

**Z39.50 КЛИЈЕНТСКА АПЛИКАЦИЈА ЗА
ПРЕУЗИМАЊЕ БИБЛИОГРАФСКИХ ЗАПИСА
Z39.50 CLIENT APPLICATION FOR RETRIEVAL
OF BIBLIOGRAPHIC RECORDS**

Данијела Боберић Крстићев

РЕЗИМЕ: У оквиру развоја софтверског система БИСИС извршено је моделирање и имплементација клијентске стране Z39.50 протокола. Z39.50 протокол је широко распрострањен у оквиру библиотечке заједнице и намењен је за претраживања и добављања информација од удаљених система. У овом раду су приказане функционалности корисничког интерфејса имплементираних клијентске стране. Описане су различите могућности формирања упита. Клијентска страна подржава упит типа-1 који је дефинисан Z39.50 стандардом као и скуп атрибута биб-1. Могуће је креирати упите помоћу префикса који могу бити повезани и са логичким операторима. Поред тога, могуће је дефинисати и додатна ограничења на поједине префиксе, коришћењем осталих типова атрибута из скупа биб-1 (relation, position, structure...). Апликација је реализована као независна софтверска компонента и архитектура компоненте је таква да омогућава интеграцију апликације у различите библиотечке информационе системе.

КЉУЧНЕ РЕЧИ: Z39.50 протокол, кориснички интерфејс, претраживање, упитни језик

ABSTRACT: Modelling and implementation of the client side of Z39.50 protocol are obtained during the development of the software system BISIS. Z39.50 protocol is widely used in the library community and it is committed to searching and retrieving information from remote systems. Major functionalities of the user interface of the implemented client side are described in the paper. The main part of the paper is related to creating various types of queries. Client side supports the query formulation of the type-1 defined by the Z39.50 standard as well as attribute set known as bib-1. It is possible to create queries including one or more prefixes which may be joined with logical operators. In addition, there is possibility to add some constraints on particular prefixes, for instance order relations (less, greater, equal). The application is implemented as independent software components and component architecture is such that it allows the integration of applications in a variety of library information systems.

KEY WORDS: Z39.50 protocol, user interface, search, query language

1 УВОД

Развој библиотечких информационих система у последње време првенствено је окренут крајњем кориснику коме је потребно омогућити да на јединствен и једноставан начин приступи фондовима различитих библиотека. У циљу постизања ове интероперабилности осмишљени су различити библиотечки стандарди као што је на пример Z39.50 [1]. Овај стандард је широко распрострањен унутар библиотечких система и променио је начин функционисања ових система. На сајту Конгресне библиотеке [2] може се наћи списак преко 1000 библиотека чији фондови се могу претраживати помоћу Z39.50 протокола.

Z39.50 (*Information Retrieval: Application Service Definition and Protocol Specification*) је стандард организације NISO (*National Information Standards Organization*). Стандард описује сервисе који се користе приликом претраживања и добављања информација и даје спецификацију одговарајућег протокола који се користи при клијент/сервер комуникацији и који подржава дате сервисе. Неки од највише распрострањених серверских апликација које подржавају Z39.50 протокол су: **Zebra** [3], **Voyager LMS-Z39.50 Server** [4] и други.

Детаљно су анализирани функционални захтеви већег броја постојећих софтверских решења Z-клијената, као што су MarcEdit [5], Zsearcher Information [6], Surpass Coruscat [7], BookWhere [8]. Ове клијентске апликације

су доступне или као *open source* софтвери или као демо верзије одговарајућег комерцијалног софтвера. Већина анализираних софтвера омогућава додавање нове библиотеке за претраживање, затим пружају подршку за UTF-8 кодни распоред као и коришћење логичких оператора. Код већине софтвера пронађени записи се приказују у тзв. full formatu и приликом формирања упита користе се само вредности Use атрибута из скупа атрибута биб-1.

Нека од ових софтверских решења детаљно су описана у радовима [9-13] а у раду [14] извршена је компаративна анализа тих софтверских решења на основу усвојених критеријума. Поред тога у раду [15] су наведени критеријуми које треба да задовољи Z-клијент апликација, а у раду [16] су наведени критеријуми које треба да задовољи софтверски систем за библиотечко пословање. Један од критеријума подразумева и постојање Z-клијента.

Постоје радови у којима је дат преглед стања библиотечких информационих система на националном нивоу који пружају могућност претраживања по Z39.50, као што су [17-19]. Један од система који се користи у нашој земљи је библиотечки софтверски систем БИСИС [20]. Развој актуелне четврте верзије базиран је на XML технологијама. Део који се односи на развој едитора за каталогизацију описан је у радовима [21-24]. У радовима [25-27] описан је део система који се односи на коришћење библиотечке грађе, односно циркулацију.

Анализом постојећих софтверских решења Z-клијената као и прегледом релевантне литературе, уочене су одређене функционалности које би Z39.50 клијентска апликација требала да испуни и у овом раду описан је кориснички интерфејс Z39.50 клијентске апликације за претраживање и преузимање записа по Z39.50 стандарду који задовољава те функционалности. Z39.50 клијентска апликација која је описана у овом раду део је библиотечког информационог система БИСИС.

2. МОДЕЛИРАЊЕ И ИМПЛЕМЕНТАЦИЈА Z39.50 КЛИЈЕНТСКЕ АПЛИКАЦИЈЕ

Основне карактеристике апликације чији је кориснички интерфејс описан у овом раду су:

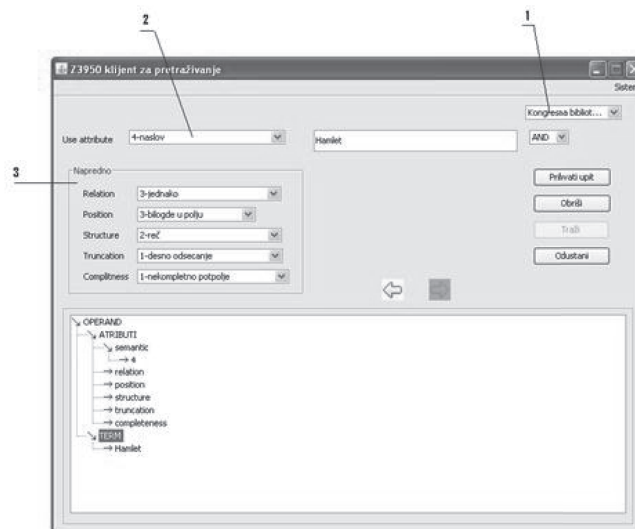
- Формирање упита типа -1
- При креирању упита користе се сви типове атрибуте из скупа биб-1
- Формирани упит се може ажурирати и графички се приказује
- За сваку библиотеку се дефинишу одговарајући параметри као и ограничења на скуп атрибута.
- Преузимање библиографског записа у XML формату
- Подршка за UTF-8 кодни распоред

Детаљна спецификација информационог захтева и имплементација Z39.50 клијентске апликације за претраживање и преузимање библиографских записа по стандарду Z39.50 детаљно је описана у радовима [28,29]. Апликација је имплементирана у програмском језику Јава, а за реализацију клијентске стране протокола Z39.50 искоришћен је алат JAFER (Java Access For Electronic Resources) [30]. Овај алат имплементира клијентску страну протокола Z39.50 у програмском језику Јава и заснован је на XML технологијама. Алат се развија на универзитету у Оксфорду и представља *open source* библиотеку. Верификација овог пакета је урађена на Оксфордском универзитету у оквиру имплементације *web* портала који пружа преузимање библиографских записа по протоколу Z39.50.

3. ФУНКЦИОНАЛНОСТИ КОРИСНИЧКОГ ИНТЕРФЕЈСА

У овом поглављу описане су функционалности апликације. Употреба апликације приказана је на примерима различитих упита.

Приликом покретања апликације кориснику се отвара почетна екранска форма која је приказана на слици 1. На овој екранској форми корисник може изабрати библиотеку коју жели да претражи избором из падајуће листе означене бројем 1. У свим примерима који следи биће изабрана Конгресна библиотека у Вашингтону, и на њој је уједно извршена и верификација система.



Слика 1. – Почетна екранска форма

3.1 Претраживање по вредностима Use атрибута

Z39.50 стандард дефинише употребу одређених упитних језика и најчешће се користи упитни језик типа-1. Овај упитни језик везује се са појмом скупова атрибута и најчешће коришћени скуп атрибута који се користи у претраживању библиографских записа је скуп атрибута *bib-1* [31]. *Bib-1* скуп атрибута састоји се од шест типова атрибута:

- *Use Attributes*
- *Relation Attributes*
- *Truncation Attributes*
- *Structure Attributes*
- *Completeness Attributes*
- *Position Attributes*

Use атрибут - Дефинише улазне критеријуме претраге (наслов, аутор, одредница, итд).

Relation атрибут - Дефинише релацију (на пример: мање, веће, једнако, итд) која треба да важи између изабраног Use атрибута и израза који се тражи.

Truncation атрибут - Дефинише који део вредности критеријума претраге ће бити коришћен током претраге (почетак речи, крај речи, итд). Улога овог атрибута може се упоредити са цокер знацима који се користе у стандардним претраживањима.

Structure атрибут - Дефинише формат термина који се тражи, односно да ли терм представља реч, фразу, датум или нешто друго.

Position атрибут - Дефинише где се у оквиру поља налази тражени термин. Атрибутом *Position* дефинише се локација термина који се тражи у оквиру поља или потпоља у којима се терм јавља.

Completeness атрибут - Дефинише да ли терм за претрагу мора или не мора бити једина вредност у датом пољу/потпољу.

У овом одељку на следећем примеру описана је употреба апликације приликом претраживања по вредностима *Use* атрибута.

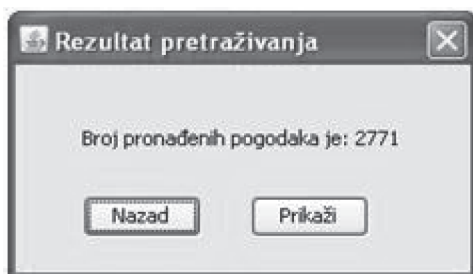
Пример 1: Пронаћи све библиографске записе који у наслову садрже реч *Hamlet*.

Да би се задовољио постављени задатак корисник прво мора да изабере одговарајући *Use* атрибут из падајуће листе означене бројем 2 на слици 1 чија вредност је *4-naslov*, и служи за претраживање по наслову. Ефикасно претраживање обавезује корисника да познаје семантику вредности *Use* атрибута.

Бројем 3, на слици 1 означене су падајуће листе из којих се бирају напредни атрибуту из скупа *bib-1* и то ће бити детаљније објашњено у одељку 3.3. Атрибуту који су груписани у напредне атрибуте су опциони и уколико их корисник не селекује апликација ће узети подрезумеване вредности тих атрибута које су дефинисане у оквиру саме апликације. У овом конкретном примеру корисник не мора да селекује напредне атрибуте јер постављени упит то не захтева. Терм *Hamlet* који се тражи уноси се у једнолинијски едитор.

Притиском на дугме *Prihvati upit* врши се формирање упита који је приказан у доњем делу екранске форме на слици 1. Формирани упит је графички репрезентован као стабло где у овом примеру имамо коренски елемент *OPERAND* који у себи садржи поделементе којима су описане вредности изабраних атрибута (подеlement *ATRIBUTI*) и вредност термина (подеlement *TERM*). Уколико корисник није задовољан креираним упитом притиском на дугме *Odustani* обрисаће се целокупни садржај упита који је до тада креиран.

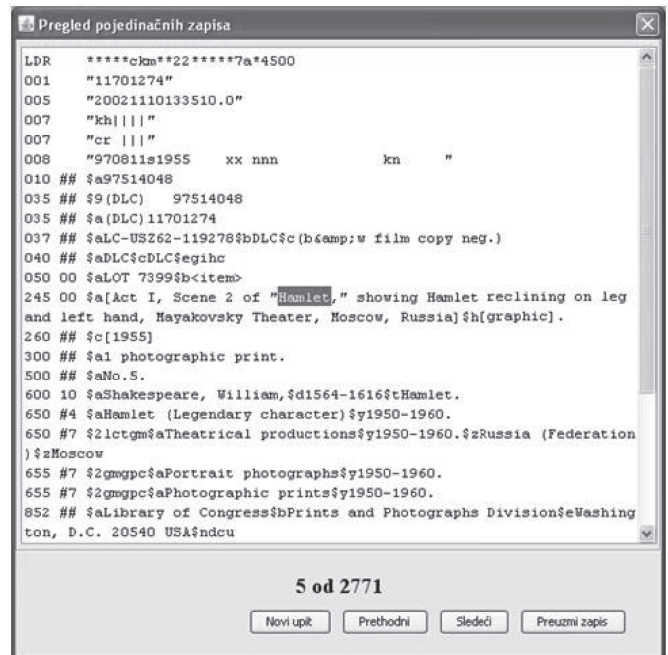
Након притиска на дугме *Traži*, могу се јавити одређене грешке типа одбијена конекција на сервер, сервер не подржава неке атрибуте и томе слично и у том случају ће систем генерисати изузетак а кориснику приказати одговарајући дијалог са поруком о грешци. Уколико није дошло до грешке појављује се прозор (слика 2) са информацијом о броју погодака.



Слика 2. – Прозор са бројем погодака

Притиском на дугме *Nazad* корисник одустаје од прегледа записа и враћа се на почетну екранску форму (слика 1). Притиском на дугме *Prikaži* отвара се нова екранска форма за преглед резултата (слика 3).

Централни део екранске форме представља део за приказ пронађеног записа у пуном формату. Такође постоји информација о редном броју записа који се тренутно гледа. Кориснику је омогућена навигација кроз скуп добијених резултата притиском на дугме *Prethodni* и *Sledeći*, а притиском на дугме *Preuzmi zapis*, библиографски запис се снима у фајл систем у виду XML документа. Притиском на дугме *Novi upit* корисник се враћа на почетну екранску форму (слика 1) за постављање новог упита.



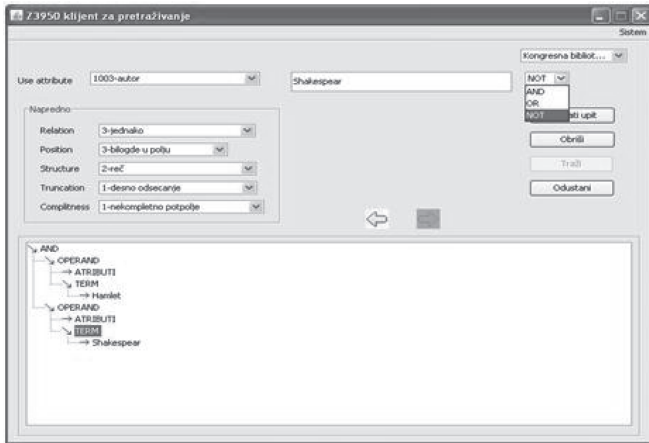
Слика 3. – Прозор за преглед појединачних записа

3.3 Претраживање употребом логичких оператора

Имплементирани едитор омогућава употребу логичких оператора *AND*, *NOT* и *OR* и њихов избор врши се избором из падајуће листе.

Пример 2: Наћи све записе који у наслову садрже реч *Hamlet*, аутор је *Shakespear* (Шекспир) и нису на енглеском језику.

Да би се формирао упит којима се решава постављени задатак, корисник прво мора креирати операнд где је *Use* атрибут *4-naslov*, а терм који се тражи је *Hamlet*. Ово је досада исто као и у примеру 1, једино што још треба да се уради је да се из падајуће листе изабере оператор *AND* којим ће се повезати следећи операнд. Када је формиран први операнд (притиском на дугме *Prihvati upit*), корисник треба да креира нови операнд. Потребно је да изабере *Use* атрибут *1003-autor*, који означава аутора књиге и унесе терм *Shakespear*. Пошто корисник не жели књиге на енглеском језику приликом формирања другог операнда било је неопходно да се изабере логички оператор *NOT*. Формирани операнд приказан је на слици 4.

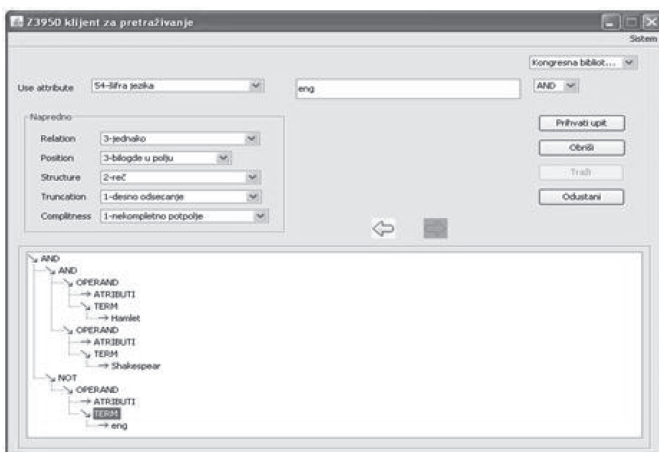


Слика 4. – Други операнд упита

На сличан начин формира се и трећи операнд, потребно је селектовати *Use* атрибут *044-šifra jezika*, који представља шифру језика на ком је написана књига, по шифарнику језика који се примењује за обраду записа по MARC 21 формату шифра за енглески језик је *eng*. На слици 5 приказан је формиран операнд.

Кориснику је омогућено кретање кроз формиране операнде и на тај начин могуће је изменити већ креирани операнд и притиском на дугме *Prihvati upit* измењени операнд се ажурира. Такође, притиском на дугме *Obriši* брише се операнд који је тренутно селектован.

Графички приказ упита дат је у доњем делу екранске форме на слици 5. Графички приказ одговара префиксној нотацији упита. Односно прво следи оператор па затим операнди. Тако у овом примеру коренски елемент у стаблу означава логички оператор који повезује операнде а поделемента OPERAND приказани су конкретни операнди са вредностима изабраних атрибута (поделемента ATRIBUTI) и изразом који се тражи (поделемента TERM).



Слика 5. Трећи операнд упита

Притиском на дугме *Traži* врши се претраживање изабране библиотеке и даљи принцип рада едитора је описан у првом примеру.

4.4 Напредно претраживање

У овом одељку су приказани примери упита који укључују напредне атрибуте из скупа *bib-1*, и који су груписани у оквиру секције *Napredno* (слика 1, под бројем 3). Сваки атрибут из овог дела допуњује вредност *Use* атрибута, што значи да се не могу селектовати само вредности ових атрибута а да при томе није селектована и вредност за *Use* атрибут.

Пример 3: Пронаћи библиографске записе где је аутор Shakespear (Шекспир), а књиге су објављене у периоду од 2000. до 2007. године, укључујући и године које представљају границе интервала.

Опис упит који корисник треба да постави дат је у табели 1

Из табеле се види да упит има три операнда и сви су повезани логички оператором AND. Први операнд има само *Use* атрибут чија вредност означава аутора. Пошто треба да се дефинише период у ком су књиге издате, искоришћена су два операнда који имају *Use* атрибут *031-datum izdavanja* који представља датум издавања књиге. Међутим, уколико се не би за сваки операнд дефинисао и атрибут *Relation*, добио би се следећи упит: „Наћи све књиге које је написао Шекспир и које су издате 2000. и 2007. године“ (што је различито од постављеног упит).

	тип атрибута	вредност атрибута	терм
Први операнд	Use	1003-autor	Shakespear
оператор	AND		
Други операнд	Use	031-datum izdavanja	2000
	Relation	4-veće ili jednako	
оператор	AND		
Трећи операнд	Use	031- datum izdavanja	2007
	Relation	2-manje ili jedanko	

Табела 1. – Табеларни приказ упит за пример 3

Из тих разлога за други операнд је дефинисан *Relation* атрибут са вредношћу *4-veće ili jednako*. *Relation* атрибут има задатак да допуни значење *Use* атрибута и у овом конкретном примеру то значи да година издања мора бити већа или једнака од 2000. Слично је за трећи операнд дефинисан *Relation* атрибут са вредношћу *2- manje ili jedanko* и на тај начин је реализован критеријум да је година издања мања или једнака од 2007.

Пример 4: Пронаћи књиге које у наслову садрже фразу *Thinking in Java* (табела 2).

Овај упит се састоји само од једног операнда који има два атрибута: *Use* атрибут са вредношћу *4-naslov* и *Structure* атрибут са вредношћу *1-fraza*. Атрибутом *Structure* је дефинисан услов да треба наћи тачну фразу, да је уместо ове вредности селектована вредност *2-reč*, тада би били пронађени и записи који у наслову садрже све наведене речи, али које не морају бити у наведеном редоследу. Такође овим упитом су пронађени и записи који садрже тачну фразу у наслову али имају поред тога и неке друге изразе. Такав један случај је запис који у наслов садржи израз *Thinking in Java programming language*.

	тип атрибута	вредност атрибута	терм
Први операнд	Use	4-naslov	<i>Thinking in Java</i>
	Structure	1-fraza	

Табела 2. – Табеларни приказ упит за пример 4

Уколико корисник жели да пронађе само записе који садрже тачан израз потребно је да се постави вредност *Truncion* атрибута на *100-bez odsecanja*.

Пример 5: Пронаћи књиге које у предметној одреднице садрже изразе које почињу са речју *Method* (табела 3).

Тражени упит има само један операнд при чему *Use* атрибут има вредност *21-predmetna odrednica* која означава предметну одредницу, и додат је и атрибут *Truncation* са вредношћу *1- desno odsecanje*. Атрибут *Truncation* служи да омогући претраживање по почетку речи, крају речи и слично. У овом примеру је искоришћена вредност *1- desno odsecanje* што значи да предметна одредница књиге мора да почне низом карактера *Method*. Формирани упит приказан је у табели 3.

	тип атрибута	вредност атрибута	терм
Први операнд	Use	21-predmetna odrednica	<i>Method</i>
	<i>Truncation</i>	1- desno odsecanje	

Табела 3. – Табеларни приказ упит за пример 5

Овако постављеним упитом биће пронађени и записе који у предметној одредници садрже и речи као што су: *Methods*, *Methodist*, *Methodology* и томе слично. Да је било потребно да се пронађе само запис који садржи тачну реч *Method* тада би *Truncation* атрибут имао вредност *100-bez odsecanja*.

Са одговарајућим променама вредности атрибута *Truncation* могу се добити записи код којих на пример

наслов почиње са задатом речју, наслов се завршава са задатим низом карактера или се задати низ карактера налази у средини израза који представља наслов.

5. ЗАКЉУЧАК

Приказан је опис функционалности Z39.50 клијентске апликације за претраживање и преузимање библиографских записа путем протокола Z39.50. Апликација је реализован као независна софтверска компонента, али је архитектура компоненте таква да омогућава интеграцију апликације у различите библиотечке информационе системе.

Апликација подржава упит типа-1, који је описан стандардом Z39.50, али одређеном модификацијом едитор би се могао прилагодити за формирање осталих врста упита које су такође дефинисане да се могу користити при комуникацији путем протокола Z39.50. Под модификацијом би се подразумевало прилагођавање екранске форме захтева које налаже нова врста упита, док би сама архитектура система остала непромењена. То значи да би се у том случају такође креирао XML документ који би био инстанца неке друге шеме која описује друге врсте упита подржане протоколом Z39.50.

Такође још једна предност XML технологија, на којим је заснован описани едитор, је могућност преузимања библиографских записа у виду XML документа који може представљати инстанцу различите XML шеме која описује библиографски запис.

Апликација је интегрисана у библиотечки информациони систем БИСИС верзија 4. Интеграцијом је омогућено претраживање библиотека које користе протокол Z39.50 и преузимање библиотечких записа. Преузети записи се складиште у објектну структуру која репрезентује библиографски запис чиме се омогућава даља обрада преузетог записа у оквиру едитор за обраду библиографске грађе система БИСИС.

ЛИТЕРАТУРА

- [1] Information Retrieval (Z39.50):Application Service Definition and Protocol Specification, <http://www.loc.gov/z3950/agency/Z39-50-2003.pdf>, погледано 14.10.2010
- [2] Z3950 servers, <http://isbnssearch.sourceforge.net/servers.html>, погледано 14.10.2010
- [3] Index Data Zebra, <http://indexdata.dk/zebra>, погледано 14.10.2010
- [4] Voyager Integrated Library System, <http://www.exlibrisgroup.com/category/Voyager>, погледано 14.10.2010
- [5] MarcEdit, <http://www.oregonstate.edu/~reese/marcedit/html/index.html>, погледано 14.10.2010
- [6] Integral Concepts ZSearcher, <http://www.biblio-tech.com/html/zsearcher.html>, погледано 14.10.2010
- [7] Surpass Copycat, <http://www.surpasssoftware.com/copycat.htm>, погледано 14.10.2010
- [8] BookWhere, WebClarity Software Inc, <http://www.bookwhere.com>, погледано 14.10.2010

- [9] Combs, Jr. J., (1997) "VIZION with Z39.50: A professional research tool from SIRSI", *Library Software Review*, vol. 16, no.2, pp. 108
- [10] Combs, Jr. J., (1997), "WinWillow--Willow for Microsoft Windows: A Z39.50 client from the University of Washington", *Library Software Review*, vol. 16, no.3, pp. 188
- [11] Combs, Jr. J., (1997), "Znavigator: A Z39.50 search and retrieval client for Windows from Enware", *Library Software Review*, vol. 16, no.4, pp. 247
- [12] Combs, Jr. J., (1998), "BookWhere? 2000 for Windows 95 and NT: An Internet search and retrieval product from SeaChange Corporation", *Library Software Review*, vol. 17, no.1, pp. 64
- [13] East, J. W., (2003), "Z39.50 and personal bibliographic software", *Library Hi Tech*, vol. 21, no. 1, pp. 34-43
- [14] Combs, Jr. J., (1998) "Z39.50: A survey of selected clients for the Microsoft Windows, Windows 95, and Windows NT Operating environment", *Library Software Review*, vol. 17, no.4, pp. 295
- [15] Turner, F. (1998), "Selecting a Z39.50 client or Web gateway", *Library Hi Tech*, issue 62
- [16] Dorner, G. D., Curtis, A., (2004), "A comparative review of common user interface products", *Library Hi Tech*, vol. 22, no. 2, pp 182-197
- [17] Baranov, V., Plemnek, A., Riabev, V., Sokolova, N., Sova, D., Usmanov, R., (2000), "Review of Z39.50 servers and Z39.50 environment in Russia", *Library Hi Tech*, vol. 18, no. 4, p.p 304-314
- [18] Evans, F. M., Sian, T., (2007), "Implementation of an integrated information management system at the National Library of Wales", *Program: electronic library and information systems*, vol. 41, no. 4, pp. 325-337
- [19] Kapidakis, S., (2002), "Document delivery system for Greek libraries", *The Electronic Library*, vol. 20, no. 6, pp. 496 - 503
- [20] Библиотечки информациони систем БИСИС, www.bisis.uns.ac.rs, погледано 14.10.2010
- [21] Dimić B., Surla D., XML Editor for UNIMARC and MARC 21 cataloguing, *The Electronic Library*, 2009, Vol. 27, No. 3, pp. 264-473
- [22] Dimić, B., Milosavljević, B., and Surla, D. "XML schema for UNIMARC and MARC 21 formats", *The Electronic Library*, 2010, Vol. 28, No 2
- [23] Milosavljević B., Dimić B., XML schema of UNIMARC format variant and bibliographic record in BISIS software system, *NSJOM*, Vol. 37, No.1, 2007, pp 115-128.
- [24] Milosavljević, B., Boberić, D., Surla, D., (2010) „Retrieval of Bibliographic Records Using Apache Lucene”, *The Electronic Library*, Vol. 28, No. 4, pp.525-539
- [25] Tešendić, D., Milosavljević, B., Surla, D., A Library Circulation System for City and Special Libraries, *The Electronic Library*, Vol. 27, No. 1, 2009, pp. 162 – 186.
- [26] Milosavljević, B., Tešendić, D., Software Architecture of Distributed Client/Server Library Circulation System, *The Electronic Library*, Vol. 28, No. 2, 2010, pp. 286-299.
- [27] Tešendić, D., A database model for library material usage, *Novi Sad Journal of Mathematics*, Vol. 37, No.1, 2007, pp 155-162.
- [28] Boberić, D., Surla, D., (2009), "XML Editor for Search and Retrieval of Bibliographic Records in the Z39.50 Standard", *The Electronic Library*, Vol. 27, No. 3
- [29] Боберић, Д., Сурла, Д., (2007), "Преузимање библиографских записа по Z39.50 стандарду", монографија, Природно-математички факултет Нови Сад, <http://www.bisis.ns.ac.yu/wiki/Publikacije>
- [30] Dovey J. M., Tatham, C., Corfield, A., (2005), "JAFER Toolkit Project", *VINE*, vol. 35, no. ½, pp.49 - 51
- [31] Attribute Set bib-1, <http://www.loc.gov/z3950/agency/defns/bib1.html>, погледано 14.10.2010



Др Данијела Боберић Крстићев, доцент,
Природно-математички факултет, Нови Сад
Контакт: dboberic@uns.ac.rs
Област интересовања: Информациони системи, Веб сервиси

