

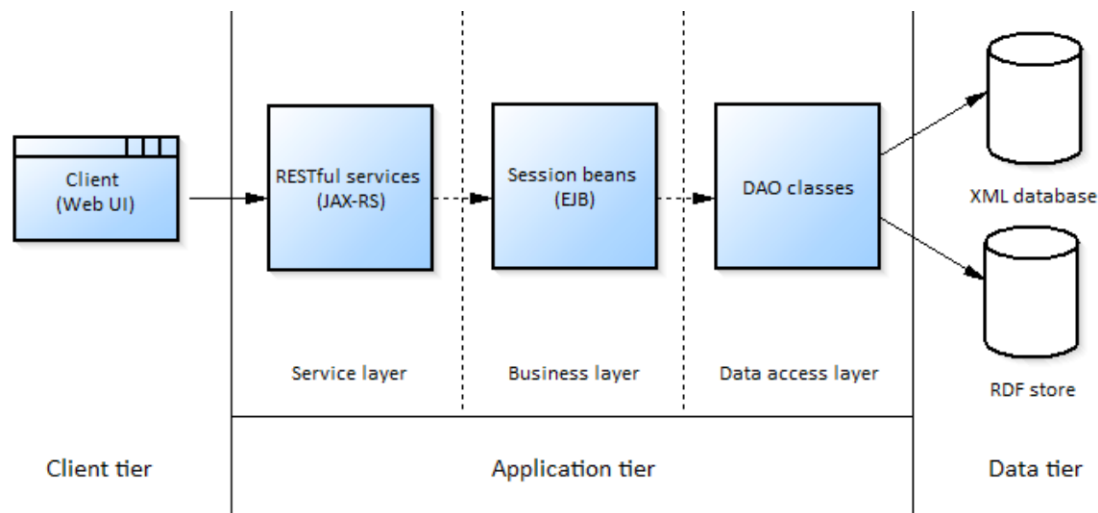
Projektni zadatak

XML i veb servisi, 2017/18. godina

Projektovati, implementirati i testirati informacijski sistem za podršku objavljivanju naučnih publikacija.

Informacijski sistem treba da podrži interne poslovne procese (kao što su predaja naučnih radova i revizija na postojeće verzije radova u sistemu, predaja recenzija, povlačenje radova i rukovođenje poslovnim procesom objavljivanja naučnih rezultata¹) i eksterne poslovne procese (kao što su pronalaženje i pregledanje naučnih radova).

Dokumente (naučne radove, revizije, recenzije i prateća pisma) u okviru informacijskog sistema je potrebno modelovati kao XML gramatike uz pomoć XML Schema jezika; dok metapodatke dokumenata treba predstaviti kao skup RDFa² atributa u okviru datih gramatika. Za modelovanje identifikatora XML elemenata u okviru dokumenata koristiti mehanizam URL-a (npr. /paper17/section2/para5). Sve dokumente i njima sadržane metapodatke je neophodno predstaviti po W3C standardima. Sistem projektovati po troslojnoj softverskoj arhitekturi, a komunikaciju između serverske i klijentske strane realizovati pomoću RESTful veb servisa. Izbor programskog jezika i platforme za implementaciju serverske i klijentske strane prepušten je studentima (primeri na vežbama su u Javi). Grafički prikaz arhitekture sistema dat je sledećim dijagramom (slika 1).



Slika 1: Višeslojna arhitektura informacijskog sistema za objavljivanje naučnih publikacija

¹ Primer evaluacije naučnih rezultata: [What is peer review?](#)

² RDFa 1.1 Primer, <https://www.w3.org/TR/rdfa-primer/>

Backend aplikacija informacionog sistema za podršku objavljivanju naučnih publikacija predstavlja višeslojnu aplikaciju koja se sastoji iz slojeva prikazanih dijagramom. Aplikacioni sloj sistema za objavljivanje naučnih radova treba realizovati kroz tri podsloja (*data access*, *business* i *service layer*), a detalji koji se odnose na samu implementaciju konkretnih podslojeva su dati u nastavku.

Data access layer

- Predstavlja generički sloj aplikacije koji enkapsulira osnovne CRUD (*create*, *retrieve*, *update*, *delete*) operacije za pristup podacima iz XML, odnosno RDF baze podataka.
- Za potrebe predmetnog projekta DAO klase implementirati bilo pomoću namenskih komponenti (npr. SLSB) ili kao obične klase.

Business layer

- Predstavlja srednji sloj aplikacije koji implementira fasadu nad slojem za pristup podacima. *Business layer* posreduje između *data access* i *service layer*-a, stavljajući servisnom sloju na raspolaganje neophodan skup funkcionalnosti kroz metode poslovne logike.
- Za potrebe predmetnog projekta *business layer* treba implementirati kao namenske komponente.

Service layer

- Predstavlja krajnji sloj aplikacije koji definiše javno dostupan API implementiranih funkcionalnosti krajnjem korisniku.
- Za potrebe predmetnog projekta servisni sloj treba implementirati kao RESTful veb servise koje referenciraju komponente iz *business layer*-a upotrebom *dependency injection* mehanizma.

Uloge korisnika su: **autor**, **recezent** (recezent je autor), **urednik** (urednik je recezent) i **neautentifikovani korisnik** tj. javnost.

Autor može da:

1. predlaže naučne radove i revizije na predložene radove

- implementirati formu u kojoj korisnik (tj. autor) ima mogućnost unosa anotiranog teksta naučnog rada kao i odgovarajućih metapodataka rada u okviru *rich edit* komponente³
- prilikom predaje rada ili njegove revizije, pored samog teksta rada autoru treba omogućiti i prilaganje propratnog pisma (*cover letter*) na sličan način kao i u slučaju rada
- minimalna funkcionalnost koju je neophodno implementirati jeste unos rada i propratnog pisma kao XML fragmente u okviru odgovarajućih *input* komponenti ili kroz *file input* komponente.
- omogućiti i unos metapodataka koji se ne generišu automatski iz dokumenta

³ Primer jedne takve komponente: [A schema-driven XML editor for the web - Xonomy](#)

2. povlači svoje naučne radove u proceduri

- omogućiti autoru da u okviru forme za pregledanje svojih naučnih radova koji su u proceduri odustane od procesa objavljivanja
- povlačenje naučnih radova u proceduri se može implementirati kao logičko brisanje, tj. brisanje izmenom statusa dokumenta

3. pronalazi objavljene naučne radove po metapodacima i po tekstu

- pronalaženje naučnih radova po tekstualnom sadržaju (tj. osnovnu pretragu) i po metapodacima (tj. naprednu pretragu) implementirati kroz zasebnu formu za pretragu naučnih radova na kojoj je korisnicima sistema omogućen unos fraza ili ključnih reči po kojima se vrši pretraga nad svim elementima naučnih radova koji sadrži date pojmove
 - pronalaženje radova po više metapodataka odjednom realizovati upotrebom logičkih operatora
 - prilikom modelovanja naučnih radova posebno obratiti pažnju na polja koja predstavljaju metapodatke (npr. kategorija rada, status rada, verzija, ID časopisa, oznaka sveske, oznaka izdanja, datum prijema, datum recenzije, datum prihvatanja rada, DOI, itd.), tj. podatke koji opisuju dokumente na osnovu kojih će biti implementirana napredna pretraga
- pretraga je omogućena samo za radove kojima korisnik može da pristupi (svi radovi sa statusom "objavljen", kao i radovi sa statusom "u proceduri" čiji je korisnik jedan od autora - videti sledeću stavku).

4. pronalazi svoje prihvaćene, odbijene i naučne radove u proceduri po metapodacima i tekstu

- pretraživanje sopstvenih naučnih radova po tekstualnom sadržaju (tj. osnovnu pretragu) kao i pretragu po metapodacima (tj. naprednu pretragu) implementirati u okviru iste forme kao i pretragu svih naučnih radova iz prethodne stavke
- autorima je omogućena pretraga svih svojih radova bez obzira na njihov status

5. pregleda naučne radove

- prilikom pregledanja nekog naučnog rada omogućiti prikaz metapodataka, kao i samog sadržaja rada u vidu XHTML-a pomoću XSLT transformacija
- pored prikaza naučnog rada u XHTML-u, omogućiti i preuzimanje rada u XML i PDF formatu

6. pregleda naučne radove po referencama (*cross-referencing*)

- prilikom pregledanja nekog naučnog rada omogućiti "skok" na referencirane naučne radove, tj. radove na koje se dati rad poziva ili ih citira; prethodnu funkcionalnost implementirati po uzoru na mehanizam linkovanja resursa (*hyperlink*)
 - prilikom modelovanja naučnog rada posebno obratiti pažnju na modelovanje referenci na druge dokumente, koje mogu da se nađu na bilo kom nivou hijerarhije unutar samog teksta rada
- pored navedenog, omogućiti pregled svih radova koji se pozivaju tj. citiraju dati rad

Recezent može da:

1. pregleda, prihvata i odbija radove dodeljene od strane urednika

- implementirati formu u okviru koje recezent pored uvida, ima mogućnost prihvatanja i odbijanja za svaki od njemu dodeljenih radova

2. evaluira naučni rad u vidu ocene i recenzije (*evaluation form*)

- omogućiti recezentu da u posebnoj formi u okviru *rich edit* komponente unese svoju recenziju na datu verziju (reviziju) naučnog rada
- omogućiti ocenu podobnosti rada u vidu upitnika, kao i konačnu ocenu rada u okviru iste forme za evaluaciju

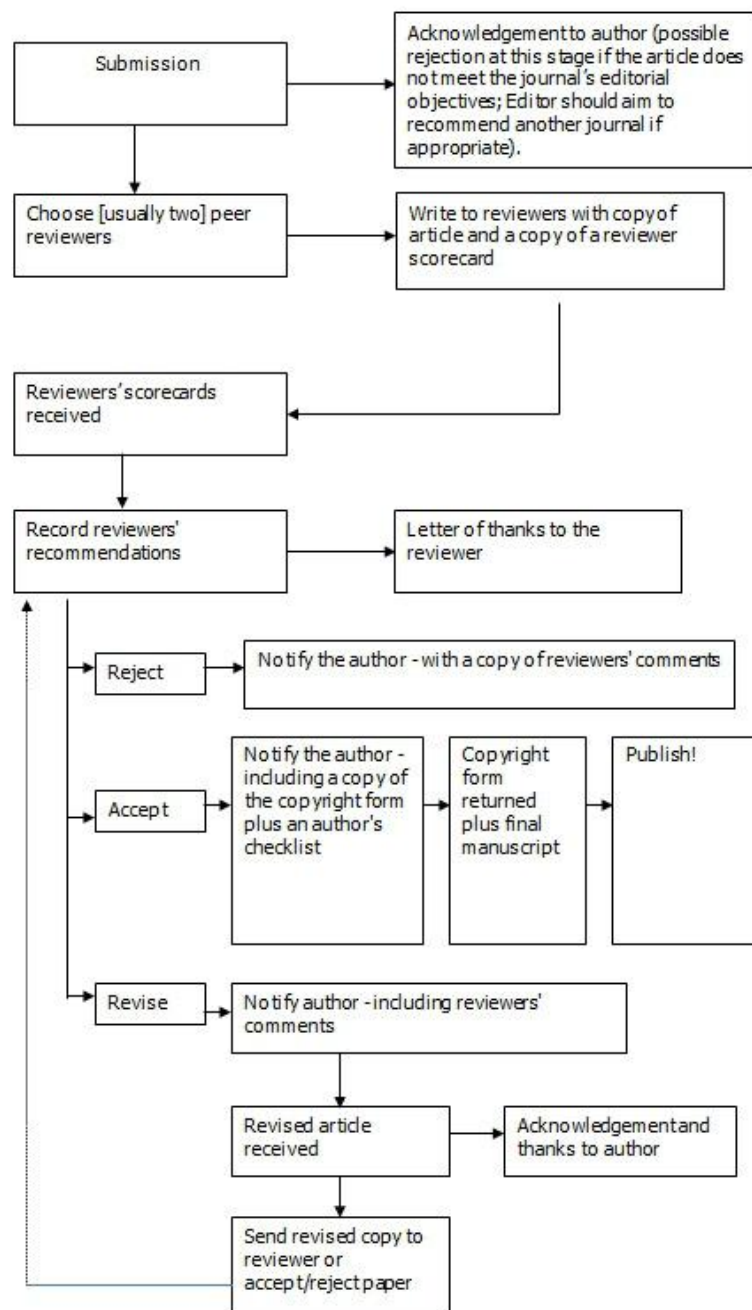
Urednik može da:

1. dodeljuje radove recezentima za evaluaciju

- implementirati posebnu formu u okviru koje se uredniku nudi spisak podobnih recezenata za svaki od predloženih radova sa mogućnošću dodele rada odabranom recezentu
- implementirati algoritam preporuke recezenata na osnovu njihove ekspertize i ključnih reči naučnog rada

2. rukovodi procesom objavljivanja naučnih publikacija

- omogućiti uredniku rukovođenje procesom objavljivanja naučnih publikacija (primer procesa je dat na slici 2), kroz mogućnost promene faza u kojima se proces nalazi
 - modelovati instancu poslovnog procesa kao XML dokument i skladištiti ga u XML bazi podataka kao perzistentan entitet
- korespondenciju između autora i recezenata (prosleđivanje recenzija autorima i revizija tj. ispravki radova recezentima) implementirati kroz sistem mail notifikacija
- u okviru forme za prikaz naučnog rada u proceduri omogućiti uredniku odbijanje i prihvatanje radova na osnovu evaluacije recezenata



Slika 2: Primer poslovnog procesa objavljivanja naučnog rada

Neautentifikovan korisnik ima mogućnost pronalaženja (po tekstu i metapodacima) i pregledanja objavljenih radova.

Pored navedenih funkcionalnosti, neophodno je omogućiti anonimizaciju radova prilikom slanja recezentima i recenzija prilikom slanja autorima rada⁴. Recezentima bi takođe trebalo omogućiti anotaciju teksta rada svojim komentarima, primedbama i sugestijama. Pored navedenog, treba

⁴ Objašnjenje anonimizacije: [Double-Blind Peer Review Guidelines](#).

implementirati i automatsko spajanje svih recenzija u jedan dokumenat (*merging*) pre slanja rada autorima na reviziju. Navedenu korespondenciju je potrebno ispratiti mail notifikacijama sa odgovarajućim sadržajem (notifikacije modelovati kao XML entitete). Omogućiti izvoz naučnih publikacija (radova), recenzija i pratećih pisama u XHTML i PDF formate, kao i izvoz metapodataka u RDF i JSON formate.