

**Пловдивски Университет „Паисий Хилендарски“
Факултет по математика и информатика**

Христо Марчов Инджов

Моделиране и управление на динамични проекти за е-обучение

**Автореферат
на дисертационен труд
за присъждане на образователна и научна степен „доктор“**

**по област на висшето образование 4. Природни науки, математика и информатика
в професионално направление 4.6. Информатика и компютърни науки
докторска програма по Информатика**

**Научен ръководител:
проф. дмн Георги Тотков**

**Рецензенти:
проф. д-р Кънчо Иванов
доц. д-р Иван Койчев**

**Пловдив
2014 г.**

Дисертационният труд е обсъден и насочен за защита пред научно жури, на разширено заседание на катедра „Компютърна информатика“ при Факултета по математика и информатика на Пловдивския университет „Паисий Хилендарски“, на 14.02.2014 г.

Дисертационният труд „Моделиране и управление на динамични проекти за е-обучение“ съдържа 153 страници. Библиографията включва 125 източника. Списъкът на авторските публикации се състои от 10 заглавия.

Защитата на дисертационния труд ще се състои на 14.4.2014 г. в Заседателната зала на новата сграда на Пловдивския университет (бул. „България“ №236).

Материалите по защитата са на разположение на интересуващите се в секретариата на Факултета по математика и информатика, нова сграда на ПУ „Паисий Хилендарски“ (бул. „България“ № 236), каб. 330, всеки работен ден от 8.30 до 17.00 часа.

Автор: Христо Марчов Инджов

Заглавие: Моделиране и управление на динамични проекти за е-обучение

Съдържание

| | |
|--|----|
| Съдържание | 3 |
| Използвани съкращения | 4 |
| Въведение..... | 4 |
| Глава 1 Адаптивни системи за е-обучение | 7 |
| 1.1 Системи за е-обучение..... | 8 |
| 1.2 Адаптивно е-обучение | 9 |
| 1.3 е-Обучението като бизнес процес..... | 9 |
| 1.4 Заключение | 11 |
| Глава 2 Проектиране на системи за ДПеО..... | 12 |
| 2.1 Подходи за реализация | 12 |
| 2.2 Принципи за интегриране на СУП в СеО | 13 |
| 2.3 Проект за е-обучение (адаптивен е-курс) | 17 |
| 2.4 Заключение | 19 |
| Глава 3 Програмна реализация на проекта „адаптивен е-курс“ | 19 |
| 3.1 Архитектура..... | 19 |
| 3.2 Функционално разширяване на Moodle | 24 |
| 3.3 Интегриране на редактор на процеси..... | 24 |
| 3.4 Заключение | 25 |
| Глава 4 Реализация на други ДПеО..... | 26 |
| 4.1 Автоматично генериране на тестови въпроси..... | 26 |
| 4.2 Автоматизирано генериране на метаданни | 27 |
| 4.3 Динамично създаване на е-курс..... | 29 |
| 4.4 Заключение | 31 |
| Заключение | 31 |
| Библиография | 33 |
| Публикации по темата на дисертацията | 36 |
| Забелязани цитирания..... | 36 |

Използвани съкращения

| | | |
|------|---|----------------------------------|
| СеО | – | Система за е-обучение |
| УО | – | Учебен обект |
| ЕО | – | Електронно обучение |
| ДПеО | – | Динамични проекти за е-обучение |
| ВиУ | – | Виртуален университет |
| LMS | – | Learning Management System |
| АСеО | – | Адаптивна СеО |
| ВРМ | – | Business Process Modeling |
| СУП | – | Система за управление на процеси |
| ПО | – | Предметна област |
| АТВ | – | Акумулиращи тестови въпроси |
| АОР | – | Акумулираща обвивка за ресурси |
| ПТД | – | Подобrena таксономия на Блум |

Въведение

Процесът на електронно обучение, на пръв поглед не толкова сложен, е свързан понякога с голяма организационна дейност, както за преподавателите, така и за учащите се. Примери за такива дейности са: предоставяне на необходимите материали за електронния курс, създаване на виртуална среда за съвместна работа (провеждане на упражнения, лекции, практики и др.), провеждане на електронен изпит и др.

Примерите по-горе могат да се представят и като конкретни проекти в областта на е-обучението. От проектантска и програмистка гледна точка е естествен въпросът – **възможно ли е създаване на обща схема за моделиране и управление на различни по своето естество проекти за е-обучение.**

Основната цел на дисертационния труд е проектиране и създаване на система за моделиране и управление на динамични проекти в областта на е-обучението.

В изследването се формулира и доказва следната **хипотеза**: възможен общ подход за проектиране и реализация на система за моделиране и управление на динамични проекти в областта на е-обучението е интеграцията на подходящи средства за е-обучение и на система за управление и моделиране на бизнес процеси.

В дисертацията, за понятията „проект” и „управление на проект” се ползват следните две работни дефиниции.

Проектът е специално организирано и ограничено във времето *целенасочено изменение на отделна система*, провеждано в рамките на *планирани ресурси* и установени *изисквания към качеството на резултата*.

Управлението на проект може да се разглежда като:

1. Разработване на *системен план (модел)* за постигане на *целта* на проекта в рамките на наложени ограничения – *планирани ресурси* (времеви, човешки, материални, финансови), *специфични особености* (на организацията на работа, на предметната област), *установени изисквания* към качеството на резултата;
2. *Целенасочено осъществяване на плана* (интерпретация на модела) във фактичката предметна област.

Ще отбележим следните **признаци, характерни за всеки проект**:

- има определена цел;
- притежава (като цяло) определен елемент на новост (спрямо други практики);
- ограничен е във времето;
- разполага с ограничени ресурси (времеви, човешки, материални, финансови);
- определя се като комплексен и различим (от други инициативи);
- изисква специална организация;
- осъществява се под формата на редица изменения на системата (с цел превеждане от едно състояние в друго);

Планът за управление на проект може да се представи с комплекс от документи и схеми (финансови, технически, организационни и др.), представящи системата от разнородни дейности, насочени към достигане на *целта* при наложените ограничения. *Моделът* може да се представи като поток от разнородни дейности, насочени към достигане на *целта* при наложените ограничения.

За постигане на поставената цел на дисертационното изследване бяха планирани следните пет основни задачи:

Задача 1. Изследване на **възможността за постигане на целта** (вкл. за осигуряване на *адаптивност в е-обучението*) на базата на *моделиране и управление на бизнес процеси*;

Задача 2. **Анализ на подходи**, подходящи за ефективна реализация на динамични проекти за е-обучение (ДПеО), базирани на СеО и СУП;

Задача 3. Създаване на **обща схема за разработване** на бързи прототипи на системи за ДПеО.

Задача 4. Програмна реализация на системи за моделиране и управление на ДПеО над популярни СеО и СУП;

Задача 5. Приложения: създаване на бързи прототипи на ДПеО

Структура на дисертацията

Дисертацията е в обем от 153 страници, разделена на Въведение, четири глави, Заключение и Приложение. Литературата се състои от 125 източника, от които 8 са на кирилица, а 30 са интернет-източници.

В **глава 1. Адаптивни системи за е-обучение** е направен обзор на системи за е-обучение (СеО). Дефинирани са техните основни характеристики и са разгледани популярни СеО с отворен код и с комерсиални лицензи. Разгледана е възможността за представяне на учебния процес като бизнес процес. Представени са СеО, използващи системи за моделиране и изпълнение на бизнес процеси за моделиране и изпълнение на учебен процес.

В **глава 2. Проектиране на системи за ДПеО** са разгледани подходи за реализация на ДПеО. Представени са предимства и недостатъци на подходите. Извлечен е общ подход за осъществяване на конкретен динамичен проект чрез интеграция на подходящи средства за е-обучение и на система за управление и моделиране на бизнес процеси. Определени са основни принципи и са наложени ограничения към системите кандидати за интегриране на СУП в СеО. Определени са най-подходящите СеО и СУП (разгледани в глава първа) и на тяхна база е създаден проект за е-обучение „Адаптивен курс в Moodle“

Глава 3. Програмна реализация на проекта „адаптивен курс“ съдържа програмна реализация на проект за адаптивен курс в Moodle. Представени са архитектура на интегрираната система и детайли на нейната реализация. Специално внимание е обърнато на необходимите промени и разширения на базовите системи Moodle и Activiti (интегрираните в случая СеО и СУП).

В **глава 4. Реализация на други ДПеО** са представени допълнителни проекти за е-обучение, основани на т. нар. ‘акумулативни учебни дейности’ [2].

В **Заключението** са систематизирани получените резултати и са посочени научно-приложните и приложните приноси на дисертационния труд. Очертани са насоките за бъдещо развитие на изследването.

В **Приложение** се съдържа ръководство за потребителя на представения в глава 3. проект за е-обучение „адаптивен е-курс“.

Апробация

Резултатите на дисертационното изследване са докладвани на 8 (осем) научни конференции (5 международни и 3 национални). Част от резултатите на дисертационното изследване са докладвани на семинари по електронно обучение в Пловдивския университет „Паисий Хилендарски“ и университет ELTE, Будапеща, Унгария. Пълният списък с публикациите е даден в раздел **Публикации по темата на дисертацията**. Публикацията П10.*Supporting Adaptive E-Learning with an Open-source Learning Management System* е отличена с награда за най-добра статия на международната конференция CompSysTech'13, а 3 (три) са цитирани в 9 (девет) публикации (4 на чуждестранни автори и 5 от български). Резултати на дисертационния труд са използвани в 6 (шест) проекта (3 с европейско финансиране, 2 – с национално и 1 – с университетско):

- Моделиране на учебни процеси и управление на проекти за е-обучение, ВУ-МИ-203 (Фонд „Научни изследвания“ към МОН).
- Подкрепа на творческото развитие на докторанти, пост-докторанти и млади учени в областта на компютърните науки, BG051PO001-3.3.04/13 (ОП „Развитие на човешките ресурси“ на Европейския Социален Фонд)
- Автоматизирано генериране на метаданни за спецификации и стандарти на е-документи, ДО 02-308 (Фонд „Научни изследвания“ към МОН).
- „Университетска информационна система за осигуряване и поддържане на качеството в науката” УАСОПН11 ФМИ 002/30.05.2011 (Фонд „Научни изследвания“ към ПУ „Паисий Хилендарски“).
- Пловдивски електронен университет: национален еталон за провеждане на качествено електронно обучение, BG051PO001-4.3.04-0064 (ОП „Развитие на човешките ресурси“ на Европейския Социален Фонд).
- Стандартизация и интегриране на разнотипни университетски информационни ресурси и системи, BG051PO001-3.1.08-0041 (ОП „Развитие на човешките ресурси“ на Европейския Социален Фонд).

Глава 1 Адаптивни системи за е-обучение

В настоящата глава от дисертационния труд е направен обзор на системи за е-обучение (СеО). Дефинирани са техните основни характеристики и са разгледани популярни СеО с отворен код и с комерсиални лицензи. Дефинирани са понятията адаптивност и адаптируемост. Представени са основните характеристики на адаптивни СеО и са разгледани различни подходи за постигане на адаптивност. Разгледана е възможността за представяне на учебния процес като бизнес процес. Представени са стандарти

за моделиране и изпълнение на бизнес процеси, системи реализиращи тези стандарти и CeO използващи системи за моделиране и изпълнение на бизнес процеси за моделиране и изпълнение на учебен процес. Разгледано е значението на стандартите в е-обучението и метаданните, прикрепени към учебни обекти за реализацията на адаптивни проекти в е-обучението.

1.1 Системи за е-обучение

Все повече CeO използват максимално ефективно възможностите, които предлага WWW така, че качеството на виртуалното обучение да бъде все по-добро. Тези системи са познати още като системи за управление на курсове (Course Management Systems), Системи за управление на обучението (Learning Management System) (LMS) и виртуални среди за обучение (Virtual Learning Environments). Това са системи за многомодулно електронно представяне на лекционни материали за студентите. Съществуват множество такива софтуерни системи, които предоставят определен набор от функции.

Електронното обучение е решение на предизвикателството, свързано с образованието и комуникациите, което отправя глобалната мрежа. Тази нова форма на обучение представлява предаване на знания и умения, което днес се подпомага или изцяло се провежда чрез Интернет. Първоначално е-обучението започва да се използва във фирмени отдели за повишаване на квалификацията на персонала, в училищата и в университетите като допълнение към стандартните методи за обучение, а днес включва широк набор от интерактивни възможности за обучение, обмен на информация, опит и идеи.

Възможни са различни класификации на CeO. Според една от тях [6], има три основни типа CeO:

- *самостоятелни виртуални образователни среди* (уеб-страници с учебно съдържание, в които студентите търсят материали и се подготвят за изпити);
- *интелигентни мултимедийни виртуални среди* (използват виртуални светове, с които студентите активно взаимодействат и така прилагат своите познания по дадени предмет; напр. – виртуални лаборатории, организирани на принципа на интелигентното подсказване и насочване на студентите чрез кратки съобщения и прекъсване на работата им, когато работят в грешна насока);
- *разпределени среди* (включват разпределени приложения за посещения от множество студенти, организирани в групи - виртуални класни стаи).

Разгледани са най-използваните CeO в академична и смесена среда (академична и бизнес). За целта е използвано изследване на Capterra [13], според което това са (в низходящ ред на популярност): Moodle, Edmodo, Blackboard, Desire2Learn, Schoology, Sakai

и др. Проучвания за 2008 и 2009 година, разработени от *eLearning Guild Academy* [19], класира Moodle като най-използваната LMS за тези години.

1.2 Адаптивно е-обучение

Адаптацията в контекста на ЕО придоби голяма популярност през последните години, но доста от проблемите, свързани с прилагането ѝ при системи за ЕО (СЕО) все още са нерешени [36]. Прилагането на адаптивност в съвременните форми на електронно обучение подпомага обогатяването на комуникацията между обучаваните и системата и спомага за постигането на основната цел на електронното обучение – повишаване на ефективността и ефикасността на обучението. Следвайки това направление се създава ново поколение системи, наречени АСеО (системи от четвърто поколение) [12, 11].

Компютърните системи (в това число и СеО), способни на адаптация са два вида: адаптивни (от английски *adaptive*) и адаптируеми (от английски *adaptable*) [36]. Въпреки общия корен на двете прилагателни, между адаптивни и адаптируеми системи съществуват фундаментални разлики. Адаптируеми са системите, даващи възможност на потребителя да променя определени системни параметри като по този начин персонализира поведението им, а системи, които автоматично се приспособяват към нуждите на потребителя се наричат адаптивни.

1.3 е-Обучението като бизнес процес

Удовлетворяването на подобни изисквания за адаптивност и приспособимост на виртуалния учебен процес (едновременно със спазване на определени методически и ресурсни ограничения), обаче, поставя на дневен ред редица проблеми, свързани със създаването на адекватни средства за неговото моделиране, изпълнение в реално време и управление (вкл. адаптивни потребителски интерфейси, средства за автоматизация и др.). В последните години за решаване на посочените проблеми се привличат технологии и средства за моделиране и управление на процеси. Дисертационното изследване е насочено към прилагане на системи за моделиране и управление на бизнес процеси в областта на проектирането и създаването на адаптивни СеО за реализация на динамични проекти за е-обучение (ДПеО). Стремелът е към обогатяване на възможностите на съществуващите СеО чрез въвеждане на допълнителни функционалности, предоставящи по-ефективни методи в обучението.

Съвременните системи за електронно обучение (СеО) са фокусирани върху осигуряване на отделни виртуални учебни дейности, оставяйки на заден план процеса на

обучение (в рамките на който се провеждат) и свързаните с тях изисквания и ограничения от логически и методически характер. Иначе казано, съвременните СеО не са подходящи за моделиране на разнообразните форми на обучение, наблюдавани в практиката. Нещо повече, дори опитите за представяне на адаптивни виртуални учебни курсове в подобни системи среща определени трудности [50, 26].

Системи за управление на бизнес процеси

Подобно на системите за е-обучение съществува богат избор и от системи за управление на процеси. Следва описание на три от най-популярните системи с отворен код: *jBPM*, *Activiti*, *Bonita Open Solution*.

Activiti BPM Platform

Activiti BPM Platform [7] е платформа за потоци с общо предназначение и за управление на бизнес процеси. Тя е с отворен код и насочена към бизнес средите, разработчиците и системните администратори. Поддържа напълно BPMN 2.0 (Business Process Modeling and Notation) [10], може да се използва във всяко Java приложение, разработена на базата на Spring Framework [47] и има ниски системни изисквания.

Основни конкуренти на Activiti сред системите с отворен код са *JBoss BPM (jBPM)* и *Bonita Open Solution*. Други проекти с отворен код за управление на работни потоци, са *WfMOpen* [54], *XFlow* [55], *Runa WFE* [41] и др., но те не поддържат изпълнение на процесни модели.

Моделиране и изпълнение на учебни процеси

Средства за моделиране и изпълнение на процеси се използват за изграждане на СеО с различен успех. Така например, в [29, 31] се анализира необходимостта от създаване на средства за по-гъвкаво описание на учебните курсове и на начините за протичане на учебния процес. Като възможно решение е предложено разширение на авторската система за електронно обучение *Flex-eL* с интегриране на *FlowMake* (средство за моделиране управление на работни потоци).

В [31] авторите се фокусират върху конкретен модел на процеси, който прилагат при описание на университетски учебни курсове. Въведеният [42] графичен език за описание на процеси отговаря по спецификация на стандарта на коалиция *WfMC* [21], като използва само основният набор от шаблони, въведен там, а именно: *Sequence*, *XOR-Split*, *XOR-Join*, *AND-Split* и *AND-Join*.

По-късно в [8] е предложен нов подход за проектиране на системи за е-обучение, базирани на [22], описвайки е-обучението като бизнес-процес, и използвайки инструменти за моделиране на бизнес-процеси. В разработката се използват концепции за моделиране на бизнес-процеси, заимствани от UML и Rational Unified Process [25, 49], като се предлага методика за описание и анализ на системите за е-обучение, която да подпомогне тяхното следващо разработване.

Друго цялостно решение - от моделиране до изпълнение в реално време, е представено в [15]. Прилагат се корпоративни инструменти, сред които и Microsoft BizTalk като средство за виртуално изпълнение на бизнес-процеси. За описание на процесите се използва авторска графична среда, която свежда съответните модели до представяне на езика XLANG (използван в BizTalk).

В [28] се предлага - задачата за следене и контрол на изпълнението на виртуалните учебни дейности, да се решава с използване на обектно-ориентиран метод за моделиране на учебните потоци и изграждане на архитектура на три нива. Практическа реализация на приложение на базата на предложените метод и архитектура не е описана.

В [37, 38] се предлага интересна стратегия за колаборативно обучение базирано на процеси. Използва се средата Moodle и jBPM [24] съответно за система за е-обучение и система за управление на процеси, като по този начин се постига нелинейност в учебния процес. Недостатък в разработката обаче е фактът, че нелинейността достига едва до ниво на секция (група от учебни обекти/дейности) в Moodle. Иначе казано не могат да бъдат създавани процеси посредством персонални учебни обекти/дейности.

1.4 Заключение

В текущата глава е изследвана **възможността за постигане на целта** (вкл. за осигуряване на *адаптивност в е-обучението*) на базата на *моделиране и управление на бизнес процеси*. Адаптивността в СеО е необходима за гарантиране на гъвкаво съдържание на е-курсовете като се приспособяват учебните обекти – материали и *е-дейности* към профила на всеки обучаван. Чрез подбор на подходящи пътища за учене се оптимизира учебното съдържание, повишава се мотивацията на обучаваните и се увеличава ефективността при усвояване на знанията. Възможността за представяне на учебния процес като бизнес процес е потвърдена.

Глава 2 Проектиране на системи за ДПеО

В тази глава са разгледани подходи за реализация на ДПеО. Представени са предимства и недостатъци на подходите. Извлечен е общ подход за осъществяване на конкретен динамичен проект чрез интеграция на подходящи средства за е-обучение и на система за управление и моделиране на бизнес процеси. Определени са основни принципи и са наложени ограничения към системите кандидати за интегриране на СУП в СеО. Определени са най-подходящите СеО и СУП (разгледани в глава първа) и на тяхна база е създаден проект за е-обучение „Адаптивен курс в Moodle“

2.1 Подходи за реализация

За да се постигне идеята за работа на система за електронно обучение, в която учебният процес да бъде описван с идеите на бизнес процесите и съпътстващата ги технология на поточните системи, са възможни следните подходи [1]:

- Подход 1: Създаване на нова специализирана СеО, оборудвана с интегрирани инструменти за моделиране и управление на потоци;
- Подход 2: Създаване на нова СеО, като се използват вече създадени инструменти за моделиране на бизнес процеси, т.е. на базата на популярна СУП;
- Подход 3: Разширяване на функционалността на популярна СеО със средства за моделиране и управление на учебни процеси ;
- Подход 4: Интеграция на популярни СеО и СУП.

При първия подход се започва изграждането „от нула“, което е твърде трудоемко. Освен това се засяга проблемът с „поредната система“ (твърде много системи за едно и също нещо, което затруднява потребителите, а и не предлага различна функционалност). Учебните заведения вече са свикнали с работата с конкретна СеО, чиято евентуална смяна с новия продукт изисква време, за да се свикне с нея и потребителите да я разучат. Поради тези причини се отхвърля като възможен за разработката. Той би бил подходящ в среда, в която не се използва активно СеО.

Подходи 2 и 3 много си приличат по това, че опитват да разширят съществуваща система с нова функционалност. Тук обаче се появява не само проблемът с трудоемкостта, но и този с правата на собственост на базовите системи и авторските им права. Редица известни и широко използвани в практиката СеО и СУП са затворени за неотворени решения.

Подход 4 предлага използването на вече утвърдена в даденото учебно заведение СеО и интегрирането ѝ със система за управление на процеси. Този подход има няколко важни предимства пред до момента изброените:

- преодолява синдрома ‘още една СеО’ за проектантите и потребителите (интерфейсът и функционалността на избраната популярна СеО се запазват);
- позволява реализация на ‘бърз прототип’ на СеО с управление на процесите (не е необходимо да се проектира и реализира ‘от нула’ – използват се възможностите на избраната популярна СеО като специфичните компоненти на е-обучението се моделират със средства предоставени от СУП);
- разширява функционалността на конкретната СеО в посока на използване на технологии и средства за моделиране и управление на виртуални учебни процеси;
- с развитието на избраната СеО (напр. с включване на нови виртуални учебни дейности и събития) автоматично се обогатява функционалността на системата като цяло (в модела на процеса може да се включват и успешно управляват и новите дейности);
- възможно е мултиплициране на резултатите (избор на друга СеО като основа), и др.

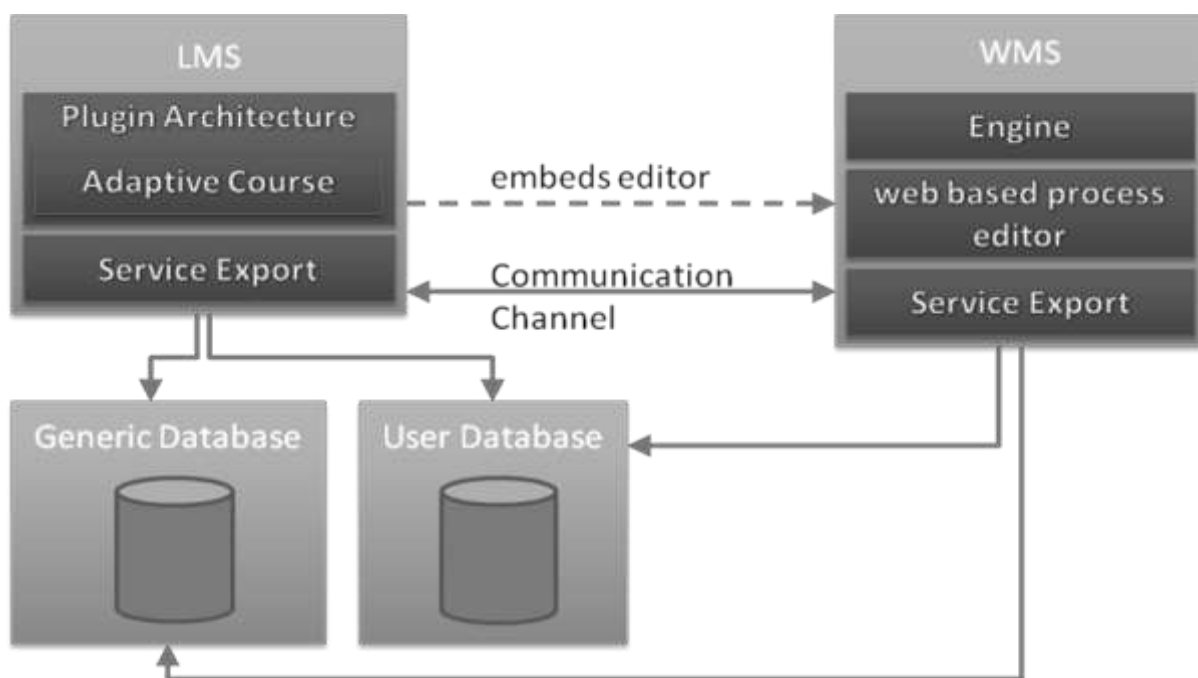
На база на изброените преимущества подход 4. предлага бърз и ефективен начин за реализация на динамични проекти в е-обучението с помощта на СеО и СУП. Ще отбележим, че същият подход успешно е използван в рамките на дисертационния труд при създаването на четири проекта за е-обучение: адаптивен е-курс в Moodle (Раздел 2.3. и Глава 3.), автоматично генериране на тестови въпроси (Раздел 4.1.), автоматизирано генериране на метаданни (Раздел 4.2.) и динамично създаване е-курс (Раздел 4.3).

2.2 Принципи за интегриране на СУП в СеО

На базата на създадените в дисертационния труд проекти за е-обучение са изведени основни принципи при интеграцията на СУП в СеО. В основата е общ подход, същността на който се заключава в изследване на възможността за интегриране в СеО с отворен код (широко разпространена) на елементи на СУП (с широко предназначение). Съществено е да се отбележи какво се разбира тук под „СУП с широко предназначение“. В случая става въпрос за СУП, която притежава средства за моделиране и управление не само на бизнес-процеси, но и на компонентите на самата потокова структура (дейности, подпроцеси, ресурси, ограничителни условия и др.). Иначе казано, в подобни системи могат предварително да се моделират елементите на бизнес-средата (в случая – е-обучението), които се използват при дефиниране, изпълнение и управление на

съответните процеси, след което системата започва да действа като специализирана (за целите на е-обучението) СУП.

На Фиг. 1 е представена обща схема за реализация на Подход 4. Схемата съдържа четири основни компонента: СеО, СУП, База данни с общо предназначение и База данни от потребители. СеО включва в себе си адаптивен електронен курс, реализиран на базата на **рамка за разработка на разширения**. Моделирането на курсове се извършва с помощта на **редактор на процеси, предоставен от СУП и вграден в СеО**. Двете системи предоставят външен достъп до ресурси чрез услуги като по този начин се реализира **двупосочен канал за комуникация**, необходим за обмяна на метаданни при разклоненията на нелинеен учебен процес. Последните два компонента (База данни с общо предназначение и База данни от потребители) често са един и същи компонент. По-добрият подход при комплексни системи, обаче, е реализацията им като отделни компоненти.



Фигура 1. Обща схема за интегриране на СУП в СеО

Следва пълен списък от изисквания наложени от общата схема (фиг. 1):

- СеО трябва да е уеб базирана.
- СУП трябва да разполага с уеб базиран редактор на процеси (основната идея тук е редакторът на процеси да бъде вграден в потребителския интерфейс на СеО, за да се избегне нуждата от преминаване към друга система (в общия случай десктоп базирана) за създаване на модел на процес)

- И CeO, и СУП трябва да могат да използват една и съща база данни за потребители. По възможност, да могат да консумират бази данни за потребители като LDAP [27].
- И CeO, и СУП следва да разполагат с оперативно съвместими данни за износ на услуги. Ако е възможно, избраните системи трябва да включват „готови за употреба“ сървър и / или клиент за SOAP [43], REST [40], и т.н.
- CeO трябва да разполага с рамка за създаване на разширения. По този начин по естествен път функционалността на СУП става част от CeO.
- Редакторът на процеси трябва да позволява дефиниране на домейн специфични елементи на процес. Идеята тук е учебните дейности, налични в CeO, да се моделират посредством езика за описание на процеси, използван от СУП.
- Още едно изискване, основно и за двете системи, е те да бъдат с отворен код, за да могат да бъдат свободно разширявани и променяни.

На базата на поставените изисквания е направена сравнителна характеристика на популярни CeO (Таблица 1.).

| CeO | Moodle | Edmodo | BVLearn | Desire2Learn | Schoology | Sakai | LAMS |
|----------------------------------|--------|--------|---------|--------------|-----------|-------|------|
| Изисквания | | | | | | | |
| Уеб базирана | + | + | + | + | + | + | + |
| LDAP поддръжка | + | - | + | + | - | + | - |
| Уеб услуги (REST, SOAP) | + | + | + | + | + | + | - |
| Клиент за уеб услуги | + | - | - | - | - | + | - |
| Рамка за създаване на разширения | + | + | + | + | - | + | - |
| Отворен код | + | - | - | - | - | + | - |

Таблица 1. Сравнение на CeO.

Основните кандидати са Moodle и Sakai. Те изпълняват всички изисквания. Тъй като говорим за популярни системи, обаче, Moodle е еднозначен победител в тази категория [13, 19, 9, 20].

Сравнителна характеристика е направена и при СУП (Таблица 2.)

| СУП \ Изисквания | Activiti | Bonita | jBPM | U.NET |
|----------------------------------|----------|--------|------|-------|
| Уеб базиран редактор | + | - | + | - |
| LDAP поддръжка | + | + | + | - |
| Уеб услуги (REST, SOAP) | + | + | + | + |
| Клиент за уеб услуги | + | + | + | - |
| Рамка за създаване на разширения | - | - | - | - |
| Домейн специфични елементи | + | - | + | + |
| Отворен код | + | - | + | - |

Таблица 2. Сравнение на СУП

Тук също има два основни кандидата: jBPM и Activiti. И при двете системи липсва ясно очертана рамка за разработка на разширения. Activiti обаче превъзхожда jBPM по няколко допълнителни критерия: необходими хардуерни ресурси при изпълнение на процеси и бързодействие.

Типичен обект за предметната област „електронно обучение“ е електронният курс, а типичен проект – „създаването и провеждането на електронен курс“. В този случай на **първо място** управлението на проекта включва **разработване на план** за постигане на целта на проекта, а именно – създаване и провеждане на е-курс, в рамките на съществуващи ограничения, в съответната образователна институция (софтуерни средства за провеждане на е-обучение, стандарти и организация на учебния процес и др.). Разработването на конкретния план означава създаване на:

а) (*статичен*) модел на е-курс, съставен от обекти (видове учебни дейности и ресурси и др.), субекти (обучавани, автори, преподаватели, консултанти, администратори и др.) и структури (учебна програма, учебен график и др.);

б) (*виртуален*) модел на е-курс, представен от описания (формални или не) на процесите на взаимодействие между виртуализирани и реални обекти, субекти и структури на неговия статичен модел.

На този етап моделите са формални и не включват конкретни виртуални или реални обекти, субекти и структури.

На *второ място* следва целенасочено изпълнение на разработения план, което означава интерпретиране на модели а) и б), вкл. реално и/или виртуално осъществяване на моделите в предметна област „е-обучение“. В случай а) това означава „изпълване“ на модела със съдържание, или асоцииране с модела на е-курс на конкретни виртуални и/или реални обекти, субекти и схеми. Интерпретацията на модел б) включва *следване на плана за изпълнение на електронния курс* (вкл. стартиране на обучението с динамично съпровождане и поддържане на всички потребители до приключване на проекта).

Ще отбележим, че в CeO от типа на Moodle не може да се създават абстрактни модели от типа на а) или б). В литературата са предложени формални модели на е-курс и средства за автоматизирано създаване и редактиране на модели на е-курсове от тип а) и б) [51, 4, 52, 5, 18, 3, 17, 46].

Проектът на е-курс, реализиран в следващия раздел следва представената по-горе схема по следния начин:

- модел а) се осъществява със средства на система *Acticiti* (за моделиране на бизнес процеси) и на Moodle (за моделиране на типични учебни дейности, ресурси и потребители);
- модел б) и интерпретацията на модели а) и б) не могат да се променят – следват логиката на вградените интерпретатори на Moodle (за интерпретация на виртуалните учебни дейности, ресурси и потребители) и на *Activiti* (за интерпретация на елементите на моделираните бизнес процеси).

2.3 Проект за е-обучение (адаптивен е-курс)

На базата на общата схема дефинирана по-горе (фиг. 1) и избраните CeO (Moodle) и СУП (*Activiti*) е създаден проект за е-обучение адаптивен курс в Moodle. В раздела се определят функционалните и нефункционалните изисквания към проекта.

Функционални изисквания

От интерес са дейности като създаване на модел на курс, преминаване към следваща стъпка в модела и други, специфични за адаптивния курс. Основните функционални изисквания са групирани както следва:

- Управление на адаптивен курс – създаване, редактиране, изтриване.
- Управление на учебни материали/дейности към адаптивен курс – създаване, редактиране, изтриване.
- Управление на нелинеен модел на адаптивен курс – създаване, редактиране, активиране (въвеждане в експлоатация) в това число и моделиране на специфични Moodle дейности.
- Създаване на инстанция на модела за даден курс за всеки студент
- Проследяване на напредък и статус на студент в даден курс чрез неговата инстанция на модела.
- Преминаване към следваща стъпка/задача в модела.
- Справка за изминали стъпки/задачи за всеки студент.

Въпреки че и в Moodle (Студент, Преподавател, Администратор, Мениджър, Автор на курс, Гост и др.) и Activiti (Администратор, Мениджър, Потребител) съществуват множество роли, към момента разширението адаптивен курс разграничава две роли: преподавател/администратор и студент. Въпреки, че тези роли комбинират възможности и от двете системи, е необходимо ролите да са съвместими с тези, вече предоставени от Moodle.

Нефункционални изисквания

Следните нефункционални изисквания се дефинират към реализацията на проекта за е-обучение адаптивен курс:

- Разширяемост – да е лесен за адаптиране и да позволява бъдещо разширяване;
- Скалируемост – възможност за едновременно ефективно обслужване на голям брой конкурентни потребители;
- Преносимост – трябва да бъде лесно преносим на различни софтуерни платформи;
- Надеждност – трябва да предоставя надежден достъп до съдържанието;
- Лесна поддръжка и модификация на компонента: чрез разделяне на бизнес логиката от съдържанието и, от друга страна, на разделяне на съдържанието от представянето му;
- Минимизиране на цената на разработка чрез използване на безплатни и базирани на отворен код софтуерни технологии и среди за разработка.

2.4 Заключение

Изследваните в главата подходи са подходящи за ефективна реализация на динамични проекти за е-обучение (ДПеО), базирани на СеО и СУП. На базата на проведен анализ е предложен общ подход и схема за реализация на проекти в областта на е-обучението. Разработена е архитектура за проектиране и реализация на системи за **адаптивно е-обучение** като надстройка на съществуващи СеО и СУП.

Глава 3 Програмна реализация на проекта „адаптивен е-курс“

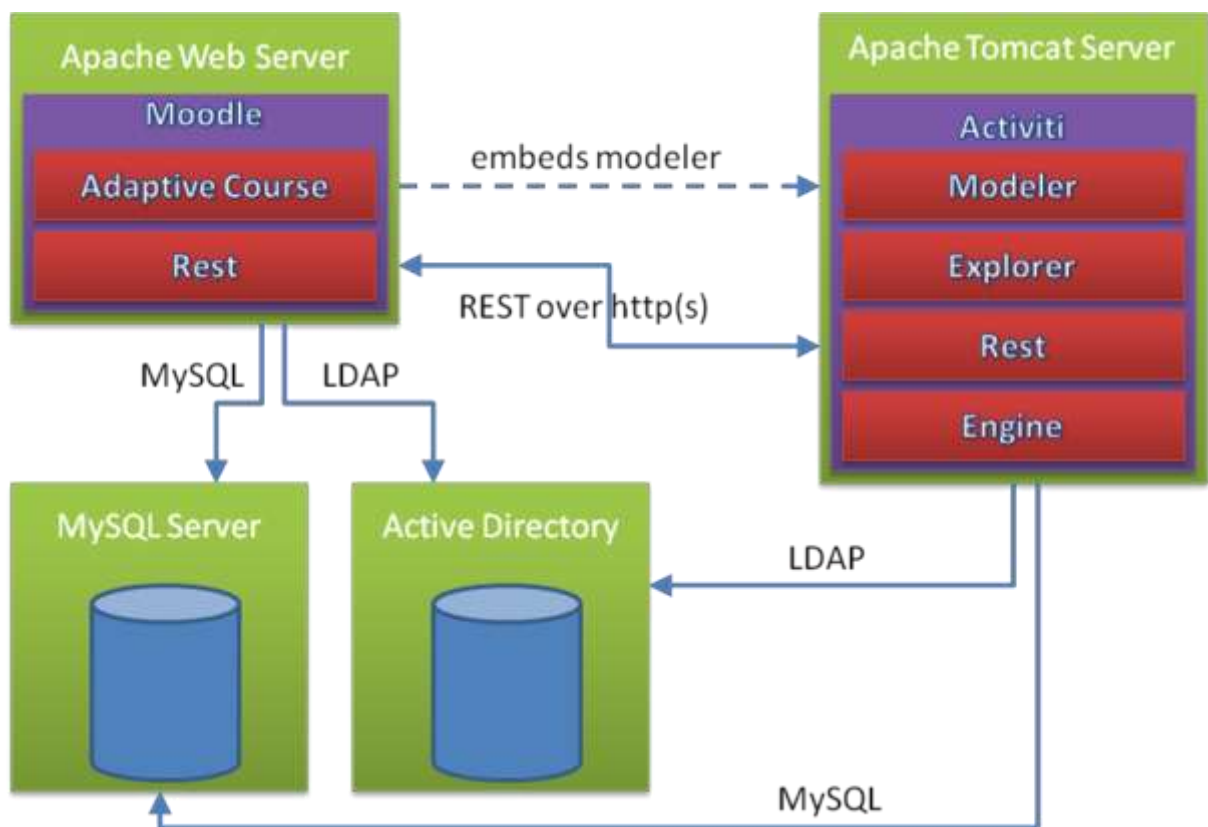
ДПеО, „адаптивен е-курс“, предлага инструменти за моделиране и управление на конкретни електронни курсове. Моделирането се извършва чрез комплексни (нелинейни) поточни структури, базирани на учебни дейности (дефинирани в Moodle). Activiti управлява тези процеси на електронно обучение за множество от студенти, които взимат участие в даден курс.

3.1 Архитектура

Фиг. 2 представя една от възможните конкретни реализации на генералния подход, представен в раздел 2.2. За всяка от необходимите абстрактни системи е поставена съответната конкретна такава. Следва подробно описание на начина, по който системите са организирани в интегрирана платформа за адаптивно е-обучение.

Интегрираната система се състои общо от три модула: Apache Http Server и Moodle образуват първия, Apache Tomcat и Activiti втория, а базата данни (MySQL в случая) третия. Възможно е всички системи да работят съвместно на една физическа машина, както и да са разположени на три отделни машини с мрежова свързаност. Основният начин за комуникация между Activiti и Moodle е канал, базиран на REST услуги. Каналът е двупосочен. С други думи и Activiti и Moodle едновременно предоставят и консумират услуги. Допълнително Moodle вгражда графичния уеб редактор на Activiti. И двете системи комуникират самостоятелно с базата данни, като е важно да се отбележи, че промяна по структурата на базата данни не е била необходима нито в Moodle, нито в Activiti.

За добавянето на адаптивен курс, който се моделира с помощта на BPMN 2.0 и изпълнява посредством engine за изпълнение на процеси, е разработен Plugin за Moodle. Въпросният plugin интегрира в себе си Rest клиент (част от Zend Framework), графичен редактор на процеси и логиката за управление на адаптивния курс.



Фигура 2. Конкретна реализация на общия подход представен в раздел 2.2

За разработката на прототипа на системата за електронно обучение са използвани множество разнородни технологии. Описаната в дисертацията интеграция на разнородни системи, предполага съвместна работа на няколко различни езици за програмиране/технологии. При текущия избор на системи (Moodle, Activiti) технологиите/езиците за разработка могат да се категоризират като Клиентски/Сървърни/Комуникационни и според системата, в която се използват.

По първия начин получаваме групирането:

- Сървърни: Java, PHP, SQL(База данни - една или повече),
- Клиентски: JavaScript, HTML,
- Комуникационни: REST, JSON.

Следва подробно описание на използваните технологии според системата, в която се използват.

Moodle

Системата Moodle (версия) е реализирана изцяло на комбинация от PHP, HTML и JavaScript. PHP е скриптов език за програмиране с общо предназначение за разработка

на уеб приложения, като може да бъде вграден и в HTML (Hyper Text Markup Language). Наложил се е като сървърен език за асемблиране на HTML страници, но в текущата си форма се характеризира като език с общо предназначение. PHP интерпретаторът е компилиран за множество операционни системи (при необходимост разработчик може да компилира PHP за непопулярна операционна система стига за нея да съществува C/C++ компилатор). PHP се поддържа и от множество уеб сървъри измежду който Apache (Apache), IIS [23], Nginx [35] и др. В допълнение PHP 4 и PHP 5 са с отворен код и свободни за употреба. С други думи използването на PHP осигурява едновременно достъпност и преносимост.

Java Script е динамичен скриптов език, поддържащ прототипно базирани обектни конструкции. Той е най-популярният програмен език по света. Той е език за уеб приложения, за сървъри, за персонални компютри, таблети, смартфони и още много други. Java Script може да бъде вграден в HTML страници и да бъде изпълняван от всички модерни уеб браузъри.

В Moodle освен функционалността, наследена от PHP, са включени и множество софтуерни рамки, добавящи допълнителни функции към ядрото на Moodle. Една от тези добавки, REST клиент, стои в основата на интеграцията и е част от Zend Framework [56]. Наличието на REST клиент улеснява драстично интеграцията, тъй като позволява консумирането на REST услуги без нужда от добавяне на допълнителен софтуер. Zend Framework е обектно ориентирана платформа, разработена на PHP и съдържаща в себе си разнородни компоненти за достъп до бази данни и различни услуги на Google, Microsoft и др. Лицензът на Zend Framework е New BSD Licence, позволяващ свободно разпространение и модифициране.

В основата на интеграцията стои възможността за създаване на услуги за достъп до данните на Moodle. Създаването на различни по тип web услуги е документиран процес при разработка на Moodle разширения.

Технологиите, използвани при разработката на Moodle и преизползвани за създаването на интегрираната система са с отворен код, преносими на различни платформи и достъпни за широка публика. Допълнително преимущество е наличието на документация за потребителя, описание на програмния интерфейс и потребителска общност за всяка от технологиите и Moodle.

Activiti

За реализацията на системата Activiti (версия) е използвана комбинация от Java, JavaScript, HTML и REST услуги. Activiti е силно модулна система и в различните модули са застъпени в различна степен горе-описаните технологии.

Лицензът, под който се разпространява Activiti, е Apache License 2.0. Той позволява свободна модификация и разпространение на продукти, базирани на Activiti.

И Activiti и Moodle осигуряват възможност за обмен на данни с други системи чрез REST услуги. Данните се предават в текстов вид в JSON формат. JSON е олекотен формат за обмяна на данни независим от програмния език, в който се използва. Съществуват парсери на JSON за всички програмни среди. В повечето от средите парсърът е част от стандартната библиотека на езика.

База данни

За база данни е използвана MySQL [34]. За улеснение при разработката е използвана една инстанция на MySQL сървър с две бази данни: една за Moodle и една за Activiti. Последното не е задължително, тъй като няма директна комуникация от една система към базата данни на другата. Комуникацията се осъществява чрез web услуги. В допълнение Activiti поддържа следните бази данни: DB2, H2, Oracle, MySQL, MS SQL, PostgreSQL. При необходимост е възможно да се добавят и нови такива (всяка база за която има JDBC driver). При Moodle положението не е по различно. Поддържаните бази са: Oracle, MySQL, MS SQL, PostgreSQL.

Уеб сървъри

Използвани са два популярни уеб сървъра. Tomcat на Apache, като сървър поддържащ Java за сървърен език за програмиране, и Servlet спецификацията (Activiti). За Moodle е използван Http Server, отново на Apache.

Инструменти за разработка

При разработката са използвани множество инструменти: Интегрирани среди за разработка – Eclipse EE Edition (Java, JavaScript), Zend Studio (PHP), администриране на бази данни – PhpMyAdmin, текстов редактор Gedit. Операционната система, използвана при разработката, е Ubuntu Linux Desktop 32bit.

Следват принципна схема и описание на изграждането на комуникационния канал между интегрираните системи и вграждането на редактора на процеси в Moodle.

Двупосочен комуникационен канал

Комуникационният канал е реализиран на базата на REST услуги. Activiti разполага с богат набор от услуги за достъп до възможностите на системата. За улеснение при разработката на Moodle plugin-a (Activity Rest Client) е създаден универсален слой за работа с Activiti чрез REST интерфейс.

От страната на Activiti, както бе уточнено по горе, се използва вече съществуващият REST слой за експортиране на функционалност и Spring Rest Template за консумиране на Moodle ресурси. Spring Rest Template е обвивка върху HTTP клиент и предлага автоматично конвертиране на JSON/XML до Java обекти.

Тъй като Moodle разширението за адаптивни курсове използва доста голяма част от REST услугите, е разработен слой Activity Rest Client. Той скрива сложността при работа с отдалечена/чужда система и представя изпълнението на процеси като локална PHP услуга. Допълнително са разработени няколко REST услуги за достъп до оценки на студентите при приключване на даден Moodle тест.

Интегриране на редактор за процеси

За интеграцията на уеб редактора на процеси са решени няколко технически предизвикателства. Първо, редакторът е част от Activiti платформата (Activiti Modeler) и трябва да се зарежда отдалечено от адаптивното разширение на Moodle и второ, при отдалеченото зареждане на съдържание съществува така нареченият Cross-Origin [16] проблем. За да работи с ресурсите на курса, редакторът трябва да има достъп до тях и тъй като е зареден от адреса на Activiti, той има достъп само до него. За справянето с проблема е разработена специална REST услуга за пренасочване. При пристигане на заявка от редактора за Moodle функционалност/ресурс то тя се пренасочва обратно към Moodle. По принцип при поставянето на двете системи под един адрес проблемът не съществува. Това обаче не е винаги възможно, а предложеното решение е универсално. Предполага се, че връзката между Moodle и Activiti е високоскоростна и пренасочването ще има минимално или никакво въздействие върху производителността на системата (1 Gbps е вече стандарт при локална свързаност, а при сървърни решения се прилага и оптичната свързаност и/или обединяване на мрежови интерфейси – network bonding).

След архитектурното представяне следва детайлно описание на програмната реализация, включваща промени и разширения направени по Moodle и Activiti.

3.2 Функционално разширяване на Moodle

В Moodle, за разлика от Activiti, е предвидена рамка за създаване на разширения (плъгини). Разработката на разширения/плъгини, използвайки тази рамка, не налага промени по основния Moodle код, а използва предварително определени места, на които разработчик може да „закачи“ (от англ. hook) допълнителна функционалност. Moodle предлага множество възможности за разширения, групирани в категории, съответстващи на различни Moodle функционалности (Потребител, Курс, Дейност, Тест и др.) (реф moodle-plugins). Разширението „Адаптивен курс“ използва именно тази рамка за създаването на нов вид курсов формат в Moodle. Важно е да се отбележи, че въпреки наличието на рамка за разработка на Moodle плъгини/разширения, документацията, описваща същинския процес на създаване на последните, е непълна. За създаването на едно разширение обикновено е необходима работа директно с изходния код на вече създадено разширение от същия тип, тъй като най-полезната информация се намира в коментарите към кода. Следователно, настоящото описание може да се разглежда като шаблон за създаване на разширения, имащи за цел добавяне на нов курсов формат.

В допълнение, Moodle предлага и рамка за експортиране на услуги/сервизи чрез интернет, използвайки протоколи като SOAP, REST и XML-RCP. Услугите, необходими за експортиране на данни от Moodle към Activiti, са разработени благодарение на тази рамка. Чрез използване на вече съществуващи механизми за разширение, предоставени от средата, се осигурява преносимост на разработката и в бъдещи Moodle версии.

3.3 Интегриране на редактор на процеси

За разлика от Moodle, където въпреки, че не е добре документирана, има рамка за създаване на разширения, във версията на Activiti, използвана при разработка на разширението, такава не съществува. В резултат на това са направени множество документиранни промени по базовия код на Activiti. Следното описание на промените по Activiti е използвано два пъти успешно за преминаване към по-нова версия на системата за управление на процеси по време на разработката на разширението.

Въпреки липсата на рамка за разширения, Activiti е модулна система и позволява изолирана промяна на части от нея. За стартиране на системата за моделиране и управление на процеси са необходими петнадесет модула. По шест от тях са правени допълнения и модификации. Това са activiti-bpmn-converter, activiti-bpmn-model, activiti-engine, activiti-json-converter, activiti-rest, activiti-webapp-explorer2. Въпреки броя на мо-

дулите, работата с Activiti е сравнително лека, тъй като се менажира, компилира и поставя на сървър автоматично чрез Maven [32]. Преди да се премине към същинското описание на промените е необходимо да се разяснят няколко основни положения при работата на системата за моделиране и управление на процеси.

Преди да може даден процес да бъде изпълнен той трябва да се моделира. Моделирането се извършва с редактор на процеси. Activiti разполага с два визуални редактора: уеб базиран (работи в браузър) и разширение за Eclipse (работи като plugin за Eclipse).

В случая на разширението „Адаптивен курс“, уеб базираният редактор се вгражда в Moodle. Преди да може процесът да бъде съхранен, активиран и изпълнен, той и елементите от които е съставен претърпяват няколко трансформации:

1. от Графика към текст (JSON)
2. от JSON към XML
3. от XML към динамично представяне в база данни и Java обекти

Описаните до момента трансформации са статични. Иначе казано, използват се за запазване на модел на процес, но не и за неговото инстанциране, изпълнение и проследяване. За да инстанцира, изпълнява и следи процеси, Activiti прилага още една финална трансформация към XML описанието на модела, а именно конвертира го в записи в база данни, които динамично променя в зависимост от резултата при приключването на дадена задача. Тази трансформацията е еднопосочна. С други думи, ако процес е активиран веднъж и неговото XML представяне е изтрито, Activiti не разполага с механизъм за възстановяване. Това не означава, че възстановяването е невъзможно.

Тъй като разширението „Адаптивен курс“ поддържа потребителски елементи за описание на задачи в процес (Moodle File, Moodle Quiz и др.), се налага включването им във веригата от трансформации, описани по горе. Това е и основната причина за допълнения и промени по базовия код на Activiti (модули `activiti-bpmn-converter`, `activiti-bpmn-model`, `activiti-engine`, `activiti-json-converter`).

Допълнителни разширения и промени са правени по редактора (модул `activiti-webapp-explorer2`) на процеси и по REST услугите, предлагани от активити (модул `activiti-rest`).

3.4 Заключение

В текущата глава е представена програмна реализация на система за моделиране и управление на ДПеО над популярни CeO (Moodle) и СУП (Activiti). Предложена е кон-

кретизация на схемата от глава 2. с представянето на архитектура на система за моделиране и управление на проекти за е-обучение на базата на СУП. Доказана е възможността за създаване на бързи прототипи на системи за моделиране и управление на ДПеО.

Глава 4 Реализация на други ДПеО

За доказване на приложимостта тук се представят реализации на три динамични проекта за е-обучение: автоматично генериране на тестови въпроси, автоматизирано генериране на метаданни и динамично създаване на е-курс. При реализацията и на трите проекта са използвани акумулативни учебни дейности [2].

4.1 Автоматично генериране на тестови въпроси

Предложена е и експериментира методика - как знания на оценявани обучавани (в дадена ПО) да се използват за конструиране на следващи тестови въпроси (в същата ПО). Методиката е общоприложима (не зависи от областта, дисциплината, обучаваните и др.) и адаптивна (вкл. с автоматично генериране на тестови въпроси относно понятия и отношения в конкретната ПО).

При съставянето на задачите, включени в е-тест, са възможни няколко подхода: а) авторски ('ръчно' създаване от експерти в ПО); б) автоматично генериране на базата на семантично описание на ПО; в) използване на техники от областта на компютърната лингвистика - извличане на данни от текст и/или Интернет-източници, включване на 'въпросно-ответни системи' и др.; г) избор измежду стандартен набор на шаблонни въпроси (с последващо редактиране за съответната ПО); д) автоматично генериране на базата на предишен опит при е-тестване, и др.

Разглеждат се конкретни решения, свързани с г) и д). На базата на разширената таксономия на Блум е създадена таблица [2], съдържаща стотици въпроси и шаблони на въпроси, независещи от изучавана ПО, дисциплина и тема.

Важно е да се отбележи, че въпросите в разглежданата таблица са от т. нар. 'акумулативен тип' [45], и могат безпроблемно да бъдат 'вградени' в автоматизирана система за е-тестване. Акумулативното генериране на въпроси позволява генериране на нови тестови въпроси от различен тип (на базата на отговори, натрупани в процеса на е-тестване). По същество, таблицата представя иновативна методика, широко приложима не само при създаване на тестови бази от данни и/или тестови задачи към отделни

учебни теми, но и за проектиране и създаване на системи за е-тестване с широко приложение.

В резултат на проведеното изследване относно автоматизирано генериране на въпроси за оценяване на по-високи равнища на знанието е създаден прототип на система за е-тестване (*ProtoSeT*). Прототипът се състои от **четири подсистеми** за: създаване на тестове, доставяне и визуализиране на тестови задания, управление на хранилище за тестови въпроси и сервиз за изпълнение и управление на тестови процеси. Първите две подсистеми обслужват клиентската част (за потребители – обучаващи и обучавани) на *ProtoSeT*, а останалите (изцяло сервизно ориентирани) - сървърната част. Комуникацията между модулите се осъществява с помощта на протокола SOAP [43], с което се осигурява платформена независимост и преносимост. Проведени са експерименти по съставяне на адаптивни тестове с използване на методики за измерване на всички нива на знанието по разширената таксономия на Блум (от ‘запамятаване’ до ‘създаване’). На базата на акумулирани отговори на въпросите от теста са генерирани стандартни тестови въпроси.

Автоматизирано генерираният тест в *ProtoSeT* не зависи от конкретната ПО, но се конструира на базата на степента на усвояване/разбиране (на съответната тема) от обучаваните. В този смисъл, *ProtoSeT* може да се определи като **адантивна** (следващ въпрос се генерира на базата на отговори, дадени на предходни въпроси от конкретната група обучавани) и **акумулативна** (отговорите се натрупват и се използват за съставяне на тестови въпроси от различен тип - множествен избор, истина/лъжа, попълване на празнини и др., за следващи групи оценявани обучавани) **тестова система**.

Реализирането на подсистема за акумулиращи въпроси (от програмистка гледна точка) е сравнително лесно, но за поддържане и на стандарти (напр. IMS) е необходимо специално разширение [44].

4.2 Автоматизирано генериране на метаданни

В разработката на съвременни СеО се правят опити за реализация на способността за адаптация към персоналните нужди на участниците в даден курс. Адаптивността в контекста на една СеО се дефинира по следния начин: учебните материали, предоставени на даден студент трябва да са съобразени с неговото ниво на знания и неговия профил.

За реализация на гореописаната адаптивност обикновено се използват технологии, свързани с моделиране и управление на процеси [6]. Обикновено множество от правила, описващи учебния процес, се кодират и залагат в система за моделиране и управление на процеси, която управлява учебния поток и предлага на потребителите да извършват предварително заложените дейности [6]. Може да се обобщи, че предоставянето на персонализирани УО се извършва на базата на профила на студента и предварително зададени метаданни (обикновено при създаването на курса) за учебните материали [48].

Основен проблем на гореописания подход е фактът, че с времето метаданните остават за разлика от постоянно развиващия се потребителски профил и прогрес.

За да се осигури съвместимостта на УО с множество системи, е препоръчително да се използва одобрен стандарт за метаданни като Learning Object Metadata [30] (LOM) например. Стандартът LOM се състои от 68 елемента, които описват даден УО както семантично така и синтактично. Някои от полетата на стандарта могат да се попълнят автоматично (от специализирани софтуерни инструменти), но повечето е необходимо да се попълват ръчно.

В [14, 53] са представени няколко автоматични (анализ на физическата структура на файла, анализ на съдържанието на документа и др.) и автоматизирани подходи за извличане на метаданни, които позволяват попълване на голяма част LOM полетата.

Процесът на извличане на метаданни чрез акумулиращи въпроси може да се дефинира чрез следните стъпки:

1. Избор на *подходящ учебен обект* за даден потребител на базата на неговия профил и вече събраните метаданни.
2. Интеракция на студента с персонализирания учебен обект.
3. Отговаряне на акумулиращи въпроси за текущия учебен обект.
4. Извличане на нови (актуализирани) метаданни за текущия учебен обект на базата на отговорите, дадени от студента.

С натрупването на отговори на акумулиращите въпроси се натрупват и потенциални метаданни. Тези метаданни/отговори се оценяват от преподавател. Отговорите с най-висока оценка се приемат за актуални метаданни за текущия учебен обект.

Ако няма задоволителни резултати след първата итерация със студенти, преподавателят може да отговори на въпросите сам, като по този начин предостави основата за генерирането на нови АТВ с подобно или по-ниско ниво на трудност (пр. множествен избор вместо свободен текст).

Прилагането на този подход за извличане на метаданни постоянно актуализира описанието на учебните обекти. Това е полезно както за търсене на учебни обекти, така и за реализиране на по-добра адаптация на учебния процес.

Представеният подход е комбинация от идеята за адаптивна и акумулираща система и идеята за автоматизирано генериране на метаданни в контекста на СеО. Реализацията ѝ ще доведе до появата на СеО от ново поколение – системи, интегриращи важните концепции: адаптивност, реализирана чрез система за управление и изпълнение на процеси и АТВ, позволяващи автоматизирано генериране на метаданни.

Употребата на АТВ за генериране на метаданни разпределя работата, обикновено извършвана от преподавателя, между студентите. Благодарение на постоянното обновяване на метаданните се подобрява и качеството на адаптивността на системата.

Друго преимущество на предложения подход се крие в това, че по-голямата част от потребителите биха предпочели интегрирането на нова функционалност (пр. адаптивно е-обучение) да се извърши в среда (СеО), която те познават и използват ежедневно (Moodle).

4.3 Динамично създаване на е-курс

Създаването на АСеО предполага наличие на стратегии за подбор на подходящи учебни обекти и дейности и определяне на тяхната последователност в процеса на виртуално обучение в зависимост от нуждите на обучаваните и поставените образователни цели.

В работата се предлага решение на проблема с въвеждане на т. нар. „акумулативни учебни е-дейности“, класифицирани според Подобрена таксономия на Блум (ПТБ). В този случай, създаването на адаптивни персонализирани е-курсове се свежда до избор между подходящи учебни е-дейности от съответната класификационна схема в зависимост от постиженията на обучавания и *информацията, натрупана при неговите изпълнения* на предходни е-дейности.

За една конкретна е-дейност – тестовото изпитване, подобен подход успешно е реализиран за проектиране и създаване на адаптивна тестова система [39], базирана на т. нар. „акумулативни тестови единици“ (АТЕ), Редът на предоставяне на АТЕ в процеса на виртуално обучение се задава с помощта на ориентиран граф, възли на който са АТЕ от съответната класификационна схема, а ребрата представят смисловите и логически връзки между тях. Различните стратегии за обхождане на графа осигуряват възможност

за динамично отчитане на равнището на знания на обучавания и неговите постижения, съобразно предварително заложените цели. Адаптивността се осигурява от алгоритъм за обхождане, който динамично избира поредния въпрос според равнището на трудност, подходящо за конкретния обучаван.

Натрупаната информация за постиженията на обучавания, включително неговите оценки, получени в резултат на предходни е-дейности, се използва за конкретизиране на заданията в съответните е-дейности на следващите стъпки. Например, отговори на обучавани, получени при разискване във форум с основна тема ‘Опишете (или дайте определение) за всяко основно понятие’, ще се използват при уточняване на задание от по-високо равнище (напр. „Сравнете понятията от темата с такива от други области“).

Нелинейното обхождане на работния поток се постига чрез използване на метод за интегриране на елементи от система за управление на потоци (СУП), представена в Business Process Model and Notation (BPMN) СеО. В конкретния случай системата Bonita Open Solution [21] е избрана за моделиращ инструмент. Едно от предимствата на системата за управление на потоци Bonita е, че стандартно се използват (BPMN 2.0) като нотации (означения), което подобрява съвместимостта с останалите системи, моделиращи потоци (Bonita Open Solution).

Осигурена е възможност за комуникация между Moodle 2.2.1 и Bonita Open Solution 5.6.2. чрез създаване на допълнителен модул, състоящ се от две части:

- *Инвазивна* – изисква промени в системата за е-обучение, с цел доставяне на точното съдържание на потребителите, от системата за управление на потоци;
- *Неинвазивна* – не се изискват промени в конкретната СеО. Тази част си служи с възможностите на Moodle да експортира данни чрез уеб услуги [33].

За конкретната реализация на архитектурата са необходими следните промени и разширения на Moodle и Bonita Open Solution:

- уеб услуги, които осигуряват на потребителите достъп до наличната информация за е-дейностите, включени в курса. Тези услуги са реализирани със стандартните средства на Moodle за разработване на допълнителни услуги.
- Bonita Connector, който достъпва информацията за е-дейностите, осигурена в Moodle, чрез описаните по-горе уеб услуги.
- модификация на начина на управление на съдържанието на е-дейности в Moodle, което позволява интерпретирането на потоците (BPMN 2.0).
- PHP клиент, който осигурява достъп до Bonita execution engine. Достъпът е осъществен чрез REpresentational State Transfer (REST) технологията. Разработен е специален Bonita REST PHP клиент, тъй като Moodle е PHP приложение.

4.4 Заключение

Реализираните посредством подхода и схемата от глава 2. три проекта за е-обучение, базирани на акумулативни учебни дейности (автоматично генериране на тестови единици, автоматизирано генериране на метаданни и динамично създаване на е-курсове) доказват възможността за създаване на бързи прототипи на системи за моделиране и управление на ДПеО.

Заключение

В глава 1. **Адаптивни системи за е-обучение** е направен преглед на състоянието на изследванията в областта и се анализира възможността за постигане на целта на дисертационния труд.

Глава 2. **Проектиране на системи за ДПеО** изследва подходи, подходящи за ефективна реализация на динамични проекти за е-обучение (ДПеО), базирани на СеО и СУП. На базата на проведен анализ е предложен общ подход и схема за реализация на проекти в областта на е-обучението. Разработена е архитектура за проектиране и реализация на системи за **адаптивно е-обучение** като надстройка на съществуващи СеО и СУП.

Глава 3. **Програмна реализация на проекта „адаптивен курс“** представя програмна реализация на система за моделиране и управление на ДПеО над популярни СеО (Moodle) и СУП (Activiti). Предложена е конкретизация на схемата от глава 2. с представянето на архитектура на система за моделиране и управление на проекти за е-обучение на базата на СУП. Доказана е възможността за създаване на бързи прототипи на системи за моделиране и управление на ДПеО.

В глава 4. **Реализация на други ДПеО**, отново посредством подхода и схемата от глава 2., са изследвани и реализирани други проекти за е-обучение, базирани на акумулативни учебни дейности, автоматично генериране на тестови единици, автоматизирано генериране на метаданни и динамично създаване на е-курсове.

В дисертационния труд са поставени и успешно решени следните **задачи**:

Задача 1. Изследване на **възможността за постигане на целта** (вкл. за осигуряване на *адаптивност в е-обучението*) на базата на *моделиране и управление на бизнес процеси*;

Задача 2. **Анализ на подходи**, подходящи за ефективна реализация на динамични проекти за е-обучение (ДПеО), базирани на СеО и СУП;

Задача 3. Създаване на **обща схема за разработване** на бързи прототипи на системи за ДПеО.

Задача 4. Програмна реализация на системи за моделиране и управление на ДПеО над популярни СеО и СУП;

Задача 5. Приложения: създаване на бързи прототипи

При решаването на задачи 1 – 5. бяха получени научни, научно-приложни и приложни приноси в областта на е-обучението както следва.

Научни приноси

1. Предложен е **общ подход и схема за реализация на проекти** в областта на е-обучението
2. На базата на 1. е предложена архитектура за проектиране и реализация на системи за **адаптивно е-обучение** като надстройка на съществуващи СеО и СУП

Научно-приложни приноси

3. Предложена е конкретизация на схемата от 1. под формата на **архитектура на система за моделиране и управление на проекти за е-обучение** на базата на СУП
4. Доказана е възможността за създаване на **бързи прототипи** на системи за моделиране и управление на процеси, базирана на архитектурата от 3

Приложни приноси

5. Реализиран е проект за **адаптивно е-обучение** (над конкретни системи – Moodle и Activiti)
6. Реализиран е проект за автоматично генериране на тестови въпроси (над конкретни системи – ProtoSet и ILNET)
7. Реализиран е проект за автоматизирано генериране на метаданни към учебни обекти (над конкретни системи – Moodle и ILNET)
8. Реализиран е проект за динамично създаване на съдържание на е-курс (над конкретни системи – Moodle и Bonita Open Solution).

В следващата таблица за всяка от задачи 1. – 5. са посочени разделите от дисертационния труд, публикациите по дисертационната тема, в които е представено нейното решение, и съответните приноси (1 – 8).

| Задача | Принос | Публикация | Раздел |
|-----------|--------------------------------|---------------------|-------------------------|
| Задача 1. | Научен принос 1. | П1., П5, П9 | 1.3., 2.1., 2.2. |
| Задача 2. | Научен принос 1. | П4., П10., П5., П9. | 1.3., 2.1., 2.2. |
| Задача 3. | Научен принос 2. | П4., П10. | 1.3., 2.2. |
| Задача 4. | Научно-приложни приноси 3., 4. | П4., П10. | 2.3., 3.1., 3.2., 3.3., |

| | | | |
|-----------|---|---------------------|---------------------------------------|
| | Приложен принос 5. | | |
| Задача 5. | Научно-приложни приноси 3., 4. Приложни приноси 5., 6., 7., 8. | П2.,П3.,П6.,П7.,П8. | 3.1., 3.2., 3.3., 4.1., 4.2., 4.3. |

Перспективи за развитие на дисертационното изследване са:

- Разпространение на подходите в друга област;
- Бърза реализация на прототипи за други проекти за е-обучение;
- Промяна и разширяване на функционалността на СеО с въвеждане на нови виртуални дейности и ресурси;
- Създаване на методи и софтуерни средства за моделиране, изучаване и управление на други аспекти от процеса на обучение (например автоматизирана оценка на качеството, методика на е-обучението, провеждане на адаптивни тестове, и др.)

Библиография

1. Благоев Д., Инджов Х., Тотков Г., Моделиране и управление на адаптивни учебни курсове, съставени от учебни дейности, базирани на Moodle, Трудове на годишна университетска научна конференция на НВУ „В. Левски“, 30.9. – 01.10.2010 г., Велико Търново, стр. 37-46.
2. Соколова М., Инджов Х., Тотков Г., Автоматизирано генериране на тестови въпроси, оценяващи знания по таксономията на Блум, 3-та Национална конференция „Образованието в информационното общество“, 27-28.5.2010 г., Пловдив, стр. 117-124, ISSN: 1314-0752.
3. Тотков Г., Д. Денев, Р. Донева, BEST: български образователен портал за е-проекти (с демонстрации), Втора национална конференция с международно участие по електронно обучение във висшето образование, 14-17 септември 2006, Китен, 69-79.
4. Тотков Г., Е. Сомова, Многофункционална среда от тип „виртуален университет“, 26-та Международна конференция „Информационни и комуникационни технологии и програмиране“, София, 26-29 юни 2001, 131-140.
5. Тотков Г., М. Соколова, Р. Донева, Организация и провеждане на тест в среда за електронно обучение PeU 2.0, Сборник на 34-та пролетна конференция на СМБ, Боровец, 6-9 април 2005, 396-400.
6. Тотков Г., Инджов Х. и колектив, Е-обучението в информационното общество: технологии, модели, системи, достъпност и качество, Пловдивско университетско издателство, Пловдив, 2010.
7. Activiti BPM Platform, <http://www.activiti.org/> .
8. Avgeriou, P., S. Retalis, and N. Papaspyrou., Modeling learning technology systems as business systems. *Software and Systems Modeling*, 2(2):120--133, July 2003.
9. Best of Elearning! Awards, <http://elceshow.com/best-of-elearning-awards/>.
10. BPMN 2.0 (www.omg.org/spec/BPMN) .
11. Brusilovsky, P. Adaptive and Intelligent Technologies for Web-based Education. In C. Rollinger and C. Peylo (eds.) *Kunstliche Intelligenz* (4), Special Issue on Intelligent Systems and Teleteaching, 19-25, (1999).

12. Brusilovsky, P., Methods and techniques of adaptive hypermedia. *User Modeling and User-Adapted Interaction Journal* Vol. 6, pp. 87-129, 1996.
13. Capterra Research Explanation, <http://www.capterra.com/blog/learning-management-systems/top-lms-software-solutions-infographic/>.
14. Cardinaels, K. Kris Cardinaels, Michael Meire, Erik Duval: Automating metadata generation: the simple indexing interface. 548-556 *Electronic Edition* (DOI: 10.1145/1060825) *BibTeX*. World Wide Web Conference Committee (IW3C2). WWW, May, 2005, Chiba, Japan.
15. Cesarini, M., M. Monga, R. Tedesco, Carrying on the e-Learning Process with a Workflow Management Engine. in *Proceedings of the 2004 ACM Symposium on Applied Computing* (Nicosia, Cyprus, March 14-17, 2004). SAC'04. ACM, New York, NY, 940-945. DOI= <http://doi.acm.org/10.1145/967900.968091>.
16. Cross-Site Scripting, https://developer.mozilla.org/en/docs/HTTP/Access_control_CORS.
17. Denev D., G. Totkov, R. Doneva, N. Kasakliev, Experimental Software Solution for e-Projects and e-Documents Management, *ACM International Conference Proceeding Series*; Vol. 285, *Int. Conf. on Computer Systems and Technologies, CompSysTech'07*, 14-15 June 2007, Rousse, II.15-1 – II.15-6.
18. Denev D., R. Doneva, G. Totkov, Best: New Lessons Module Logic, *Conf. of Computer Systems and Technologies, CompSysTech'2006*, V. Tarnovo, June 15-16, 2006, IV.5-1 IV.5-6.
19. eLearning Guild Academy, <http://www.elearningguild.com/>.
20. Enterprise Learning! Conference & Expo, <http://elceshow.com/>.
21. Hollingsworth D., The Workflow Reference Mode., The Workflow Management Coalition Specification, Document No. TC00-1003, Issue 1.1, 19-Jan-1995, <http://www.wfmc.org/standards/docs/tc003v11.pdf>.
22. IEEE Learning Technology Standards Committee: Draft standard for Learning Technology Systems Architecture (LTSA), Draft 9, November 2001 .
23. IIS, <http://www.iis.net/>.
24. JBPM, <http://www.jboss.org/jbpm/>.
25. Kruchten P., *The Rational Unified Process. An introduction*. Addison-Wesley, Reading, MA, 1999.
26. LAMS Users Guide v1.0.1, <http://lamsfoundation.org/>, 2005.
27. LDAP, <http://tools.ietf.org/search/rfc4510>.
28. Lin, J., A Method for Modeling Service Management of e-Learning. *WSEAS Trans. Info. Sci. and App.* 5, 7 (Jul. 2008), 1251-1261.
29. Lin, J., Ch. Ho, M. E. Orlowska, W. Sadiq, Using workflow technology to manage flexible e-learning services. *Educational Technology and Society*. 2002, 5 4: 1-10.
30. LOM – IEEE P1484.12 - Learning Object Metadata. 2002. Available: http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf. jun. 2004.
31. Mangan, P., S. Sadiq, On Building Workflow Models for Flexible Processes. In *Proc. Thirteenth Australasian Database Conference (ADC'2002)*, Melbourne, Australia. CRPIT, 5. Zhou, X., Ed. ACS. 103-109.
32. Maven, <http://maven.apache.org/>.
33. Moodle Activities. http://docs.moodle.org/20/en/Module_security.

34. MySQL, <http://www.mysql.com>.
35. NGINX, <http://nginx.org/>.
36. Oppermann, Reinhard; Rashev, Rossen; Kinshuk, Adaptability and Adaptivity in Learning Systems, 2004.
37. Pérez-Rodríguez R., Caeiro-Rodríguez M., Anido-Rifón L., Supporting PoEML Educational Processes in Moodle: a Middleware Approach, in Proceedings of SPDECER, 2008. International Conference on Computer Systems and Technologies - CompSysTech'13.
38. Pérez-Rodríguez R., Caeiro-Rodríguez M., Anido-Rifón L., Adding Process-Driven Collaboration Support in Moodle, 39th ASEE/IEEE Frontiers in Education Conference, October 18 - 21, 2009, San Antonio, TX.
39. Raykova, M, Kostadinova, Hr., Totkov, G., Adaptive Test System Based on Revised Bloom's Taxonomy, CompSysTech'11, 16-17 June 2011, Vienna, Austria.
40. REST, <https://www.ibm.com/developerworks/webservices/library/ws-restful/>.
41. Runa WFE, <http://wf.runa.ru/>.
42. Sadiq W., M. E. Orłowska, On Correctness Issues in Conceptual Modeling of Workflows. In Proceedings of the 5th European Conference on Information Systems ECIS '97, Cork, Ireland, June 19-21, 1997.
43. SOAP Specification, <http://www.w3.org/TR/soap12-part1/>.
44. Sokolova M., G. Totkov, Extended IMS Specification for Accumulative Test System, ACM International Conference Proceeding Series; Vol. 374, Proc. of the 9th Int. Conf. on CompSysTech, Gabrovo, Bulgaria, June 12-13, 2008, V.14-1–V.14-6.
45. Sokolova, M., G. Totkov, Towards accumulative e-learning systems. 3rd Balkan Conference in Informatics, 2007, Sofia, I.513 I.522.
46. Sokolova M., G. Totkov, Accumulative Question Types in e-Learning Environment, ACM International Conference Proceeding Series; Vol. 285, Int. Conf. on Computer Systems and Technologies, CompSysTech'07, 14-15 June 2007, Rouse, IV.21-1–IV.21-6.
47. Spring Framework, <http://www.springsource.org/spring-framework>.
48. Tang, T. & McCalla, G. Smart Recommendation for an Evolving E-Learning System: Architecture and Experiment. *International Journal on E-Learning*, 4(1), 105-129. Norfolk, VA: AACE, 2005.
49. The Rational Unified Process, v. 2001.03.00.23, Rational Software Corporation, part of the Rational Solutions for Windows suite, 2000.
50. Totkov G., E. Somova, M. Sokolova, Modelling of e-Learning Processes: an Approach Used in Plovdiv e-University, Int. Conf. on Computer Systems and Technologies (e-learning), CompSysTech'04, 17-18 June 2004, Rouse, IV.12.1 – IV.12.6.
51. Totkov G., R. Doneva, Object Modelling and Intelligent Resolving: Formal Models, Software Environment and Applications, Plovdiv, 2-5.09.2000, 258-265.
52. Totkov G., E. Somova, A Planning Model with Resources in e-Learning, Information Theories and Applications, 2003, Vol. 10, 218 – 226.
53. Warpechowski, M., Souto, M. A. M. & de Oliveira, J. P. M. Techniques for Metadata Retrieval of Learning Objects. Workshop on Applications of Semantic Web Technologies for e-Learning (SW-EL@AH'06), 2006.

54. WfMOpen, <http://wfmopen.sourceforge.net/>.
55. XFlow, <http://xflow.sourceforge.net/>.
56. Zend Framework, <http://framework.zend.com/>.

Публикации по темата на дисертацията

П1. Indzhov H., Blagoev D., Totkov G., *Executable Petri Nets: Towards Modelling and Management of e-Learning Processes*, Proceedings of the International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing - CompSysTech'09, 18-19.06.2009, Ruse, Bulgaria, IIIA.12.1-IIIА.12.6., ISSN: 1313-8936.

П2. Соколова М., **Инджов Х.**, Тотков Г., *Автоматизирано генериране на тестови въпроси, оценяващи знания по таксономията на Блум*, 3-та Национална конференция „Образованието в информационното общество”, 27-28.5.2010 г., Пловдив, стр. 117-124, ISSN: 1314-0752.

П3. Indzhov H., Sokolova M., Totkov G., *A Software Frame for Modelling and Runtime Control of Adaptive Testing*, International Conference on Computer Systems and Technologies - CompSysTech'10, 17-18.06.2010, Sofia, Bulgaria, pp. 370-375, ISBN: 978-1-4503-0243-2.

П4. Благоев Д., **Инджов Х.**, Тотков Г., *Моделиране и управление на адаптивни учебни курсове, съставени от учебни дейности, базирани на Moodle*, Трудове на годишна университетска научна конференция на НВУ „В. Левски“, 30.9. – 01.10.2010 г., Велико Търново, стр. 37-46.

П5. Тотков Г., **Инджов Х.** и колектив, *E-обучението в информационното общество: технологии, модели, системи, достъпност и качество*, Пловдивско университетско издателство, Пловдив, 2010, монография - съавтор, раздел 2.5.3 - стр. 141-147, раздел 5.6 стр. 312-319, ISBN: 978-954-423-651-9.

П6. Indzhov H., Totkov G., Doneva R., *E = MA² (E-Learning in a Moodle-based Adaptive and Accumulative System)*, International Conference on Computer Systems and Technologies - CompSysTech'11, 16-17.06.2011, Vienna, Austria, pp. 498-503, ISBN: 978-1-4503-0917-2.

П7. Костадинова Х., Тотков Г., **Инджов Х.**, *Акумулативни учебни e-дейности по Блум (с експеримент за персонализирано обучение в Moodle)*, 5-та Национална конференция „Образованието в информационното общество”, 31.05-01.06.2012, Пловдив, стр. 50-58, ISSN: 1314-0752.

П8. Kostadinova H., Totkov G., **Indzhov H.**, *Adaptive E-learning System Based on Accumulative Digital Activities in Revised Bloom's Taxonomy*. International Conference on Computer Systems and Technologies - CompSysTech'12, 22-23.06.2012, Ruse, Bulgaria, pp. 368-375, ISBN: 978-1-4503-1193-9.

П9. Stanev I., Grigorova K., *Knowledge Based Automated Software Engineering*, Cambridge Scholars Publishing, 2012, **Indzhov H.**, Doneva R., Blagoev D., Totkov G., *Modelling and Carrying on E-learning Processes with a Workflow Management Engine*, pp. 35-52, ISBN: 978-1-4438-3771-2.

П10. Indzhov H., Zhelyazkova P., Totkov G., *Supporting Adaptive E-Learning with an Open-source Learning Management System*, International Conference on Computer Systems and Technologies - CompSysTech'13, 28-29.06.2013, Ruse, Bulgaria, pp. 321-328, ISBN: 978-1-4503-2021-4.

Забелязани цитирания

1. **Indzhov H.**, Blagoev D., Totkov G., *Executable Petri Nets: Towards Modelling and*

Management of e-Learning Processes, Proceedings of the International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing - CompSysTech'09, 18-19.06.2009, Ruse, Bulgaria, IIIA.12.1-III A.12.6., ISSN 1313-8936

1. Korečko Št., Br. Sobota; R. Janoš, *Evaluation of Parallel Raytracing Strategy Improvements by Petri Nets*, Journal of Computer Science and Control Systems, Vol. 3, No. 1, May 2010, 87-92.
2. A. Guerriero, C. Pasquale, F. Quaranta, F. Ragni, *Java Grid Workflow management with fuzzy clustering approach*, in proceeding of CIMS A 2010 - IEEE International Conference on Computational Intelligence for Measurement System and Applications Taranto, Italy, 6-8 September 2010.

2. Тотков Г., **Инджов Х.** и колектив, *Е-обучението в информационното общество: технологии, модели, системи, достъпност и качество*, Пловдивско университетско издателство, Пловдив, 2010, монография - съавтор, раздел 2.5.3 - стр. 141-147, раздел 5.6 стр. 312-319, ISBN: 978-954-423-651-9.

3. Габриела Кирякова, Надежда Ангелова, Лина Йорданова, *Свободният софтуер в образованието*, Сборник на 4-та Нац. конференция „Образованието в информационното общество“, (ред. Г. Тотков и Ив. Койчев), 26-27 май 2011 г., Пловдив, Асоциация „Развитие на информационното общество“, София, стр. 96-103, ISSN: 1314-0752.
4. Габриела Кирякова, Надежда Ангелова, Лина Йорданова, *Web технологии и инструменти за създаване и управление на съдържание*, Сборник на 4-та Нац. конференция „Образованието в информационното общество“ (ред. Г. Тотков и Ив. Койчев), 26-27 май 2011 г., Пловдив, Асоциация „Развитие на информационното общество“, София, стр. 157-166, ISSN: 1314-0752.
5. Николай Касъкчиев, *Сравнителен анализ на информационни системи за висши училища*, Сборник на 4-та Нац. конференция „Образованието в информационното общество“ (ред. Г. Тотков и Ив. Койчев), 26-27 май 2011 г., Пловдив, Асоциация „Развитие на информационното общество“, София, стр. 139-148, ISSN: 1314-0752.
6. Илияна Чакърова, *Към дистанционно обучение във филиала на ПУ “Паусий Хилендарски“ – град Смолян*, Сборник на 4-та Нац. конференция „Образованието в информационното общество“ (ред. Г. Тотков и Ив. Койчев), 26-27 май 2011 г., Пловдив, Асоциация „Развитие на информационното общество“, София, стр. 167-176, ISSN: 1314-0752.

3. **Indzhov H.**, Totkov G., Doneva R., *E = MA² (E-Learning in a Moodle-based Adaptive and Accumulative System)*, International Conference on Computer Systems and Technologies - CompSysTech'11, 16-17.06.2011, Vienna, Austria, pp. 498-503, ISBN: 978-1-4503-0917-2

7. Newell, D., Davies, P., Atfield-Cutts, S., & Yearp, A., *Adaptive Virtualisation for Multi Modal Learning Objects*. In MMEDIA 2012, The Fourth International Conferences on Advances in Multimedia, Chamoni x, France, 29.04 – 04.05.2012, pp. 37-43, ISBN: 978-1-61208-195-3.
8. Stamatios A. Theocharis, George A. Tsihrintzis, *Education and Assessment of Civil Employees in e-Government: The Case of a Moodle Based Platform*, International Journal of Social, Human Science and Engineering, Vol. 7, No. 11, 2013, pp. 141-145.

9. Елена Паунова, *Проектиране и разработване на електронни игри за ученици като web-базирана услуга*, Дисертационен труд, Институт по Информационни и Комуникационни Технологии, БАН, 2013 г., София