

DOKUMENTACIJA TEHNIČKOG REŠENJA

Grafički editor za formalni opis instrukcione strategije

Autori tehničkog rešenja

Goran Savić, Milan Segedinac, Zora Konjović

Godina kada je tehničko rešenje urađeno

2012.

Oblast tehničkog rešenja

Tehničko rešenje po Pravilniku Ministarstva prosvete, nauke i tehnološkog razvoja Republike Srbije pripada kategoriji **M85** (Tehnička i razvojna rešenja - Prototip, nova metoda, softver, standardizovan ili atestiran instrument, nova genska proba, mikroorganizmi). Tehničko rešenje pripada naučnoj oblasti Informatika, uža naučna oblast elektronska nastava.

Apstrakt

Tehničko rešenje je grafički editor za kreiranje dijagrama koji predstavljaju formalni opis instrukcione strategije definisane u skladu sa E-learning Instructional Design Language (ELIDL) jezikom. ELIDL je XML bazirani meta jezik, koji omogućuje mašinski čitljiv opis instrukcione strategije u nastavnom kursu. Jezik omogućuje formalno definisanje organizacije nastavnog procesa u kursu na meta nivou, nezavisno od sadržaja konkretnog nastavnog kursa.

XML notacija na kojoj je ELIDL baziran, iako veoma pogodna za mašinsko procesiranje, nije pogodna za ručno uređivanje. Iz tog razloga, razvijen je ovde opisani grafički editor za definisanje instrukcione strategije u vizuelnoj notaciji. Editor omogućuje korisniku da grafički definiše instrukcionu strategiju raspoređivanjem grafičkih elemenata na dijagram. Ovako kreirani dijagram se skladišti u ELIDL formatu. Na ovaj način, korisnik može formalno definisati instrukcionu strategiju bez poznavanja detalja ELIDL sintakse i XML formata.

Editor je implementiran kao *stand-alone* aplikacija u programskom jeziku Java u razvojnom okruženju Eclipse Helios verzije 3.6 (Eclipse Foundation, 2010). Implementacija je izvršena korišćenjem Eclipse GMF tehnologije (Eclipse Foundation, 2012a).

Primena ovog tehničkog rešenja omogućava da se na formalan i mašinski čitljiv način definiše instrukciona strategija na kojoj je nastavni kurs baziran. Ovim je omogućeno mašinsko procesiranje instrukcione strategije, što posebnu primenu nalazi u domenu elektronske nastave. Korišćenjem prikazanog tehničkog rešenja, formalno definisanje instrukcione strategije dostupno je i osobama bez posebnog tehničkog znanja. Ovo je

naročito važno, obzirom da se razvojem instrukcionih strategija najčešće bave stručnjaci iz oblasti obrazovanja i pedagogije.

Tehničko rešenje je do sada primenjeno za automatsko generisanje elektronskog nastavnog kursa na bazi formalno definisane instrukcione strategije i za merenje efikasnosti različitih instrukcionih strategija. U oba slučaja, instrukcione strategije su formalno definisane korišćenjem ovog tehničkog rešenja.

Naučno-istraživački doprinos ovog tehničkog rešenja publikovan je u jednom naučnom radu u međunarodnom časopisu sa SCI liste (Savić et al., 2012), kao i u tri rada na međunarodnim konferencijama (Segedinac et al., 2010), (Savić et al., 2011), (Savić et al., 2013).

Tehničko rešenje se koristi u nastavi i u naučno-istraživačke svrhe na Katedri za informatiku na Fakultetu tehničkih nauka Univerziteta u Novom Sadu.

1. Uvod

Tehničko rešenje se odnosi na implementaciju računarski podržane nastave u nastavnom kursu. Preciznije, prikazano rešenje rešava probleme koji se tiču načina obavljanja nastave. Organizacija nastave definisana je instrukcionom strategijom u kursu. Instrukciona strategija podrazumeva specifične pedagoške principe na kojima je kurs baziran i definiše način izbora i organizacije nastavnih aktivnosti u kursu.

Pri implementaciji računarski podržane nastave neophodno je obezbediti mašinsko procesiranje nastavnog kursa. Prikazano tehničko rešenje omogućuje formalnu i mašinski čitljivu reprezentaciju instrukcionog dizajna u nastavnom kursu. Ovim je omogućeno mašinsko procesiranje određene instrukcione strategije. U okviru rešenja, instrukciona strategija se reprezentuje na meta nivou, nezavisno od konkretnih nastavnih aktivnosti i nastavnih sadržaja u kursu. Ovo obezbeđuje primenu jednom definisane instrukcione strategije na različitim domenima.

Ovde prikazano tehničko rešenje je softverski alat koji obezbeđuje vizuelno definisanje instrukcione strategije u grafičkom okruženju. Ovaj grafički editor obezbeđuje jednostavan i intuitivan način za formalno definisanje instrukcione strategije. Korišćenjem editora, definisanje instrukcione strategije postaje dostupno širokom skupu korisnika, bez potrebe poznavanja tehničkog formata namenjenog za mašinsko procesiranje.

2. Pregled postojećih rešenja

Instrukcioni dizajn se u (Reigeluth and Moore, 1999) definiše na generalnom nivou kao teoretski okvir koji pruža eksplicitne smernice o tome kako vršiti podučavanje. U (Frizell and Hubscher, 2011) autori konkretnije objašnjavaju da je instrukcioni dizajn proces razvoja instrukcionih sadržaja i nastavnih aktivnosti baziran na istraživanjima u oblasti teorije učenja. Proces sekvencionisanja i organizacije sadržaja, specifikacije nastavnih aktivnosti i odlučivanja kako će sadržaji i aktivnosti biti isporučeni se u (Dick and Carey, 1990) naziva instrukciona strategija. Ovde će biti dat pregled postojećih rešenja koja se

odnose na formalnu reprezentaciju instrukcionog dizajna, odnosno instrukcionih strategija u nastavnom kursu.

Instrukcioni dizajn se formalno reprezentuje putem jezika za instrukcioni dizajn (eng. *instructional design languages*). U (Botturi, 2006) je predstavljen E²ML – jednostavan jezik koji putem vizuelne notacije omogućuje definisanje nastavnih aktivnosti u kursu. Laforcade u (Laforcade, 2005) predstavlja CPM - UML-baziran jezik za vizuelno predstavljanje problem-baziranih nastavnih scenarija. coUML (Derntl and Motschnig, 2007) je drugi UML-baziran jezik za opis nastavnog procesa. Jezik predviđa definisanje globalnih informacija o kursu, učesnika u nastavi, nastavnih ciljeva, dokumenata koji se koriste u nastavi, strukture kursa i nastavnih aktivnosti u kursu. U (Caeiro-Rodríguez, 2008) je predstavljen PoEML, jezik za modelovanje obrazovnih jedinica. U okviru obrazovne jedinice, moguće je definisati pod-jedinice i njihov redosled, čime je opisan tok nastavnog procesa. LDL (Martel et al., 2006) je jezik koji je posebno namenjen za modelovanje različitih učesnika u nastavi i kolaborativnih nastavnih aktivnosti. MOT+ (Paquette et al., 2008) je generički vizuelni jezik i alat za predstavljanje instrukcionog dizajna. Korišćenjem MOT+ jezika moguće je vizuelno predstaviti sekvencu nastavnih aktivnosti, učesnike i alate u kursu. Learning Activity Management System (LAMS) (LAMS Foundation, 2005) omogućuje grafičko definisanje nastavnog procesa baziranog na nastavnoj aktivnosti kao njegovoj osnovnoj jedinici. SCORM standard (ADL, 2009), odnosno IMS SS specifikacija (IMS Global Learning Consortium, 2003a), kao njegov deo takođe omogućuju definisanje toka nastavnih aktivnosti u kursu. Još jedan globalno popularan jezik za reprezentaciju nastavnog procesa je i IMS Learning Design (IMS LD) (IMS Global Learning Consortium, 2003b).

Svi navedeni jezici definišu konkretne nastavne aktivnosti u kursu. Na ovaj način instrukcioni dizajn nije eksplicitno predstavljen nezavisno od konkretnog kursa i nije moguće samostalno upravljati instrukcionim dizajnom u kursu. Opisani jezici za instrukcioni dizajn definišu konkretne instance određenih instrukcionih strategija. Ovo tehničko rešenje se odnosi na eksplicitnu i formalnu reprezentaciju generičkih instrukcionih strategija. Generička instrukciona strategija predstavlja šablon nezavisan od konkretnog nastavnog kursa i njegovog sadržaja. Kada se nastavni proces posmatra apstraktno, moguće je nezavisno od konkretnih nastavnih aktivnosti definisati različite šablone instrukcionih strategija, koji su primenjivi na bilo koji konkretan nastavni kurs.

Definisanje generičkih instrukcionih šablona analizirano je u okviru projekta *Pedagogical Patterns* ("The Pedagogical Patterns Project," 2001). Projekat definiše kolekciju standardnih pedagoških šablona koji se koriste u nastavi. Šablon predstavlja skup univerzalnih pedagoških praksi koje se mogu primenjivati u različitim domenima na konkretnom kursu. Cilj je da u ovoj kolekciji nastavnik pronade odgovarajući šablon za izvođenje nastave na svom predmetu. Iako su veoma korisni za nastavnike i pedagoge, ovi šabloni nisu namenjeni za mašinsko procesiranje. Šabloni su reprezentovani u formi slobodnog teksta i ne postoji formalizam za njihov opis, tako da nisu mašinski-čitljivi. U radu (De Moura Filho and Derycke, 2008) opisan je način konvertovanja ovih pedagoških šablona iz tekstualnog oblika u mašinski čitljiv IMS LD jezik. Rezultat konverzije nije konkretan IMS LD kurs, već šablon koji reprezentuje apstraktne nastavne

aktivnosti. Ipak, IMS LD nije odgovarajući jezik za reprezentaciju šablona instrukcionog dizajna, obzirom da je IMS LD namenjen za definisanje konkretnog nastavnog procesa i nema mogućnost definisanja meta-modela kao apstraktnog pogleda na nastavni proces. Drugo istraživanje koje takođe formalizuje reprezentaciju pedagoških šablona iz Pedagogical Patterns projekta je opisano u (Ljubojevic and Laurillard, 2010). Autori predstavljaju strukturiran opis pedagoškog šablona. Iako je struktura šablona jasno definisana, ona se i dalje sastoji od tekstualnih polja, tako da nije pogodna za mašinsko procesiranje. U (Derntl and Motschnig-Pitrik, 2004) predstavljen je projekat PceL u okviru kojeg je kreiran repozitorijum šablona za opis nastavnog procesa. Repozitorijum sadrži predefinisani skup šablona definisanih u UML notaciji. U okviru konkretnog kursa identifikuju se specifični šabloni u toku nastavnog procesa i mapiraju na neki od šablona iz repozitorijuma. U projektu AUTC (AUTC, 2003) razvijeno je 5 reprezentativnih šablona za instrukcioni dizajn. Grafički i tekstualno u formi prirodnog teksta je dat vodič za implementaciju nastavnih kurseva na osnovu datih šablona. U (Guangzuo et al., 2009) je predstavljen semantički model za instrukcioni dizajn. Model sadrži podatke o stanju znanja učenika, nastavnim ciljevima, nastavnom materijalu i instrukcionoj strategiji. Instrukciona strategija je definisana u vidu *if-then* pravila. Na osnovu trenutnog učenikovog znanja i metapodataka definisanih u okviru nastavnog cilja i nastavnog objekta, bira se sekvenca akcija. Akcije su definisane na osnovu Žaneovog modela o 9 instrukcionih aktivnosti. Međutim, raspoloživ je samo predefinisani skup instrukcionih strategija. U (Hayashi et al., 2009) predstavljen je metod za formalno definisanje apstraktnih instrukcionih strategija putem ontologije. Strategija definiše tok nastavnih aktivnosti, međutim komplikovanije kontrole toka kao što su uslovna grananja nisu podržane. LAMS projekat je razvio alat Activity Planner (LAMS Foundation, 2012) koji omogućuje definisanje LAMS kursa na apstraktnom nivou. Korisnik ne navodi konkretne aktivnosti u kursu, već kreira šablon definisanjem sekvence apstraktnih nastavnih aktivnosti. Definisani šablon je LAMS-specifičan i namenjen je da bude osnov za kreiranje konkretnog LAMS kursa.

Analizirajući reprezentaciju instrukcionog dizajna u navedenim istraživanjima, mogu se uočiti dve vrste problema: 1) opis instrukcionog dizajna nije dovoljno strukturiran da bi bio pogodan za mašinsko procesiranje i/ili 2) opis instrukcionog dizajna nije dovoljno ekspresivan da opiše raznolike instrukcione tehnike (u većini pristupa se koristi ograničen, predefinisani skup tehnika u okviru kojih nastavnik može da izabere odgovarajuću). Da bi se komponenta koja opisuje instrukcioni dizajn mogla mašinski procesirati, nezavisno od ostalih komponenti kursa, potrebno je eksplicitno i formalno reprezentovati instrukcioni dizajn u mašinski čitljivom formatu.

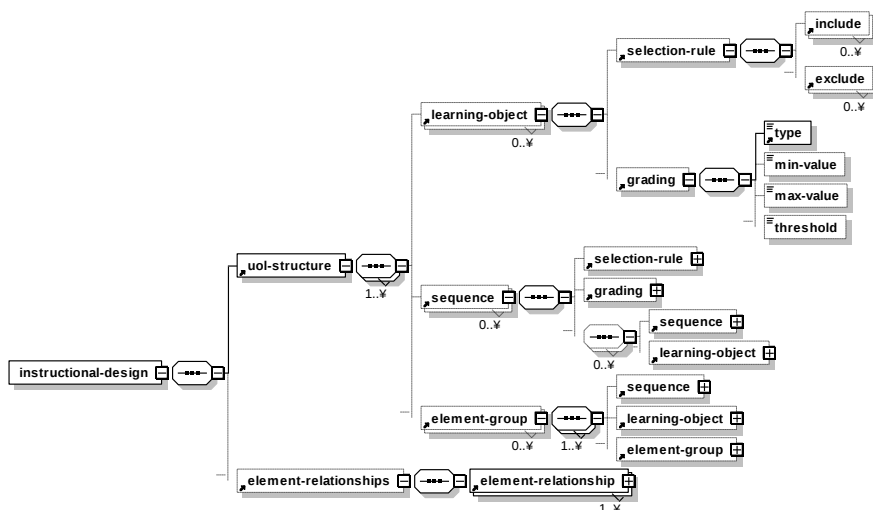
3. ELIDL jezik

Imajući u vidu navedene nedostatke trenutno zastupljenih rešenja, razvijen je novi meta jezik za opis instrukcionog dizajna u mašinski čitljivom obliku. Jezik je nazvan ELearning Instructional Design Language (ELIDL) (Savić et al., 2012) i poseduje sledeće osobine:

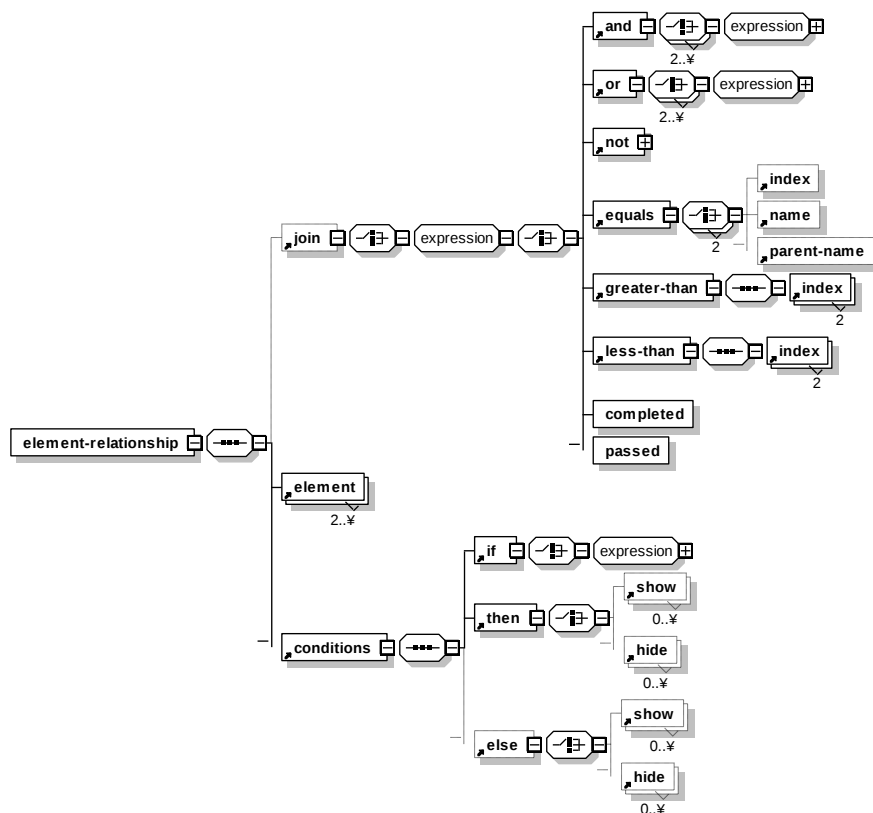
- 1) generičnost – instrukciona strategija definisana ELIDL jezikom je nezavisna od sadržaja konkretnog kursa i formata u kojem će taj kurs biti reprezentovan
- 2) ekspresivnost – ELIDL omogućuje definisanje širokog skupa različitih instrukcionih strategija

- 3) mašinska čitljivost – instrukcionu strategiju predstavljenu ELIDL jezikom je moguće mašinski procesirati

ELIDL putem XML notacije omogućuje definisanje načina izbora i organizacije nastavnih sadržaja u kursu. XML šema ELIDL jezika prikazana je na slikama 1 i 2.



Slika 1. XML šema ELIDL jezika – prvi deo



Slika 2. XML šema ELIDL jezika – drugi deo

U nastavku će ukratko biti objašnjeni elementi šeme. Korenski element `instructional-design` sadrži dva podelementa, `uol-structure`, za definisanje nastavnih aktivnosti u kursu, i `element-relationships`, za definisanje relacija između nastavnih elemenata. Nastavne aktivnosti se definišu elementima `sequence` ili `learning-object`. Element `sequence` predstavlja sekvencu nastavnih elemenata. Uloga ovog elementa je slična ulozi petlji u programskim jezicima. Element `sequence` definiše prolazak kroz sve nastavne sadržaje definisane atributom `element`. Tako, na primer, možemo definisati sekvencu lekcija, tema ili nastavnih objekata. Element `learning-object` predstavlja jedan konkretan nastavni objekat koji treba da se nalazi u kursu. I `learning-object` i `sequence element` sadrže `selection-rule element`. Ovaj element specificira koji će nastavni resursi i kojim redom biti izabrani iz skupa nastavnih resursa mapiranih na određeni nastavni cilj. Izbor i prioritet elemenata se definiše elementima `include`, odnosno `exclude`, koji su podelementi `selection-rule elementa`. Pri izboru elemenata, ne navode se konkretni nastavni sadržaji, već se izbor vrši na generičkom nivou na osnovu osobina nastavnog sadržaja definisanih metapodacima. Element `grading` definiše da li se i kako vrši ocenjivanje za nastavni objekat.

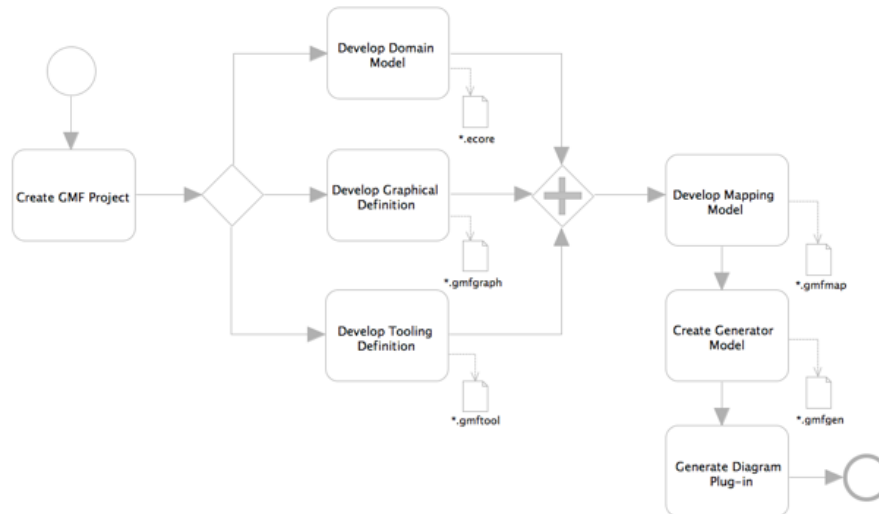
Element `element-relationships` služi za definisanje veza između nastavnih aktivnosti u kursu. Npr. između teorijskog testa i projektnog zadatka može postojati veza. Projektni zadatak je studentima dostupan samo ako su savladali test. Ovakva veza se može definisati putem `element-relationship elementa`. Elementi pod nazivom `element` definišu elemente koji ulaze u vezu. Od definisanih elemenata, veze će se formirati samo za one elemente koji ispunjavaju uslov definisan u `join elementu`. Element `conditions` sadrži niz `if-then-else` elemenata koji definišu specifične akcije koje će biti izvršene nad elementima u slučaju da je uslov ispunjen, odnosno nije ispunjen. Detaljan opis svih elemenata šeme dostupan je u (Savić, 2010a).

4. Implementacija grafičkog editora

Da bi mogućnost formalnog definisanja instrukcionog dizajna bila dostupna širokom skupu korisnika, potrebno je obezbediti jednostavan i intuitivan način koji ovo omogućuje. Postoje jezici u kojima se instrukcioni dizajn definiše grafički i to su takozvani vizuelni jezici za instrukcioni dizajn. Od jezika pomenutih u drugom poglavlju, u ovu grupu spadaju E2ML, CPM, coUML, PoEML i MOT+. Kod jezika koji nisu vizuelni, često se definisanje instrukcionog dizajna korisniku olakšava putem specijalizovanih grafičkih alata. Postoje razvijeni editori za SCORM i IMS LD. Takođe, za formalnu reprezentaciju instrukcionog dizajna prikazanu u (Hayashi et al., 2009) razvijen je alat nazvan SMARTIES.

XML notacija na kojoj je ELIDL baziran, iako veoma pogodna za mašinsko procesiranje, nije pogodna za ručno uređivanje. Zato je razvijen grafički editor, koji omogućuje definisanje instrukcionog dizajna u ELIDL jeziku u vizuelnoj notaciji. Korisnik definiše instrukcioni dizajn raspoređivanjem odgovarajućih grafičkih elemenata, a kreirani dijagram se automatski snima u ELIDL formatu. Na ovaj način korisnik može formalno definisati instrukcioni dizajn bez poznavanja detalja ELIDL sintakse i XML formata.

Editor je kreiran korišćenjem Eclipse Graphical Modeling Framework (GMF) tehnologije. Tehnologija je zasnovana na modelom upravljanim pristupu i omogućuje relativno jednostavan razvoj grafičkog editora u programskom jeziku Java u okviru razvojnog okruženja Eclipse (Eclipse Foundation, 2012b). Slika **Error! Reference source not found.**3 prikazuje postupak razvoja editora u GMF tehnologiji.

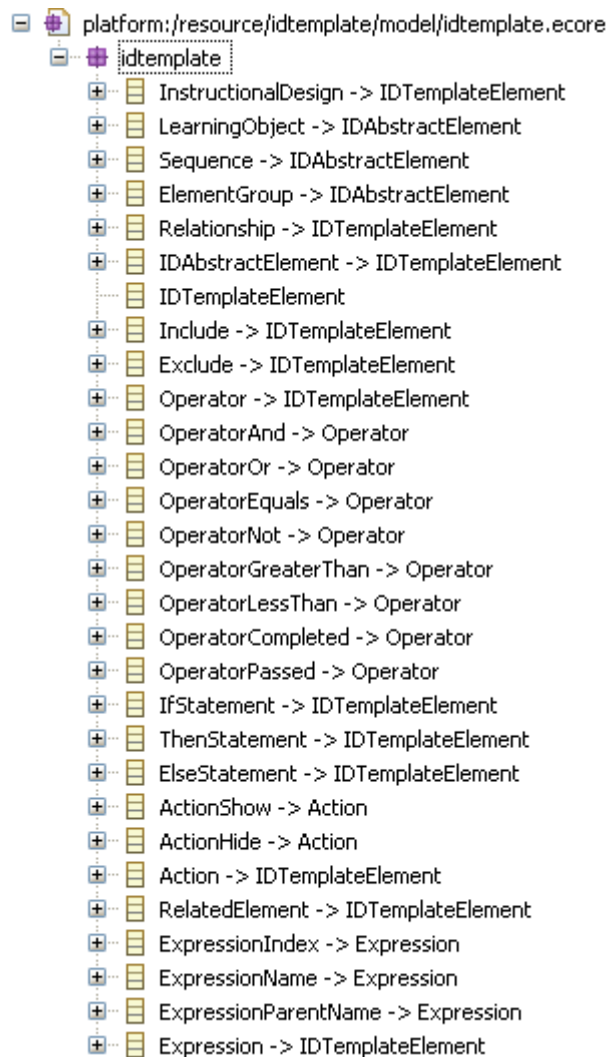


Slika 3. Proces razvoja grafičkog editora u GMF tehnologiji (Eclipse Foundation, 2012c)

GMF se bazira na dva razvojna okvira – Eclipse Modeling Framework (EMF) (Eclipse Foundation, 2012d) i Graphical Editing Framework (GEF) (Eclipse Foundation, 2012e). EMF je tehnologija za definisanje domenskog modela i generisanje programskog koda i alata koji omogućuju pristup modelu. GEF tehnologija omogućuje definisanje grafičkih komponenti editora. Kao što se može videti na slici 3 u okviru GMF projekta potrebno je definisati domenski model, grafičke komponente i alate za interakciju sa korisnikom. Zatim je potrebno izvršiti međusobno mapiranje ove tri komponente projekta. Na osnovu definisanih parametara, GMF vrši generisanje programskog koda grafičkog editora. Izvršavanjem generisanog koda, pokreće se aplikacija koja predstavlja grafički editor.

U skladu sa prikazanim procesom razvoja, pri kreiranju grafičkog editora za ELIDL, najpre je u EMF tehnologiji kreiran domenski model instrukcionog dizajna. Domenski model je kreiran u skladu sa XML šemom ELIDL jezika. Model sadrži elemente XML šeme ELIDL jezika i odgovarajuće relacije između njih.

Na slici 4 grafički je prikazan sadržaj fajla `idtemplate.ecore` koji predstavlja EMF domenski model.



Slika 04. EMF domenski model instrukcionog dizajna

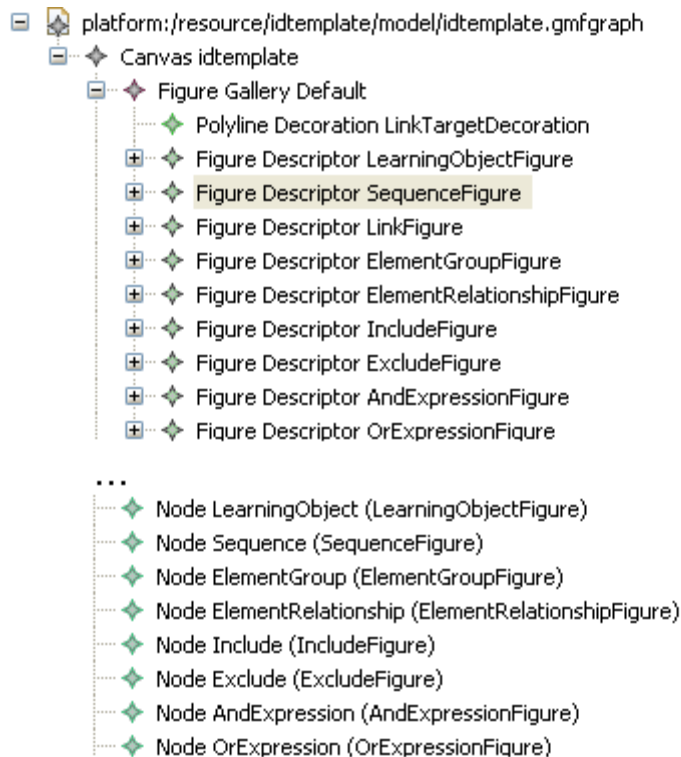
Za većinu elemenata XML šeme ELIDL jezika kreirane su odgovarajuće klase u domenskom modelu. Izuzetak su kontejnerski elementi u XML šemi, čija je uloga da grupišu druge elemente šeme. Tako elementi `uol-structure`, `element-relationships`, `selection-rule` i `conditions` nisu reprezentovani u domenskom modelu. Atributi tagova ELIDL XML šeme, reprezentovani su atributima domenskog modela. Atributima su takođe reprezentovani i element `grading`, kao i njegovi podelementi iz XML šeme ELIDL jezika. Element `join` je u domenskom modelu predstavljen referencom klase `Relationship` (koja odgovara XML elementu `element-relationship`) na klasu `Operator`.

Na slici se vidi da je u domenskom modelu definisan generički element `IDTemplateElement` kojeg nasleđuju svi ostali elementi modela. Elementi koji predstavljaju nastavne elemente u opisu instrukcionog dizajna (`LearningObject`, `Sequence` i `ElementGroup`) grupisani su tako da nasleđuju element `IDAbstractElement`. Operatori su predstavljeni elementima koji nasleđuju element

Operator. Izrazi su predstavljeni naslednicima elementa `Expression`, a akcije naslednicima elementa `Action`.

U skladu sa sadržajem `idtemplate.ecore` fajla, kreira se fajl `idtemplate.genmodel`, na osnovu kojeg se vrši generisanje programskog koda modela.

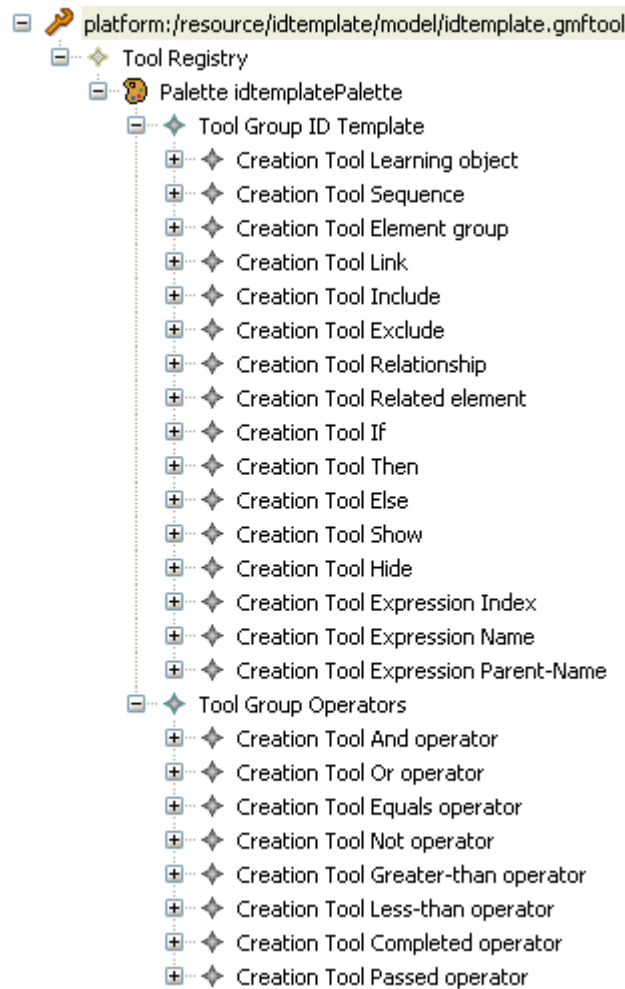
Nakon domenskog modela, koristeći GEF tehnologiju, u fajlu `idtemplate.gmfgraph` definisani su grafički elementi koje editor sadrži, što je prikazano na slici **Error! Reference source not found. 5.**



Slika 5. Definicija grafičkih elemenata u ELIDL editoru

U gornjem delu slike 5 se vidi galerija grafičkih elemenata. U donjem delu slike vide se definisani grafički čvorovi koji se iscrtavaju u editoru i vizuelno reprezentuju elemente ELIDL jezika.

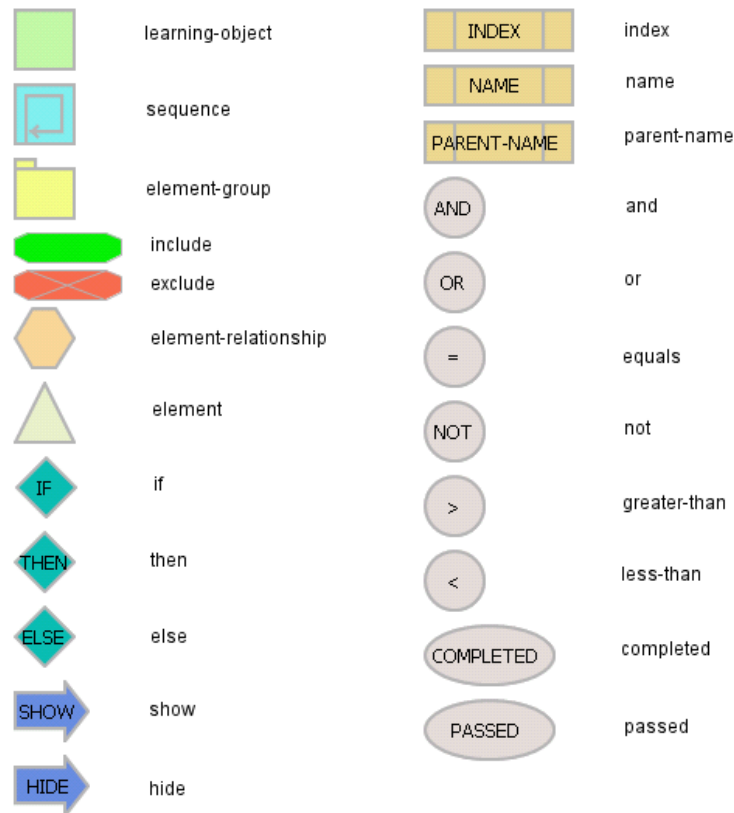
Zatim je definisana paleta elemenata, kao alat za interakciju sa korisnikom. Korisnik bira određeni grafički element sa palete i dodaje ga na dijagram. Paleta je definisana u fajlu `idtemplate.gmftool`, čiji je sadržaj grafički prikazan na slici 6.



Slika 6. Definicija elemenata u paleti ELIDL editora

Na slici 6 se vidi da su elementi palete organizovani u dve grupe. Jednu grupu čine operatori, a u drugu grupu su svrstani svi ostali elementi palete.

Zatim je izvršeno mapiranje između domenskog modela, grafičkih elemenata i elemenata palete. Elementi domenskog modela mapiraju se na grafičke elemente, koji predstavljaju njihovu vizuelnu reprezentaciju u editoru. Na slici **Error! Reference source not found.**7 prikazane su vizuelne reprezentacije elemenata ELIDL jezika u grafičkom editoru.



Slika 7. Vizuelna reprezentacija elemenata u ELIDL editoru

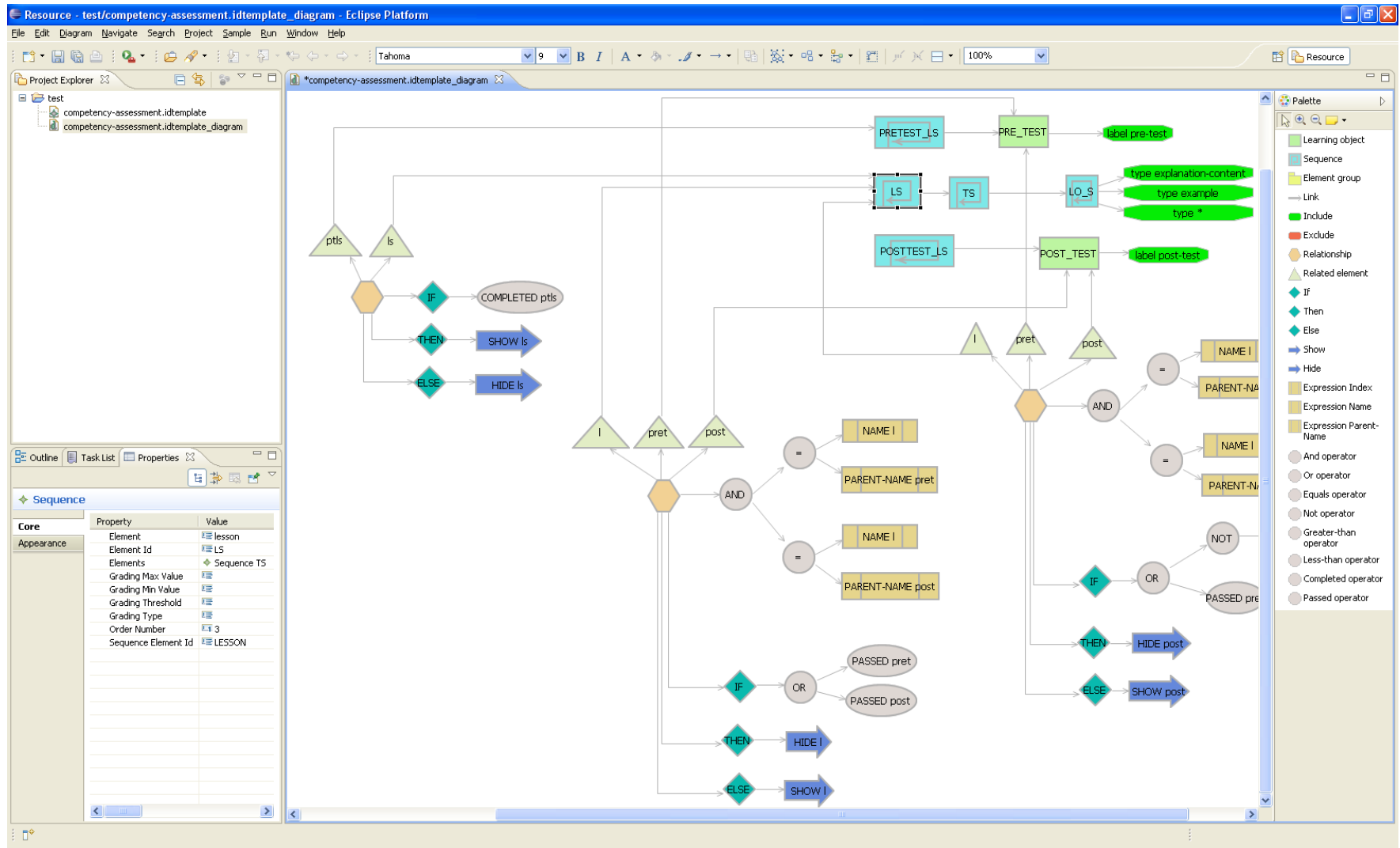
Grafički elementi se u editoru dodaju na dijagram, klikom na određeni element palete. Ovo je definisano mapiranjem grafičkih elemenata na elemente palete. Fajl `idtemplate.gmfmap` definiše sva opisana mapiranja i njegov sadržaj je grafički prikazan na slici 8.



Slika 8. Mapiranje domenskog modela na grafičke komponente ELIDL editora

Ovim su svi potrebni elementi editora definisani. Na osnovu elemenata editora, vrši se generisanje fajla `idtemplate.gmfgen`. Konačno, u skladu sa sadržajem ovog fajla, vrši se generisanje programskog koda ELIDL editora. Izvršavanjem generisanog programskog koda pokreće se grafički editor. U grafičkom editoru je moguće kreirati dijagrame, koji sadrže gore opisane grafičke elemente. Dodatno, izvršena je izmena u generisanom programskom kodu, tako da se dijagram snima u formi ELIDL XML dokumenta u skladu sa XML šemom prikazanom u poglavlju 3.

Izgled prozora grafičkog editora za ELIDL prikazan je na slici 9 **Error! Reference source not found.** Centralni deo prozora zauzima prostor za prikaz dijagrama. Elementi se na dijagram dodaju sa palete koja se nalazi na desnoj strani prikazanog prozora. Dijagram prikazan na slici 9 se odnosi na instrukcionu strategiju pod nazivom „Competency Assessment“ (Chew and Hua, 2008) što je jedna od strategija na kojima je tehničko rešenje verifikovano. Za svaki element dijagrama, mogu se definisati različita svojstva na panelu *Properties*, koji se nalazi u donjem delu prozora. Dijagrami su organizovani u projekte. Hijerarhija projekata i dijagrama prikazuje se na levoj strani prozora. Grafički interfejs editora omogućuje proizvoljnu organizaciju pozicija i dimenzija pomenutih panela. U gornjem delu prozora se nalazi glavni meni koji sadrži standardne funkcionalnosti koje Eclipse platforma podržava (rad sa projektima i drugim fajlovima, pretraga, organizacija prikaza, pomoć itd).



Slika 9. Izgled prozora grafičkog editora za ELIDL

5. Primena tehničkog rešenja

Verifikacija tehničkog rešenja izvršena je definisanjem šest reprezentativnih instrukcionih strategija u okviru softverskog alata koji predstavlja tehničko rešenje. Strategije nad kojima je softver verifikovan su: *No sequencing*, *Linear*, *Knowledge paced*, *Competency assessment* i *Remediation*, sve opisane u (Chew and Hua, 2008) i *Project-based learning*, opisana u (Derntl and Motschnig-Pitrik, 2004). ELIDL dokumenti, kao i prikazi dijagrama za opisane strategije kreirani korišćenjem ovog tehničkog rešenja javno su dostupni na adresi (Savić, 2010b).

Svoju primenu rešenje nalazi u svim slučajevima u kojima je potrebno eksplicitno opisati organizaciju nastave na formalan i mašinski čitljiv način. Formalan opis omogućuje komunikaciju i deljenje znanja između domenskih stručnjaka, kao i jednoznačnu identifikaciju strategije, što omogućuje poređenje različitih strategija. Mašinska-čitljivost instrukcione strategije obezbeđuje mašinsko procesiranje strategije. Mašinsko procesiranje strategije je u (Savić et al., 2012) i (Savić et al., 2011) primenjeno za programsko generisanje nastavnog kursa na bazi instrukcionih strategija formalno definisanih korišćenjem ovog tehničkog rešenja. Korišćenjem ovog tehničkog rešenja najpre je formalno definisano šest gore pomenutih instrukcionih strategija. Definisane strategije su primenjene nad nastavnim materijalom iz kursa Veb programiranje, tako što je programski generisano šest nastavnih kurseva Veb programiranja, pri čemu je svaki od kurseva baziran na različitoj instrukcionoj strategiji. Kursevi su generisani u IMS LD formatu.

U (Savić et al., 2013) je objašnjena primena tehničkog rešenja za merenje efikasnosti formalno opisanih instrukcionih strategija. Tri instrukcione strategije, pod nazivima *Unguided learning*, *Process-worksheets* i *Worked-out examples* su formalno definisane korišćenjem ovde opisanog editora. Formirane su tri grupe studenata od kojih je svaka grupa obavljala nastavne aktivnosti organizovane u skladu sa jednom od tri navedene strategije. Na osnovu rezultata koje su učenici ostvarili na testu, poređene su efikasnosti ovih instrukcionih strategija.

Tehničko rešenje je primenjeno u nastavi na Fakultetu tehničkih nauka na predmetu Numerički algoritmi i numerički softver u inženjerstvu u školskoj godini 2010/2011. Tehničko rešenje je iskorišćeno za formalno definisanje instrukcione strategije u ovom nastavnom kursu. Na osnovu ovako definisane instrukcione strategije i nastavnog materijala u kursu, programski je izvršeno generisanje kursa u IMS LD formatu. Ovaj elektronski kurs studenti su koristili na vežbama u okviru priprema za kolokvijum.

6. Zaključak

Tehničko rešenje je grafički editor za formalno definisanje instrukcione strategije u skladu sa ELIDL jezikom. ELIDL je XML-bazirani meta jezik, koji na generalnom nivou opisuje način obavljanja nastavnog kursa. Tehničko rešenje omogućuje korisniku definisanje ELIDL instrukcionih strategija bez potrebe poznavanja XML tehnologije i sintakse ELIDL jezika. U okviru editora, raspoređivanjem grafičkih elemenata korisnik

kreira dijagram koji se skladišti u formi ELIDL dokumenta. Tehničko rešenje je implementirano u programskom jeziku Java korišćenjem Eclipse GMF razvojnog okvira.

Tehničko rešenje obezbeđuje formalan i mašinski čitljiv opis instrukcione strategije, čime je omogućeno deljenje znanja između domenskih stručnjaka, kao i mašinsko procesiranje instrukcionih strategija. Opis tehničkog rešenja, kao i neke od njegovih konkretnih primena prikazani su u 4 naučna rada, od kojih jedan pripada kategoriji M23, a preostala tri kategoriji M33.

Literatura

- ADL, 2009. Sharable Content Object Reference Model, SCORM 2004 4th Edition [WWW Document]. URL <http://www.adlnet.gov> (accessed 7.5.12).
- AUTC, 2003. Learning Designs [WWW Document]. URL <http://www.learningdesigns.uow.edu.au/> (accessed 4.25.12).
- Botturi, L., 2006. E2ML: A Visual Language for the Design of Instruction. *Educational Technology Research and Development* 54, 265–293.
- Caeiro-Rodríguez, M., 2008. poEML: A Separation of Concerns Proposal to Instructional Design, in: Botturi, L., Stubbs, G. (Eds.), *Handbook of Visual Languages for Instructional Design Theories and Practices*. Information Science Reference, pp. 185–209.
- Chew, L.K., Hua, T.G., 2008. Instructional Strategies and Limitations of the SCORM 2004, in: *Proceedings of the 16th International Conference on Computers in Education (ICCE 2008)*. Taipei, Taiwan, pp. 153 – 160.
- De Moura Filho, C.O., Derycke, A., 2008. Pedagogical Patterns and Learning Design: When Two Worlds Cooperate, in: *Proceedings of the UNFOLD-PROLEARN Joint Workshop*. Valkenburg, The Netherlands.
- Derntl, M., Motschnig, R., 2007. coUML - A Visual Language for Modeling Cooperative Environments, in: Botturi, L., Stubbs, T. (Eds.), *Handbook of Visual Languages for Instructional Design: Theories and Practices*. Information Science Reference, pp. 155–184.
- Derntl, M., Motschnig-Pitrik, R., 2004. Patterns for blended, Person-Centered learning: strategy, concepts, experiences, and evaluation, in: *Proceedings of the 2004 ACM Symposium on Applied Computing, SAC '04*. ACM, New York, NY, USA, pp. 916–923.
- Dick, W., Carey, L., 1990. *The Systematic Design of Instruction*, 3rd ed. Scott, Foresman and Company, Glenview, IL.
- Eclipse Foundation, 2010. Eclipse Helios (3.6) [WWW Document]. URL <http://www.eclipse.org/platform/> (accessed 7.30.12).
- Eclipse Foundation, 2012a. Graphical Modeling Framework [WWW Document]. URL <http://www.eclipse.org/modeling/gmp/> (accessed 4.26.12).
- Eclipse Foundation, 2012b. Eclipse Platform [WWW Document]. URL <http://www.eclipse.org/platform/> (accessed 7.30.12).
- Eclipse Foundation, 2012c. Graphical Modeling Framework - Tutorial [WWW Document]. URL

- http://wiki.eclipse.org/Graphical_Modeling_Framework/Tutorial/Part_1 (accessed 7.30.12).
- Eclipse Foundation, 2012d. Eclipse Modeling Framework Project - EMF [WWW Document]. URL <http://www.eclipse.org/modeling/emf/> (accessed 4.26.12).
- Eclipse Foundation, 2012e. Graphical Editing Framework - GEF [WWW Document]. URL <http://www.eclipse.org/gef/> (accessed 4.26.12).
- Frizell, S., Hubscher, R., 2011. Using Design Patterns to Support E-Learning Design, in: *Instructional Design: Concepts, Methodologies, Tools and Applications*. IGI Global, pp. 114–134.
- Guangzuo, C., Xinqi, R., Haitao, Z., Ronghuai, H., 2009. SMID: A Semantic Model of Instructional Design. *IEEE*, pp. 130–134.
- Hayashi, Y., Bourdeau, J., Mizoguchi, R., 2009. Using Ontological Engineering to Organize Learning/Instructional Theories and Build a Theory-Aware Authoring System. *I. J. Artificial Intelligence in Education* 19, 211–252.
- IMS Global Learning Consortium, 2003a. IMS Simple Sequencing Specification [WWW Document]. URL <http://www.imsglobal.org/simplesequencing/> (accessed 4.25.12).
- IMS Global Learning Consortium, 2003b. IMS Learning Design Specification [WWW Document]. URL <http://www.imsglobal.org/learningdesign/> (accessed 4.25.12).
- Laforcade, P., 2005. Towards a UML-based educational modeling language, in: *Proceedings of the IEEE International Conference on Advanced Learning Technologies ICALT'05*. IEEE, Kaohsiung, Taiwan, pp. 855–859.
- LAMS Foundation, 2005. Learning Activity Management System (LAMS).
- LAMS Foundation, 2012. Activity Planner [WWW Document]. URL <http://wiki.lamsfoundation.org/display/planner/Activity+Planner> (accessed 4.25.12).
- Ljubojevic, D., Laurillard, D., 2010. A theoretical approach to distillation of pedagogical patterns from practice to enable transfer and reuse of good teaching, in: *LAMS and Learning Design*. pp. 70–79.
- Martel, C., Vignollet, L., Ferraris, C., David, J.-P., Lejeune, A., 2006. Modeling Collaborative Learning Activities on e-Learning Platforms, in: *Proceedings of the Sixth IEEE International Conference on Advanced Learning Technologies ICALT'06*. IEEE, pp. 707–709.
- Paquette, G., Léonard, M., Lundgren-Cayrol, K., 2008. The MOT+ Visual Language for Knowledge-Based Instructional Design, in: Botturi, L., Stubbs, G. (Eds.), *Handbook of Visual Languages for Instructional Design Theories and Practices*. Hershey PA Information Science Reference. IGI Global, pp. 133–154.
- Reigeluth, C., Moore, J., 1999. Cognitive Education and the Cognitive Domain, in: Reigeluth, C. (Ed.), *Instructional-Design Theories and Models: A New Paradigm of Instructional Theory*. (2nd Ed). Lawrence Erlbaum Associates, Hillsdale, NJ, pp. 51–68.
- Savić, G., 2010a. ELIDL - E-learning Instructional Design Language - XML Schema.
- Savić, G., 2010b. ELIDL - E-learning Instructional Design Language - Examples.
- Savić, G., Segedinac, M., Konjović, Z., 2011. The Implementation of the IMS LD E-course Generator, in: *Proceedings of the 1st International Conference on*

- Information Society Technology and Management (ICIST 2011). Kopaonik, Srbija.
- Savić, G., Segedinac, M., Konjović, Z., 2012. Automatic Generation of E-Courses Based on Explicit Representation of Instructional Design. *Computer Science and Information Systems* 9, 839 – 869.
- Savić, G., Segedinac, M., Konjović, Z., 2013. Measuring Efficiency of Formally Represented Instructional Strategies, in: *Proceedings of the 3rd International Conference on Information Society Technology and Management (ICIST 2013)*. Kopaonik, Srbija.
- Segedinac, M., Savić, G., Konjović, Z., 2010. Knowledge Representation Framework for Curriculum Development, in: *KEOD 2010 - Proceedings of the International Conference on Knowledge Engineering and Ontology Development*. Presented at the KEOD, Valencia, Spain, pp. 327–330.
- The Pedagogical Patterns Project [WWW Document], 2001. . URL <http://www.pedagogicalpatterns.org/> (accessed 4.25.12).

Рецензија техничког решења

На основу Одлуке Наставно научног већа Факултета техничких наука Универзитета у Новом Саду, од 26.12.2013. године којом је именован за рецензента техничког решења под насловом „Графички едитор за формални опис инструкционе стратегије” и достављене документације о техничком решењу, а у складу са одредбама *Правилника о поступку и начину вредновања, и квантитавном исказивању научноистраживачких резултата истраживача*, који је донео Национални савет за научни и технолошки развој Републике Србије («Службени гласник РС», бр. 38/08) рецензент проф. др Душан Сурла оценио је да су испуњени услови за признање својства техничког решења следећем резултату научноистраживачког рада:

Назив: Графички едитор за формални опис инструкционе стратегије

Аутори: др Горан Савић, Милан Сегединац, дипл. мастер инж., др Зора Коњовић
Категорија техничког решења: (M85) Прототип, нова метода, софтвер, стандардизован или атестиран инструмент, нова генска проба, микроорганизми

Образложење

Предложено решење урађено је за: Факултет техничких наука у Новом саду (За потребе пројекта са ев. бр. ИИИ 47003: „Инфратруктура за технолошки подржано учење у Србији“.

Предложено решење је урађено: 2013. год.

Субјект који је решење прихватио и примењује: Прототип решења се примењује на Катедри за информатику Факултета техничких наука у Новом Саду и на Катедри за методику наставе хемије Природно-математичког факултета у Новом Саду.

Област на коју се техничко решење односи је: Електронски подржана настава.

Проблем који се техничким решењем решава:

Аутоматско генерисање и модификација инструкционог дизајна у електронским курсевима.

Стање решености тог проблема у свету

Анализирајући репрезентацију инструкционог дизајна у постојећим системима за електронски подржано учење, могу се уочити две врсте проблема: 1) опис инструкционог дизајна није довољно струкуриран да би био погодан за машинско процесирање и/или 2) опис инструкционог дизајна није довољно експресиван да опише разнолике инструкционе технике (у већини приступа се користи ограничен, предефинисан скуп техника у оквиру којих наставник може да изабере одговарајућу). Да би се компонента која описује инструкциони дизајн могла машински процесирати, независно од осталих компоненти курса, потребно је експлицитно и формално репрезентовати инструкциони дизајн у машински читљивом формату. Имајући у виду наведене недостатке тренутно заступљених решења, у оквиру техничког решења развијен је нови мета језик за опис инструкционог дизајна у машински читљивом облику и графички едитор за креирање фајлова са описом инструкционог дизајна тако да се отклоне уочени недостаци постојећих система.

Суштина техничког решења.

Техничким решењем се да креаторима курса обезбеђује алат који омогућује једноставно креирање и модификацију инструкцијског дизајна и аутоматско генерисање курса за коришћење у системима за електронски подржано учење на основу дефинисаног инструкцијског дизајна. При томе, су задовољени следећи захтеви:

- генеричност – инструкциона стратегија је независна од садржаја конкретног курса и формата у којем ће тај курс бити репрезентован
- експресивност – омогућује се дефинисање широког скупа различитих инструкционих стратегија
- машинска читљивост – генерисану инструкциону стратегију је могуће машински процесирати

Карактеристике предложеног техничког решења су следеће:

Основу апликације чини нови мета језик ELearning Instructional Design Language (ELIDL) за опис инструкционог дизајна у машински читљивом облику. Техничко решење састоји се од следећих модула:

- 1 Језика ELearning Instructional Design Language (ELIDL)
- 2 Графичког едитора који омогућује дефинисање инструкционог дизајна у ELIDL језику и визуелној нотацији.

ELIDL језик путем XML нотације (XML шема) омогућује дефинисање начина избора и организације наставних садржаја у курсу. За ручно уређивање користи се графички едитор, који омогућује дефинисање инструкционог дизајна у ELIDL језику у визуелној нотацији. Корисник дефинише инструкциони дизајн распоређивањем одговарајућих графичких елемената, а креирани дијаграм се аутоматски снима у ELIDL формату. На овај начин корисник може формално дефинисати инструкциони дизајн без познавања детаља ELIDL синтаксе и XML формата. Едитор је креиран коришћењем Eclipse Graphical Modeling Framework (GMF) технологије.

Могућности примене предложеног техничког решења:

Апликација је намењена за креирање електронских курса на начин који омогућује експлицитну и формалну репрезентацију инструкцијског дизајна у машински читљивом облику. Апликацију могу да користе образовне институције свих нивоа образовања.

На основу свега наведеног као рецензент оцењујем да је резултат научноистраживачког рада под називом: „Графички едитор за формални опис инструкционе стратегије“ успешно реализован, да представља истраживачки резултат који је практично употребљив и предлажем да се прихвати у категорију научно-истраживачких резултата као софтверски производ (M85).

У Новом Саду, 28.12.2013.

Рецензент:

Проф. др Душан Сурла

Природно-математички факултет Нови Сад

Рецензија техничког решења

На основу Одлуке Наставно научног већа Факултета техничких наука Универзитета у Новом Саду, од 26.12.2013. године којом је именована за рецензента техничког решења под насловом „Графички едитор за формални опис инструкционе стратегије” и достављене документације о техничком решењу, а у складу са одредбама *Правилника о поступку и начину вредновања, и квантитавном исказивању научноистраживачких резултата истраживача*, који је донео Национални савет за научни и технолошки развој Републике Србије («Службени гласник РС», бр. 38/08) рецензент проф. др Мирјана Сегединац оценила је да су испуњени услови за признање својства техничког решења следећем резултату научноистраживачког рада:

Назив: Графички едитор за формални опис инструкционе стратегије

Аутори: др Горан Савић, Милан Сегединац, дипл. мастер инж., др Зора Коњовић
Категорија техничког решења: (M85) Прототип, нова метода, софтвер, стандардизован или атестиран инструмент, нова генска проба, микроорганизми

Образложење

Предложено решење урађено је за: Факултет техничких наука у Новом саду (За потребе пројекта са ев. бр. ИИИ 47003: „Инфратруктура за технолошки подржано учење у Србији“.

Предложено решење је урађено: 2013. год.

Субјект који је решење прихватио и примењује: Прототип решења се примењује на Катедри за информатику Факултета техничких наука у Новом Саду и на Катедри за методику наставе хемије Природно-математичког факултета у Новом Саду.

Област на коју се техничко решење односи је: Електронски подржана настава.

Проблем који се техничким решењем решава:

Аутоматско генерисање и модификација инструкционог дизајна у електронским курсевима.

Стање решености тог проблема у свету

Анализирајући репрезентацију инструкционог дизајна у постојећим системима за електронски подржано учење, могу се уочити две врсте проблема: 1) опис инструкционог дизајна није довољно струкуриран да би био погодан за машинско процесирање и/или 2) опис инструкционог дизајна није довољно експресиван да опише разнолике инструкционе технике (у већини приступа се користи ограничен, предефинисан скуп техника у оквиру којих наставник може да изабере одговарајућу). Да би се компонента која описује инструкциони дизајн могла машински процесирати, независно од осталих компоненти курса, потребно је експлицитно и формално репрезентовати инструкциони дизајн у машински читљивом формату. Имајући у виду наведене недостатке тренутно заступљених решења, у оквиру техничког решења развијен је нови мета језик за опис инструкционог дизајна у машински читљивом облику и графички едитор за креирање фајлова са описом инструкционог дизајна тако да се отклоне уочени недостаци постојећих система.

Суштина техничког решења.

Техничким решењем се да креаторима курсева обезбеђује алат који омогућује једноставно креирање и модификацију инструкцијског дизајна и аутоматско генерисање курсева за коришћење у системима за електронски подржано учење на основу дефинисаног инструкцијског дизајна. При томе, су задовољени следећи захтеви:

- генеричност – инструкциона стратегија је независна од садржаја конкретног курса и формата у којем ће тај курс бити репрезентован
- експресивност – омогућује се дефинисање широког скупа различитих инструкционих стратегија
- машинска читљивост – генерисану инструкциону стратегију је могуће машински процесирати

Карактеристике предложеног техничког решења су следеће:

Основу апликације чини нови мета језик ELearning Instructional Design Language (ELIDL) за опис инструкционог дизајна у машински читљивом облику. Техничко решење састоји се од следећих модула:

- 1 Језика ELearning Instructional Design Language (ELIDL)
- 2 Графичког едитора који омогућује дефинисање инструкционог дизајна у ELIDL језику и визуелној нотацији.

ELIDL језик путем XML нотације (XML шема) омогућује дефинисање начина избора и организације наставних садржаја у курсу. За ручно уређивање користи се графички едитор, који омогућује дефинисање инструкционог дизајна у ELIDL језику у визуелној нотацији. Корисник дефинише инструкциони дизајн распоређивањем одговарајућих графичких елемената, а креирани дијаграм се аутоматски снима у ELIDL формату. На овај начин корисник може формално дефинисати инструкциони дизајн без познавања детаља ELIDL синтаксе и XML формата. Едитор је креиран коришћењем Eclipse Graphical Modeling Framework (GMF) технологије.

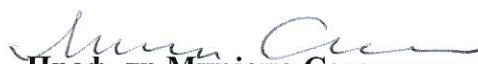
Могућности примене предложеног техничког решења:

Апликација је намењена за креирање електронских курсева на начин који омогућује експлицитну и формалну репрезентацију инструкцијског дизајна у машински читљивом облику. Апликацију могу да користе образовне институције свих нивоа образовања.

На основу свега наведеног као рецензент оцењујем да је резултат научноистраживачког рада под називом: „Графички едитор за формални опис инструкционе стратегије“ успешно реализован, да представља истраживачки резултат који је практично употребљив и предлажем да се прихвати у категорију научно-истраживачких резултата као софтверски производ (M85).

У Новом Саду, 28.12.2013.

Рецензент:


Проф. др Мирјана Сегединац

Природно-математички факултет Нови Сад



УНИВЕРЗИТЕТ
У НОВОМ САДУ



ФАКУЛТЕТ
ТЕХНИЧКИХ НАУКА

Трг Доситеја Обрадовића 6, 21000 Нови Сад, Република Србија
Деканат: 021 6350-413; 021 450-810; Централa: 021 485 2000
Рачуноводство: 021 458-220; Студентска служба: 021 6350-763
Телефакс: 021 458-133; e-mail: ftndean@uns.ac.rs

ИНТЕГРИСАНИ
СИСТЕМ
МЕНАџМЕНТА
СЕРТИФИКОВАН ОД:



Наш број: 01.сл

Ваш број:

Датум: 2013-12-27

ИЗВОД ИЗ ЗАПИСНИКА

Наставно-научног већа Факултета техничких наука у Новом Саду, на 18. редовној седници одржаној дана 26.12.2013. године, донело је следећу одлуку:

-непотребно изостављено-

Тачка 11.1. Верификација нових техничких решења и именовање рецензента

Тачка 11.1.49. у циљу верификације новог техничког решења усвајају се рецензенти:

- Др Душан Сурла, професор емеритус, Природно-математички факултет Нови Сад
- Др Мирјана Сегединац, редовни професор, Природно-математички факултет Нови Сад

Назив техничког решења:

ГРАФИЧКИ ЕДИТОР ЗА ФОРМАЛНИ ОПИС ИНСТРУКЦИОНЕ СТРАТЕГИЈЕ

Аутори техничког решења: др Горан Савић, асистент; мастер инжењер Милан Сегединац, асистент; др Зора Коњовић, редовни професор

-непотребно изостављено-

Записник водила:

Јасмина Димић, дипл. правник

Тачност података оверава:
Секретар

Иван Нешковић, дипл. правник

Декан



Проф. др Раде Дорословачки



УНИВЕРЗИТЕТ
У НОВОМ САДУ



ФАКУЛТЕТ
ТЕХНИЧКИХ НАУКА

Трг Доситеја Обрадовића 6, 21000 Нови Сад, Република Србија
Деканат: 021 6350-413; 021 450-810; Централa: 021 485 2000
Рачуноводство: 021 458-220; Студентска служба: 021 6350-763
Телефакс: 021 458-133; e-mail: ftndean@uns.ac.rs

ИНТЕГРИСАНИ
СИСТЕМ
МЕНАџМЕНТА
СЕРТИФИКОВАН ОД:



Наш број: 01.сл

Ваш број:

Датум: 2014-01-16

ИЗВОД ИЗ ЗАПИСНИКА

Наставно-научног већа Факултета техничких наука у Новом Саду, на 19. седници одржаној дана 15.01.2014. године, донело је следећу одлуку:

-непотребно изостављено-

Тачка 3. Верификација техничких решења

Тачка 3.45: На основу позитивног извештаја рецензената верификује се техничко решења под називом:

ГРАФИЧКИ ЕДИТОР ЗА ФОРМАЛНИ ОПИС ИНСТРУКЦИОНЕ СТРАТЕГИЈЕ

Аутори техничког решења: др Горан Савић, асистент; мастер инжењер Милан Сегединац, асистент; др Зора Коњовић, редовни професор

-непотребно изостављено-

Записник водила:

Јасмина Димић, дипл. правник

Тачност података оверава:
Секретар

Иван Нешковић, дипл. правник

