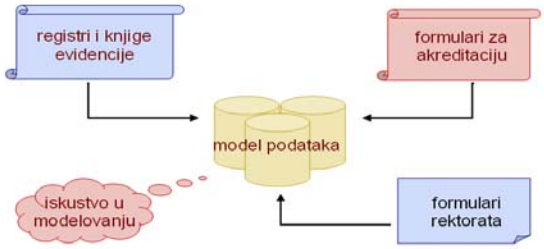
Prenošenje niza radnih procesa sa institucija članica Univerziteta na nivo Rektorata impliciralo je potrebu za prikupljanjem, obradom i prezentacijom velike količine podataka. U prvoj fazi objedinjavanja procesa na Univerzitetu prikupljani su podaci za Školu doktorskih studija, koju je trebalo organizovati u skladu sa novim akreditacionim standardima [1]. Za potrebe akreditacije Univerziteta zahtevane su detaljne informacije o samim institucijama članicama, o njihovom nastavnom i nenastavnom osoblju, studentima, studijskim programima i organizaciji nastave. Najzad, informacije o Univerzitetu u Beogradu treba da postanu dostupne u Registru visokoškolskih ustanova i da se adekvatno prezentiraju na Web sajtu Univerziteta.

Univerzitet u Beogradu sastoji se od 31 fakulteta, 11 instituta i 6 centara i zapošljava preko 5000 nastavnika i saradnika, kao i preko 2000 nenastavnog osoblja. Nastava se izvodi na osnovu 400 studijskih programa na sva tri nivoa studija. Na Univerzitetu još uvek ne postoji centralni informacioni sistem, a nivo primene informaciono-komunikacionih tehnologija po fakultetima veoma varira, kao i nivo obučenosti zaposlenih. Na osnovu analize prikazane u [3], pojedini fakulteti poseduju složene, raznorodne informacione sisteme koje su sami nezavisno razvijali [2], dok drugi jedva da imaju podatke o nastavnicima i studentima u elektronskoj formi, a informacione tehnologije koriste na sasvim skromnom nivou. Prikupljanje podataka moralo je biti obavljeno u veoma kratkom vremenskom periodu.

Iz svega je zaključeno da je kratkoročno rešenje osmišljavanje formulara koji bi bio distribuiran po institucijama, tamo popunjen i vraćen u Rektorat. U Rektoratu bi bila izvršena validacija formulara i prenos u centralnu bazu podataka. Dugoročno rešenje jeste implementacija centralnog informacionog sistema koji je tada postao projekat visokog prioriteta.

## II. Definisanje modela podataka

Prilikom definisanja modela podataka vodilo se računa o tome da se obuhvate svi podaci koji bi trebalo da se nađu u centralnom informacionom sistemu jednog univerziteta. Pored podataka koji su neophodni za Školu doktorskih studija i za akreditaciju, obuhvaćeni su i podaci važni za razne statističke analize Rektorata i resornih ministarstava, kao i podaci koji su potrebni za registre i javne isprave. U model je implementiran i distribuiran pristup dostavljanja podataka pomoću informacionog sistema samih institucija koji je planiran kao sledeća faza, a koristio bi Web servise. Takođe, model je dizajniran tako da može da prati istoriju podataka, da podržava prikupljanje podataka iz više izvora kao i prosleđivanje podataka institucijama članicama u zavisnosti od toga čija je odgovornost ažuriranje podataka.




Sl. 1. Proces nastajanja modela podataka za informacioni sistem Univerziteta.

## III. Definisanje formulara

Formular je trebalo da zadovolji više zahteva. Obrazac treba da bude na nekom od formata koji je poznat svima i koji je moguće popuniti i sa osnovnim poznavanjem rada

I. O. Autor, Univerzitet u Beogradu, Srbija (telefon: 381-64-1204252; faks: 381-11-3248681; e-mail: irena@rect.bg.ac.rs).

J. P. Autor, Elektrotehnički fakultet u Beogradu, Bulevar kralja Aleksandra 73, 11120 Beograd, Srbija; (e-mail: jelica.protic@etf.rs).

na računaru, pa i bez mogućnosti pristupa Internetu. Sa druge strane, tehnički zahtevi bili su složeniji, naime neki od podataka morali su da budu u nekom unapred definisanom formatu ili su morali da budu izabrani sa liste dozvoljenih vrednosti. Kao prirodno rešenje, na osnovu analize [4, 6], izabran je *Microsoft Excel*. Formular o nastavniku je primer formulara koji je velikog obima i koji obuhvata sve neophodne podatke na jednom mestu.

Formular o nastavniku podeljen je na nekoliko celina u zavisnosti od oblasti za koju su traženi podaci, pri čemu je za svaku celinu odvojen po jedan *worksheet*. Primer fragmenata popunjenog formulara dat je na slici 2. Pojedina polja su imala zadati format, kao što su datumi, JMBG-ovi, dok su neka polja bile liste sa padajućim menijem koje su bile definisane na posebnom *worksheet*-u. Za neka polja je ostala slobodna forma. Sva polja u koja korisnik ne upisuje podatke zaključana su lozinkom [6]. Definisana je i logika imenovanja fajlova.

Zaključani formular je dostavljen institucijama-članicama, a bio je dostupan i na Internetu. Institucije su popunile, prikupile i poslale sakupljene formulare, uglavnom na *CD*-ovima. Kao pomoć u popunjavanju organizovana je prezentacija.

## IV. Obrada formulara

Pošto su u Rektoratu prikupljeni formulari, pristupilo se problemu uvoza podataka u bazu. Na osnovu iskustava Računskog centra Univerziteta u Beogradu, odlučeno je da se za RDBMS koristi MySQL 5.0.45, a za parsiranje Java platforma sa korišćenjem Apache POI 3.1. biblioteke [4, 5]. Treba napomenuti da su u bazi već postojali osnovni podaci o nastavnicima i saradnicima i da je bilo potrebno ukrstiti ih sa pristiglim podacima iz formulara. U parseru je za pristup bazi korišćen standardni konekcioni string i drajver za MySQL.

### A. Otvaranje fajla

Parser je implementiran tako da se svi fajlovi smeštaju u jedan folder i čitaju se redom u petlji, tako da je celokupan unos validnih fajlova iniciran i završavan na jednom mestu. Bilo je formulara koji nisu čitljivi, kao i onih koji su bili kopije zadatih formulara sa izmenjenom formom. Problema sa imenima fajlova nije bilo.

```
POIFSFileSystem fs = new POIFSFileSystem(new
FileInputStream(putanja));
HSSFWorkbook wb = new HSSFWorkbook(fs);
```

### B. Čitanje ćelije

Čitanje formulara se vrši redom, po svakom *worksheet*-u. U jednoj petlji prolazi se kroz sve ćelije na *worksheet*-u koje su od interesa. Do svake ćelije vrši se precizna navigacija.

```
HSSFSheet sheet = wb.getSheetAt(sheetBr);
HSSFRow row = sheet.getRow(1);
HSSFCell cell = row.getCell((short) 1);
```

U zavisnosti od tipa podatka koji se čita iz ćelije, postoje različite metode za pristup.

```
String podatak   = cell.toString();
double realan  = cell.getNumericCellValue();
Date datum  = cell.getDateCellValue();
```

Prilikom čitanja ćelija javljali su se problemi, najvećim delom zbog nepridržavanja pravila za popunjavanje formulara, ali i delom zbog nedostataka *Excel*-a i *Windows* okruženja.

Najčešći problemi odnosili su se na korišćenje latinice umesto ćirilice, nekorišćenje naših slova, kao i korišćenje samo velikih, odnosno samo malih slova.

### C. Obrada podataka

Svi podaci iz formulara mogu se razvrstati u 5 grupa:

1. podaci su čistog tekstualnog karaktera, proizvoljne dužine i formata,
2. podaci koji moraju da zadovoljavaju određen format (datum, bodovi, JMBG),
3. podaci koji moraju biti izabrani iz zadate liste,
4. podaci u formularu koji nisu iz liste, ali se moraju upariti sa podacima koji su u bazi u šifarniku,
5. slike.

Podaci iz prve grupe ne zahtevaju nikakvu obradu i mogu se direktno unositi u bazu. Iako se ovde nisu očekivali problemi, bilo je slučajeva gde su korisnici koristili polja u druge svrhe ili čak ubacivali skenirane slike umesto teksta. Sam *Excel* je pravio probleme prilikom čitanja nekog podatka neodgovarajućeg formata, koji je upisan prevazilaženjem uvedene validacije, komandom *Copy / Paste*.

Podaci iz druge grupe zahtevaju proveru formata i eventualnu korekciju. Često su se javljali nepotpuni datumi, a bilo je i slučajeva gde je naveden broj bodova, pa u zagradi alternativa tom broju.

Podaci iz treće grupe zahtevaju implementiranje metode koja će pronaći odgovarajuće ključeve iz šifarnika za zadati podatak. Ukoliko je pronađen odgovarajući ključ, preostaje još samo ubacivanje u bazu. I pored definisanih lista, za koje je uložen trud da budu što potpunije, korisnici su ponekad smatrali da izbor nije dovoljan i pronalazili načine da ih ne primene, na primer kopiranjem *text box*-a preko zadatog polja.

Podaci iz četvrte grupe zahtevaju najobimniju obradu. Manje zahtevni su podaci za koje se traži uklapanje samo sa jednim poljem iz baze kao što su mesta, opštine, države, a tamo gde treba izvršiti uparivanje sa više polja iz baze, bio je neophodan razvoj dodatne logike.

Za pronalaženje jednog parametra korišćeni su *properties* fajlovi, koji su pretraživani uz pomoć ključeva. Zbog podataka ovog tipa bilo je potrebno pokretati parser više puta u test režimu kako bi se u *properties* fajlove upisale sve varijante koje su korisnici uneli za jednu istu vrednost. Recimo, za upis podataka o državljanstvu korišćen je sledeći spisak ključeva:

```
Srbija = Srbija
Srpsko = Srbija
srpsko = Srbija
Republike_Srbije = Srbija
```

Republika_Srbija = Srbija
=Nepoznato
Srpska =Srbija
R._Srbije = Srbija
R_Srbije=Srbija
Dr\u017eavljanin_Republike_Srbije = Srbija
SRB=Srbija
Republika_Srbija_(_biv\u0161a_SFRJ_) = Srbija
R._Srbija = Srbija
Jugoslavija = Nepoznato
R.Srbija = Srbija
SFRJ = Nepoznato
Jugoslavija,_Srbija = Srbija
Crpsko = Srbija
spsko = Srbija
Crnogorsko = Crna Gora
crnogorsko = Crna Gora
crnogorski = Crna Gora
Srbija,_SFRJ = Srbija
Srbijansko = Srbija
SFRJ_Jugoslavija = Nepoznato
SFRJugoslavija = Nepoznato
Jugoslavija_(Srbija)=Srbija
SR_Srbija=Srbija
Srbije=Srbija
Sr.Srbija = Srbija
srpko = Srbija

Primer pronalaženja podataka koji već postoji u bazi sa više atributa jeste osoba, u ovom slučaju nastavnik. Nastavnik je u bazi opisan sa JMBG-om, imenom, srednjim imenom i prezimenom. Broj kombinacija koje se mogu postići sa podacima koji mogu biti nepotpuni i netačni zahteva korišćenje dodatne logike. Najveći problem predstavlja uparivanje prezimena osoba ženskog pola koje mogu promeniti prezime ili na njega dodati drugo prezime. Dodatnu teškoću predstavlja i činjenica da i same osobe pri popunjavanju formulara ne unose svoje ime i prezime uvek na isti način. Na primer, ponekad koriste crticu između dva prezimena, ponekad ne, izostavljaju neko od prezimena, ubacuju srednje slovo, a dešava se i da zamene redosled dva prezimena. Za pretragu baze definisan je veći broj metoda u zavisnosti od broja parametara. Takođe su definisane metode koje pokušavaju da isprave i greške u podacima.

Za podatke koji spadaju u petu grupu u biblioteci POI 3.1. postoji podrška i njihovo čitanje se vrši u jednom koraku. Pri tome se definišu format, veličina i rezolucija koje bi slika trebalo da zadovolji, jer su stizale i slike neprikladne da se nađu u bazi.

```
List<?> lst = wb.getAllPictures();
for (Iterator<?> it = lst.iterator(); it.hasNext();)
{
    HSSFPictureData pict = (HSSFPictureData)
    it.next();
    String ext = pict.suggestFileExtension();
    // obrada slike
}
```

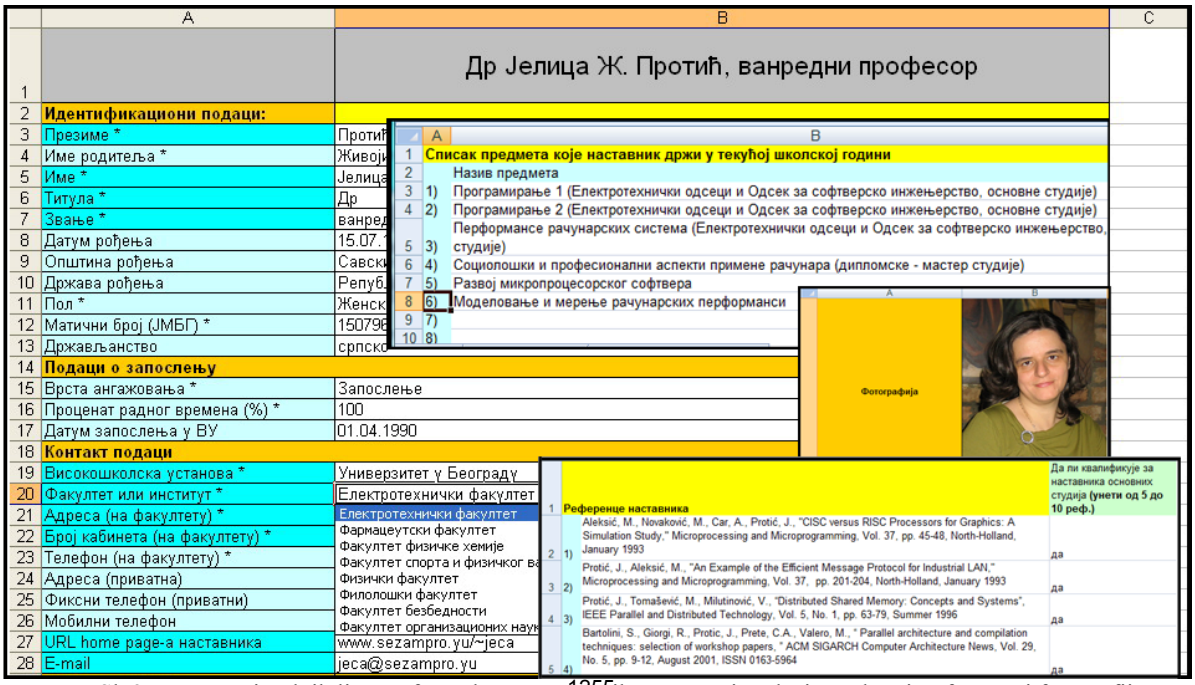
### D. *Uvoz podataka u bazu*

Prilikom unosa podataka u bazu bilo je potrebno proveriti da li zadati podatak već tamo postoji. Ukoliko postoji, potrebno je proveriti da li je zastareo pa ga ažurirati, ili je već validan, pa ga ne treba unositi ponovo. Svi podaci koji su uneti na ovaj način obeleženi su da potiču iz formulara.

## V. PREZENTACIJA PODATAKA

Jedan od ciljeva prikupljanja podataka jeste njihova unifikacija, kako bi prezentacija podataka na Internetu bila jednaka za sve institucije - članice Univerziteta. Internet prezentacija urađena je tako da automatski prikazuje podatke koji se nalaze u bazi nezavisno od izvora podataka. Primer prikaza podataka o nastavniku i predmetu dat je na slici 3.

Na osnovu unetih podataka moguće je uraditi pretraživanje i statistike po različitim parametrima.




Sl. 2. Fragmenti radnih listova formulara o nastavniku: osnovni podaci, predmeti, reference i fotografija

Sl. 3. Primer prezentacije podataka na Internetu

## VI. Zaključak

U uslovima gde se traži brzo prikupljanje veće količine podataka od korisnika koji nemaju obezbeđen pristup Internetu i čije institucije ne poseduju adekvatan informacioni sistem, korišćenje formulara je opravdano. Ukoliko se odluči za ovaj pristup, treba razmisliti o ažurnosti podataka, koji se stalno menjaju, a u bazi ostaju u stanju u kakvom su bili u trenutku uvoza, pa se samim tim otvara pitanje ponovnog prikupljanja podataka. Ipak, gotovo svi podaci koji se nalaze u formularu ostaju važeći i na dalje i potrebno je samo dodavati nove podatke (radove, projekte itd). I sam model podataka, koji se koristi kao osnova u troslojnoj arhitekturi, podložan je čestom menjanju pa je samim tim neophodno i često ažuriranje dela parsera koji radi sa bazom podataka.

Drugi zaključak odnosi se na sam pristup definisanju formulara: veliku pažnju treba posvetiti pravilima koja korisnik mora da poštuje prilikom popunjavanja. Definisanje granice između slobode i udobnosti popunjavanja je od suštinskog značaja. Naime, što je formular strožije definisan, a polja samim tim uniformnija i lakša za obradu, to je korisniku teže da popuni formular, a takve probleme korisnik uglavnom rešava izuzecima koje je jako teško programski podržati.

Iako su korisnici koji su popunjavali formulare bili na visokom nivou znanja rada sa *Excel*-om, bilo je potrebno proslediti veliki broj primera validno popunjenih formulara radi lakšeg snalaženja.

Parser treba pisati tako da pozivi po mogućstvu budu idempotentni, a da metode budu što univerzalnije kako bi obuhvatale što više izuzetaka.

Zbog svega navedenog, centralni informacioni sistem Univerziteta u Beogradu stavljen je na mesto broj jedan kada su prioriteti u pitanju. Ideja je da podatke dostavljaju institucije članice uz pomoć automatizovanih Web servisa. Oformljena je radna grupa koju čine predstavnici svih fakulteta. Radna grupa je usvojila model podataka na nivou Univerziteta i svaki tekući problem pokušava da reši na nivou komunikacije informacionih sistema.

## Literatura

[1] I. Jovanov, I. Tucaković, J. Protić, "Model doktorskih studija zasnovanih na akreditacionim pravilima", Trend 2009, zbornik radova, pp. 43-46, Kopaonik
[2] I. Odžić, O. Blagojević, J. Protić, "Informacioni sistem Univerziteta," Trend 2009, zbornik radova, pp. 148-151, Kopaonik
[3] B. Babić, O. Blagojević, I. Odžić, "Struktura i funkcionalnosti informacionog sistema Dositej za podršku i kontrolu viso-koškolskog nastavnog procesa," Trend 2006, zbornik radova, pp. 218-221, Kopaonik
[4] http://www.lacher.com/toc.htm
[5] http://poi.apache.org/
[6] http://www.ozgrid.com/Excel/ExcelSpreadsheetDesign.htm

## Abstract

University of Belgrade, as a complex organization with more than 40 institutional members and over 5000 employees, initiated the process of data integration in order to persuade University accreditation, establish The School of doctoral studies, prepare data for the registry of institutions of higher education, and publish relevant information on the University Web site. The process started by defining multiple worksheets with data validations in Microsoft Excel, which were filled-in and collected by the institutional members and delivered to the Rector's Office. After the system analysis, data model was developed and the University of Belgrade database was implemented. Data fields from the worksheets were additionally verified by the parser and imported to the database. Parser was developed using Java and Apache POI 3.1. library.

## BUILDING THE UNIVERSITY OF BELGRADE DATA BASE USING WORKSHEETS FROM MULTIPLE SOURCES: LESSONS LEARNED

Irena Odžić, University of Belgrade, Jelica Protić,
School of Electrical Engineering, University of Belgrade