

МОДЕЛИРАЊЕ СИСТЕМА ЗА ФОРМИРАЊЕ КУРИКУЛУМА СТУДИЈСКИХ ПРОГРАМА¹ MODELLING OF SYSTEM FOR CREATING CURRICULUM

Коса Ненадић, Душан Сурла

РЕЗИМЕ: Део студијског програма на факултетима односи се на Курикулум студијског програма. Циљ овог рада је моделирање информационих захтева за имплементацију едитора за XML документа Курикулум студијског програма. Коришћена је објектно оријентисана методологија за пројектовање информационих система. Основни допринос рада је у архитектури софтверског система која је заснована на XML документима Курикулума студијског програма и омогућава интеграцију овог едитора у различите информационе системе студентске службе. Улазна информација система за едитовање је XML шема Курикулума студијског програма док је излазна информација формирана XML документ Курикулума студијског програма. Овај XML документ може се складиштити у различите софтверске системе за чување и претраживање података.

КЉУЧНЕ РЕЧИ: UML, XML, XML Schema, XMLBeans

ABSTRACT: Syllabus of faculty includes Curriculum as one of its parts. The main goal of this paper is modeling of information requirements in order to implement editor for XML curriculum documents. Object-oriented methodology is used for modeling of information systems. The major contribution of this work is in the system model architecture that is based on the XML curriculum documents. Also, the system model architecture enables integration of the editor into various information systems of student service. The XML schema of the Curriculum is the input information into the editor and the output information is XML curriculum document which presents instance of given XML schema. This XML document can be stored in diverse software systems for storing and searching data.

KEY WORDS: UML, XML, XML Schema, XMLBeans.

1. УВОД

Студијски програми факултета дефинисани су статутом факултета и статутом универзитета који су у сагласности са законом о високом образовању Републике Србије. Факултети, на основу специфичности своје делатности, могу дефинисати Курикулуме студијских програма на различите начине. Детаљно су анализирани Курикулуми студијских програма на Природно-математичком факултету у Новом Саду и упоређивани са Курикулумима студијских програма других факултета. Начин задавања и структура Курикулума студијских програма разликује се по факултетима као и у оквиру факултета.

Постоји више информационих система за електронско пословање студентске службе који су сервисно оријентисани. На пример, Oracle је развио апликације за потребе високо школских информационих система [1]. Компанија Eduventures [2] се такође бави унапређењем постојећих информационих система високо-школских институција. IBM је у сарадњи са Бостонским универзитетом направио план у развоју будућих информационих система [3].

На Природно-математичком факултету у Новом Саду такође се развија информациони систем студентске службе. У оквиру развоја овог система објављено је више радова. Део радова односи се на управљање документима који су прописани законом, као што су пријавни лист за упис студената и матична књига студената. У раду [4] описана је XML шема за пријавни лист за упис студената. На основу ове шеме могуће је генерисати потпуно исти изглед

оригиналног штампаног обрасца. На основу ове шеме у раду [5] описана је веб апликација за попуњавање односно ажурирање пријавног листа за упис студената. Архитектура ове апликације је таква да се може користити у оквиру различитих информационих система студентске службе, као и за електронску размену XML докумената пријавних листова за упис студената и статистичку обраду ових докумената. Предмет истраживања у раду [6] је моделирање и имплементација система за вођење матичне књиге студената. Систем се састоји од Интернет апликације за вођења матичне књиге студената и веб сервиса који омогућава коришћење овог система од стране других апликација. Слој апликације који се односи на пословну логику је базиран на XML шеми обрасца матичне књиге студената.

Циљ истраживања у тези [7] је моделирање и имплементација система за архивирање докумената студентске службе. Систем за архивирање је реализован као посебна софтверска компонента, а приступ систему је решен у виду веб сервиса. Различита софтверска решења студентске службе могу да користе систем за архивирање докумената. У раду [8] приказано је моделирање и имплементација система за електронско пријављивање испита. Верификација ових истраживања урађена је на примеру информационог система студентске службе и њеног подсистема испитна пријава.

У овом раду дата је спецификација информационих захтева за формирање Курикулума студијског програма. Спецификација је урађена у UML 2.0 нотацији.

¹ Овај рад је део истраживачког пројекта Апстрактни модели и апликације у рачунарској науци, подржаног од стране Министарства за науку Републике Србије (Пројекат бр.144017)

2. КУРИКУЛУМ СТУДИЈСКОГ ПРОГРАМА

Студијски програми детаљно су описани у закону о високом образовању Републике Србије [9]. У овом закону дефинисан је студијски програм у члану 24, који гласи: «Студијски програм је скуп обавезних и изборних студијских подручја, односно предмета, са оквирним садржајем, чијим се савладавањем обезбеђују неопходна знања и вештине за стицање дипломе одговарајућег нивоа и врсте студија».

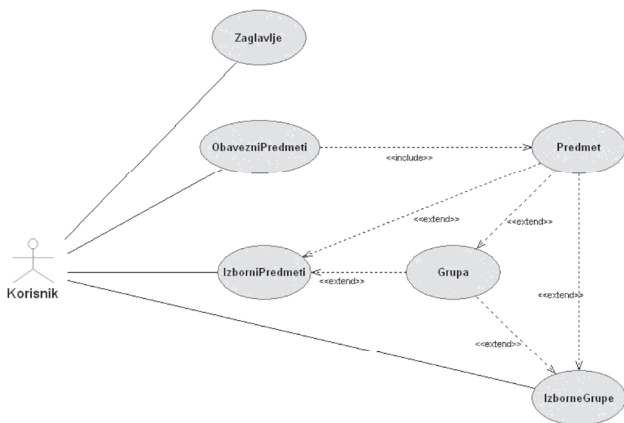
Садржај студијског програма дефинисан је чланом 28. Део тог садржаја је Курикулум студијског програма, који садржи листу и структуру обавезних и изборних предмета и модула и њихов опис. Детаљном анализом Курикулума студијских програма на факултетима Универзитета у Новом Саду (посебно ПМФ-а) уочена је следећа структура концепата Курикулума студијских програма:

1. Заглавље
2. Предмети
 - 2.1. Обавезан предмет
 - 2.2. Изборна група
 - 2.2.1. Предмет изборне групе
 - 2.2.2. Подгрупа
 - 2.3. Изборни предмети
 - 2.3.1. Изборни предмет
 - 2.3.2. Списак изборних предмета
 - 2.3.2.1. Предмет списка
 - 2.3.3. Списак група изборних предмета
 - 2.3.3.1. Група изборних предмета
3. Семестри
4. Напомена

Семестри и Напомена појављују се у оквиру више наведених концепата. Заглавље садржи информације о идентификацији студијског програма. Детаљан опис наведених концепата дат је у наредном одељку у оквиру описа сличајева коришћења.

3. СЛУЧАЈЕВИ КОРИШЋЕЊА СИСТЕМА НАСТАВНИ ПЛАНОВИ

Случај коришћења: Заглавље. Ажурирање заглавља обухвата обраду општих података о студијском програму а то су: ознака односно шифра; назив; датум усвајања и датум



Slika 1. – Дијаграм случајева коришћења система Курикулуми студијских програма

истицања (ако је датум познат). Ако је идентификован студијски програм за који је донета одлука о измени Курикулума студијског програма, потребно је ажурирати одговарајуће измене.

Случај коришћења: Обавезни предмети. Обавезни предмети представљају предмете који су обавезни по Курикулуму студијског програма. За ажурирање података о обавезним предметима укључује се случај коришћења *Предмет*.

Случај коришћења: Предмет. Ажурирање предмета обухвата обраду следећих података о предмету: шифра, назив, семестар, број часова, Европски систем преноса бодова (ЕСПБ). Опционо се наводе опис предмета и напомена.

Случај коришћења: Изборне групе. Изборне групе наставног плана представљају групе предмета из којих је неопходно положити одређен број предмета. Корисник уноси податке о изборним групама реализујући случај коришћења *Група*.

Случај коришћења: Група. У Курикулумима наставних програма предмети могу бити организовани у групе предмета. За сваку групу наводи се назив групе, број предмета и/или (под)група који се бира из групе, предмети и/или (под)групе предмета који јој припадају. Ако група садржи предмете који нису даље издељени на групе реализује се случај коришћења *Предмет*. Ако су предмети издељени на групе, врши се њихово груписање и реализује случај коришћења *Група*. Дозвољен је унос, модификација и брисање група.

Случај коришћења: Изборни предмети. Ажурирање листе изборних предмета по годинама обухвата обраду података о изборним предметима посматране школске године. Наводе се назив изборног предмета, његов опис, подаци о семестрима у којима се изборни предмет слуша и евентуалне напомене. За изборне предмете једне школске године везује се листа предмета који се бирају као изборни (реализована случајем коришћења *Предмет*) или листа изборних група предмета (реализована случајем коришћења *Група*).

4. ДИЈАГРАМ КЛАСА КУРИКУЛУМА СТУДИЈСКОГ ПРОГРАМА

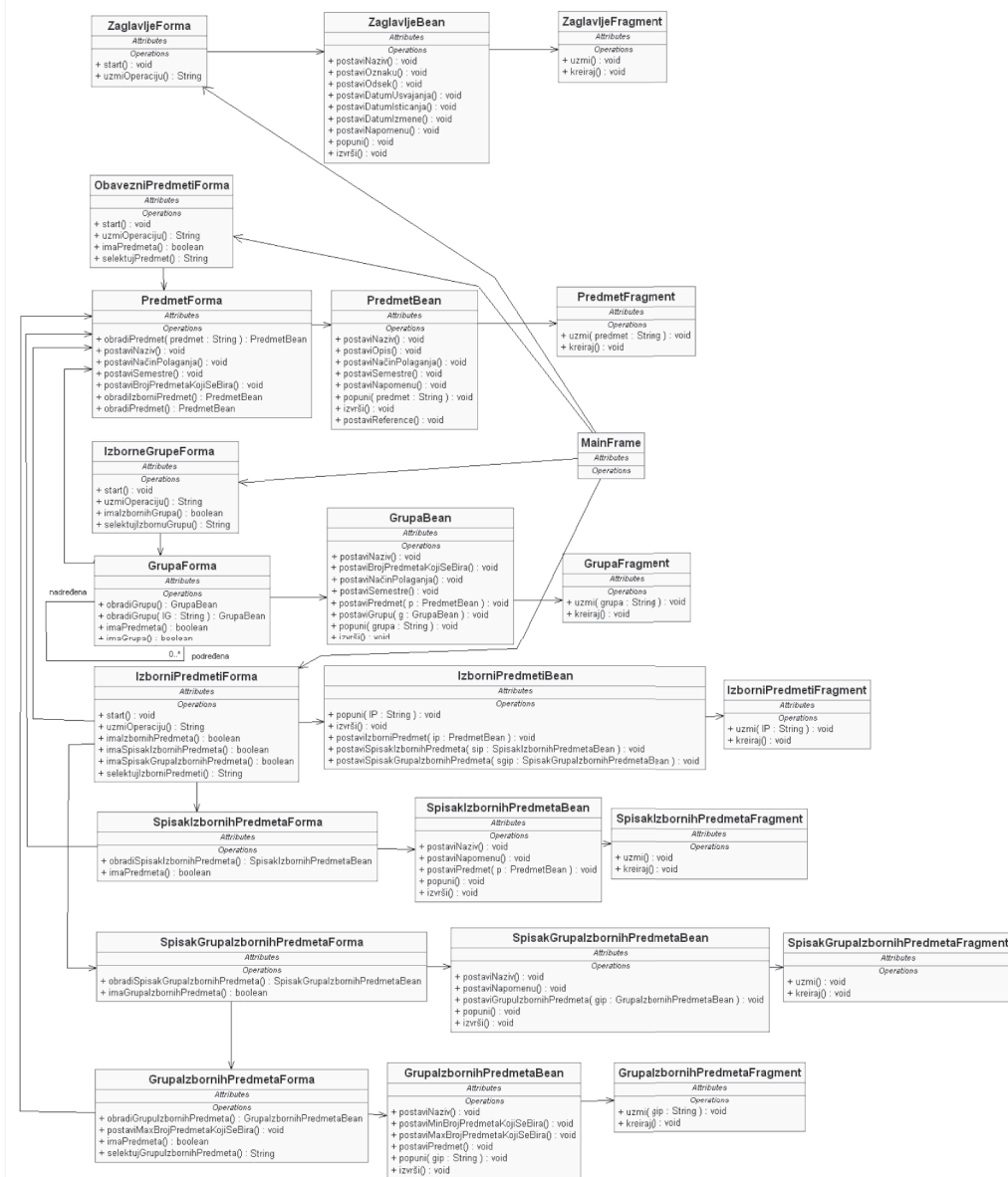
На слици 2 приказан је дијаграм класа система Курикулуми студијских програма. На основу функције коју обављају класе система подељене су у три целине (*xxxForma*, *xxxBean* и *xxxFragment*).

Класа *MainFrame* је полазна класа система.

Класе система чији је назив облика *xxxForma* реализују кориснички интерфејс омогућавајући рад са одређеним елементима Курикулума студијских програма.

Класама облика *xxxBean* реализовани су подаци који се ажурирају на формама и наведени су у опису случајева коришћења.

Класе система облика *xxxFragment* служе за складиштење ажурираних података у изабраном формату.



Slika 2. – Дијаграм класа система Курикулума студијских програма

5. ДИЈАГРАМИ СЕКВЕНЦИ СИСТЕМА КУРИКУЛУМИ СТУДИЈСКИХ ПРОГРАМА

На основу приказаног дијаграма случајева коришћења система Курикулуми студијских програма (слика 1) и дијаграма класа (слика 2) креирани су дијаграми секвенци који описују одређене сценарије случајева коришћења. Дијаграми секвенци су моделирани у UML 2.0.

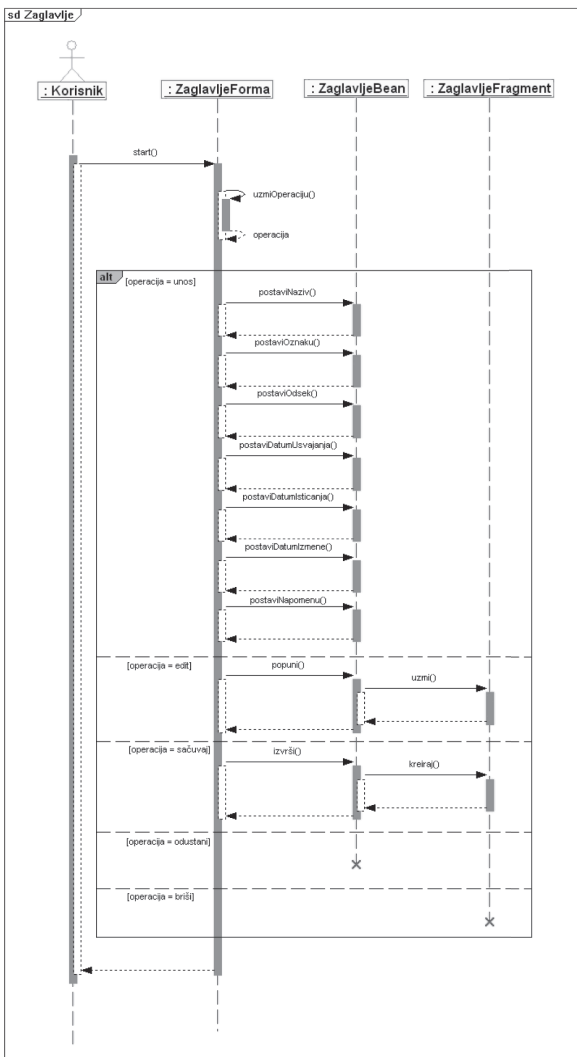
Креирање дијаграма секвенци је урађено у Embarcadero Describe Enterprise 6.1.7 [10] развојном окружењу које подржава UML 2.0 спецификацију и омогућава анализу и дизајн заснован на моделу. Креирани су следеће дијаграми секвенци: *Zaglavlje*, *ObavezniPredmeti*, *Predmet*, *IzborneGrupe*, *Grupa*, *IzborniPredmeti*, *SpisakIzbornihPredmeta*, *SpisakGrupalzbornihPredmeta* и *GrupalzbornihPredmeta*.

Дијаграм секвенци *Zaglavlje*

На слици 3 приказан је дијаграм секвенци *Zaglavlje*. Дијаграм описује ажурирање података о образовном програму.

Корисник апликације приступа објекту *ZaglavljeForma*, на коме селекује операцију захтевом *uzmiOperaciju()*. Од избора операције зависи секвенца порука која се потом извршава (комбиновани фрагмент *алтернатива* моделира задате услове).

Избором операције *unos* извршава се секвенца порука којом се постављају вредности општих података о образовном програму у објекат *ZaglavljeBean* захтеви-ма *postaviNaziv()*, *postaviOznaku()*, *postaviOdsek()*, *postaviDatumUsvajanja()*, *postaviDatumIsticanja()*, *postaviDatumIzmene()*, и *postaviNapomenu()*. Улога појаве класе *ZaglavljeBean* је да опште податке о образовном програму држи ажурним за приказ на форми.



Slika 3. – Дијаграм секвенци Zaglavlje

Избором операције *edit ZaglavljeForma* шаље захтев *popuni()* објекту *ZaglavljeBean*, који иницира захтев *uzmi()* за преузимањем садржаја *ZaglavljeFragment-a*. Овим се објекат *ZaglavljeBean* попуњава подацима из *ZaglavljeFragment-a* и приказује на форми.

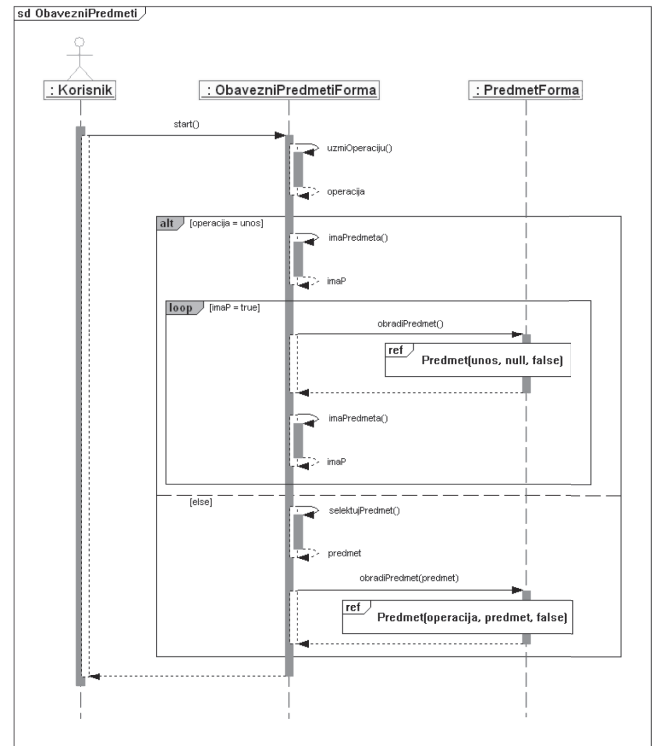
Избором операције *sačuvaj* извршава се захтев *izvrši()* објекта *ZaglavljeBean* који захтевом *kreiraj()* иницира креирање *ZaglavljeFragment-a* у којем се чувају подаци. Из објекта *ZaglavljeFragment* је после могуће трансформисати податке у друге формате.

Операција *odustani* изазива уништење објекта *ZaglavljeBean*, чиме се подаци уклањају са форме за приказ.

Операција *briši* уништава податке меморисане у објекту *ZaglavljeFragment*.

Дијаграм секвенци *ObavezniPredmeti*

На слици 4 описан је поступак ажурирања предмета обавезних по Курикулуму студијског програма. Корисник приступа форми за унос обавезних предмета класе *ObavezniPredmetiForma* и бира операцију захтевом *uzmiOperaciju()*.



Slika 4. – Дијаграм секвенци *ObavezniPredmeti*

Ако је одабрана операција *unos*, тада се порука *obradiPredmet()* шаље објекту *PredmetForma* све док има предмета за унос, и при томе се позива дијаграм секвенци *Predmet* са параметрима *unos, null* и *false*.

Засве остале операције врши се одабир предмета поруком *selektujPredmet()* и шаље порука *obradiPredmet(predmet)* објекту *PredmetForma*. Потом се позива дијаграм секвенци *Predmet* са параметрима *operacija, predmet* и *false*, којим се врши обрада селектованог предмета задатом операцијом.

На сличан начин могу се описати и остали наведени дијаграми секвенци.

На основу описног модела у раду [11] детаљно је приказана имплементација веб апликација за едитовање Курикулума студијских програма. За имплементацију је коришћена XML технологија [12,13]. и Јава окружење [14]. Као развојно окружење искоришћен је IBM WebSphere Studio Application Developer који представља комплетно, интегрисано развојно окружење за визуелно дизајнирање, креирање, тестирање и имплементирање веб сервиса, портала и J2EE апликација [15, 16]. Имплементација је реализована у овом окружењу, које је изграђено на отвореној Eclipse Workbench платформи за развој алата, чиме је омогућено његово прилагођавање и проширење како IBM тако и Eclipse plug-in алатима.

6. ЗАКЉУЧАК

Циљ истраживања описаних у овом раду је моделирање едитора за XML документе Курикулума студијских програма. Моделирање је реализовано као подсистем тако да се може интегрисати у различите информационе система студентске

службе. Моделом едитора омогућено је формирање докумената Курикулума студијских програма као XML докумената. За конкретну реализацију складиштења података о наставним плановима потребно је само трансформисати податке из XML-а у циљни модел за складиштење података. На основу моделираног система имплементирана је веб апликација за едитовање Курикулума студијских програма.

ЛИТЕРАТУРА

- [1] Higher Education Institutions Expand Services, http://www.oracle.com/corporate/press/2006_may/he-momentum-rls.html [октобар 2006]
- [2] Service-Oriented Architecture and Web Services: The Next Big Thing in University Enterprise Computing, <http://www.oracle.com/industries/education/eduventures-service-oriented-architecture-and-web-services.pdf>, [октобар 2006]
- [3] Student Service System – Next Generation, http://www-03.ibm.com/industries/education/doc/content/bin/IBM_BCS_White_Paper_Student_Services_System_FINAL.pdf, [октобар 2006]
- [4] M. Mirković, D. Surla, XML schema of entry form for students, NSJOM Vol 37. No. 1 pp. 75-84.
- [5] M. Mirković, D. Surla, Modelling and implementation of the application for students enroll form update, NSJOM (prihvaćen za štampu) .
- [6] Ћук, Д., *Софтверски систем за вођење матичне књиге студената*, Магистарска теза, прихваћена тема магистарске тезе, Природно-математички факултет, Департман за математику и информатику, Нови Сад, 2006.
- [7] Шумић, Д., *Архивирање XML докумената студентске службе*, Магистарска теза, Природно-математички факултет, Департман за математику и информатику, Нови Сад, 2006.
- [8] Стојадиновић, Ј., В., Тодорић-Вукашин, Д., *Web апликација за пријављивање испита*, 31. Симпозијум о операционим истраживањима, SYM-OP-IS 04, Иришки венац, 2004, зборник радова, стр. 157-160.
- [9] Закон о високом образовању, службени гласник 76/05, 30.8.2005.
- [10] Describe, Embarcadero, version 6.1.7, <http://www.embarcadero.com/products/describe/dedatasheet.html>, [октобар 2006]
- [11] Ненадић, К., *Едитор за XML документа наставних планова*, Магистарска теза, Природно-математички факултет, Департман за математику и информатику, Нови Сад, 2006.
- [12] The Apache XML Project, Apache XMLBeans, <http://xmlbeans.apache.org/index.html> [јануар 2005.]
- [13] O. Takagiwa, D. Hassan, O. Quixchan, J. Singh, *The XML Files: Development of XML/XSL Applications Using WebSphere Studio Version 5*, 2002, <http://www.redbooks.ibm.com>
- [14] U. Wahli, M. Fielding, G. Mackown, D. Shaddon, G. Hekkenberg, *Servlet and JSP Programming with IBM WebSphere Studio and VisualAge for Java*, 2000, <http://www.redbooks.ibm.com>
- [15] U. Wahli, I. Brown, F. Ferraz, M. Schumacher, H. Sjostrand, *WebSphere Studio Application Developer Version 5 Programming Guide*, 2003, <http://www.redbooks.ibm.com>
- [16] C. Sadtler, J. Bogers, S. Borkar, A. Firtiyan, *IBM WebSphere Application Server - Express V5.0.2 Developer Handbook*, 2003, <http://www.redbooks.ibm.com>



Коса Ненадић
ДДОР Нови Сад а.д.о. Дирекција за информатику,
Област: Информациони системи



Душан Сурла
Природно-математички факултет
Област: Информациони системи

