

ОПЫТ РАЗРАБОТКИ И ВНЕДРЕНИЯ МОДУЛЯ «ЭЛЕКТРОННЫЙ ДЕКАНАТ» В СИСТЕМУ ДИСТАНЦИОННО ОБУЧЕНИЯ НА ОСНОВЕ LMS MOODLE

Макашова В.Н., Филимошин В.Ю.

Аннотация. Развитие электронного обучения является адекватным откликом систем образования многих стран на происходящие в мире процессы интеграции, движение к информационному обществу. В статье рассмотрены особенности электронного обучения. Приведены официальные документы, регулирующие вопросы использования электронного обучения в образовательной системе и наиболее крупные ВУЗы, успешно это реализующие. В работе обоснована необходимость создания модуля «Электронный деканат», который позволяет хранить и обрабатывать информацию о ходе учебного процесса и его участниках, а также автоматизировать взаимодействие между участниками учебного процесса. При разработке модуля были реализованы следующие задачи: идентификация студентов, добавление студентов, подписка студентов на курсы, отписка от курса, подписка преподавателей, формирование электронной зачетной книжки. Разработка модуля потребовала интеграции с уже имеющимися информационными системами университета. В работе приведены этапы разработки модуля и листинги программного обеспечения. Описана апробация модуля на примере системы дистанционного обучения ФГБОУ ВПО «Магнитогорский государственный технический университет им. Г.И. Носова». Материалы исследования могут использоваться при модернизации и развитии уже имеющихся систем дистанционного обучения, а так же для вновь разрабатываемых систем.

Ключевые слова: система дистанционного обучения, дистанционное обучение, электронный деканат, идентификация студентов, ИТ-инфраструктура, интеграция.

EXPERIENCE IN THE DEVELOPMENT AND IMPLEMENTATION OF THE MODULE "ELECTRONIC DEAN'S OFFICE" INTO DISTANCE LEARNING SYSTEM BASED ON LMS MOODLE

Makashova V.N., Filimoshin V.Y.

Abstract. The development of e-learning is an adequate response of the education systems of many countries on the world processes of integration, the movement towards the information society. The article describes the features of e-learning. Are the official documents governing the use of e-learning in the educational system and most major universities, is successfully implemented. The paper substantiates the necessity of the creation of the module "Electronic deanery," which allows you to store and process information on the educational process and its participants as well as to automate the interaction between the participants of the educational process. In the development of the module have been implemented the following tasks: identification of students, adding students, students on subscription rates, evasion of course, subscription Teachers, formation of electronic record-book. Development of the module required to integrate with existing information systems of the university. The paper presents the development stages of the module and software listings. The paper describes the testing of the module as an example of distance learning Nosov Magnitogorsk State Technical University. Research materials may be used for the modernization and development of existing systems of distance learning, as well as for newly developed systems.

Keywords: system of distance learning, distance learning, e-deanery, identification of students, IT infrastructure integration.

Введение

Актуальность развития электронного образования в России обуславливается потребностью повышения доступности образовательных услуг для широкого круга обучаемых, решения задач обеспечения мобильности обучаемых и роста их численности в условиях высочайшей конкуренции на рынке образовательных услуг. На данный момент в России разработан ряд официальных документов, регулирующих вопросы использования электронного обучения в образовательной системе. В федеральном законе №273-ФЗ «Об образовании в Российской Федерации» под электронным обучением понимается «организация образовательной деятельности с применением содержащейся в базах данных и используемой при реализации образовательных программ информации и обеспечивающих ее обработку информационных технологий, технических средств, а также информационно-телекоммуникационных сетей, обеспечивающих передачу по линиям связи указанной информации, взаимодействие обучающихся и педагогических работников». Вместе с понятием «электронное обучение» часто используют «дистанционное обучение». В федеральном законе №273-ФЗ говорится, что: «под дистанционными технологиями понимаются образовательные технологии, реализуемые в основном с применением информационных и телекоммуникационных технологий при опосредованном (на расстоянии) или не полностью опосредованном взаимодействии обучающихся и педагогических работников».

В приказе министерства образования и науки Российской Федерации № 137 от 06.05.2005 «Об использовании дистанционных образовательных технологий» предполагает-

ся, что под дистанционными образовательными технологиями понимаются образовательные технологии, реализуемые в основном с применением информационных и телекоммуникационных технологий при опосредованном или не полностью опосредованном взаимодействии обучающегося и педагогического работника. Образовательное учреждение вправе использовать дистанционные образовательные технологии при всех предусмотренных законодательством Российской Федерации формах получения образования или при их сочетании, при проведении различных видов учебных, лабораторных и практических занятий, практик (за исключением производственной практики), текущего контроля, промежуточной аттестации обучающихся.

Дистанционное обучение (ДО) обладает следующими особенностями: распределённость, гибкость, комплексность, интерактивность. Все эти особенности позволяют построить обучение таким образом, чтобы не было ни каких пространственных ограничений между обучающимися и преподавателями, выбирать более подходящий график обучения, использовать специально разработанные учебно-методические материалы, а также позволяют взаимодействовать между обучающимися и преподавателями с применением информационных и телекоммуникационных технологий. Вследствие этого использование дистанционных образовательных технологий становится всё более востребованным.

Первыми вузами, использующими ДО, были Международный институт менеджмента ЛИНК и МЭСИ, Современный гуманитарный университет. Сегодня ВУЗов, реализующих эту форму обучения, насчитывается больше сотни. Крупными центрами ДО на сегодняшний день является Москва, Санкт-Петербург, Томск, Красноярск, Новосибирск.

Для обеспечения эффективности образовательного и научно-исследовательского процессов, в том числе и дистанционных, в современных вузах строится сложная, масштабная и достаточно затратная ИТ-инфраструктура. Сегодня наблюдается резкий рост сложности интеграции существующих ИТ-инфраструктур с вновь вводимыми информационными системами. Существующие ИТ-инфраструктуры становится все труднее поддерживать из-за огромного числа межинтеграционных связей и сложным регламентом взаимодействия интегрированных ИС.

Так в Магнитогорском государственном техническом университете им. Г.И. Носова в настоящее время активно используется электронное обучение на основе системы дистанционного обучения (<http://lms.masu.ru>). Созданный портал дистанционного обучения позволяет: организовывать и сопровождать образовательный процесс, создавать электронные курсы, управлять студентами, управлять взаимодействием преподавателей, методистов и обучающихся, управлять журналами успеваемости и промежуточной аттестацией, формировать расписание занятий.

Система дистанционного обучения (далее СДО) создана на основе LMS (*Learning Management System*) Moodle, которая используется во многих вузах страны. Эта система универсальна, но в системе дистанционного обучения нет возможности: вести удобный полноценный учёт успеваемости студентов; просматривать оценки за экзамен/зачёт по дисциплине из другой системы вуза; подписывать глобальную группу сразу на множество курсов; массовой рассылки сообщений по определённым группам.

Для устранения описанных выше недостатков было принято **технологическое решение** разработать и внедрить модуль «Электронный деканат».

Реализация задач, поставленных при разработке и внедрении модуля «Электронный деканат», потребовала интеграции с существующими информационными системами. Это обусловливается тем, что в ВУЗе уже имелся определенный набор информационных систем, которые обслуживают те или иные бизнес-процессы, например, подсистемы «Студент», «Абитуриент», «Авторасписание», «Расчёт нагрузки», «Распределение нагрузки по преподавателям», «Индивидуальные планы». Поэтому задача интеграции с существующими информационными системами стоит достаточно остро.

Перед началом разработки необходимо было определить какие базы данных (далее БД) используются в СДО и в информационных системах (далее ИС) университета. В СДО БД

была установлена *MySQL*, в ИС университета используются приоритетно *MSSQL*. Исходя из этого, необходимо было установить расширение *mssql.so* для *PHP* на сервере СДО, чтобы с этого сервера можно было посылать запросы в БД ИС университета и получать обратно данные. Также необходимо было учитывать кодировки БД, в БД СДО используется по умолчанию *UTF-8*, в БД ИС университета, где находятся нужные данные, используется *CP1251*, поэтому при получении данных с БД ИС университета необходимо было их конвертировать в *UTF-8*. Для этих целей была выбрана встроенная функция в *php* – *iconv()*. Далее приведены этапы разработки электронного деканата.

Идентификация студентов системы дистанционного обучения в других системах

Для связи СДО с другими ИС университета нам необходимо было хранить уникальные идентификаторы из БД ИС университета в БД СДО. Решение было очень простым – добавлять уникальный идентификатор в поле таблицы пользователей, которое было предусмотрено разработчиками moodle для каких либо сторонних идентификаторов – *'idnumber'* в таблице *'*_user'* (* – в зависимости, что было выбрано при установке *LMS Moodle*, по умолчанию *«mdl»*). Идентификатор представляет собой целое число, например «114792».

Добавление студентов в СДО

В СДО есть возможность загружать массово пользователей, но для этого нужно формировать *csv* файлы с данными по студентам (ФИО, логин, пароль, *e-mail*, глобальная группа), которые хранятся в БД ИС университета. Изначально мы экспортировали данные из БД ИС университета в *excel* файл, приводили его в нужный вид для СДО по шаблону (вместе с конвертацией в *csv*) и проводили импорт.

Листинг 1. Шаблон из документации *LMS Moodle*.

```
Username, password, firstname, lastname, email, course1, group1, cohort1
jonest, verysecret, Tom, Jones, jonest@someplace.edu, math102, Section1, year3
reznort, somesecret, Trent, Reznor, reznort@someplace.edu, math102, Section3, year4
```

где: *username* – логин, *password* – пароль, *firstname* – имя и отчество, *lastname* – фамилия, *email* – адрес электронной почты, *course1* – курс, *group1* – локальная группа, *cohort1* – глобальная группа.

Нами использовались только нужные нам поля – это *username, password, firstname, lastname, email, cohort*.

Но мы посчитали этот способ не оптимальным, так как необходимо было формировать промежуточные файлы *csv*, на создание которых уходило много времени. Для оптимизации было принято решение напрямую из БД ИС университета экспортировать данные в СДО.

Для этого был написан скрипт (с применением *php, js, ajax, html, css*), который соединяется с БД ИС университета, выбирает нужные данные (ФИО, группа, уникальный идентификатор) и вносит их в БД СДО. Таким образом, мы избавились от промежуточных файлов *csv*, что существенно сократило время на добавление студентов в СДО. Написанный скрипт состоит из единственного запроса к БД ИС университета и алгоритмов различных проверок и различных запросов, которые в итоге либо добавляют нового студента или «размораживают» заблокированного, либо выдают сообщение, что такой студент уже существует в БД СДО. Так же скрипт сразу назначает глобальную роль студенту «студент».

При добавлении студента затрагиваются следующие таблицы:

- 1) *'mdl_user'* – основные данные о пользователе (ФИО, логин, пароль и т.д.);
- 2) *'mdl_role_assignments'* – в этой таблице присваивается глобальная роль студента;
- 3) *'mdl_crypt'* – дополнительно созданная зашифрованная таблица, для хранения логинов и паролей студентов. Эта таблица нужна для того, чтобы кураторы могли выдавать логины и пароли своим студентам.

Проверка на существование студента в БД СДО происходит на основе уникального идентификатора из ИС университета.

Листинг 2. Запрос на проверку по уникальному идентификатору.

```
$user_row=mysql_fetch_array(mysql_query("SELECT `id` `suspended` FROM `mdl_user`
WHERE `idnumber`='".$user_id."'"));
```

В этом запросе нужно обратить внимание ещё на то, что выбирается поле `suspended`, по которому можно определить, был студент заблокирован или нет, то есть если студент уже существует в БД СДО и он заблокирован, то значит необходимо снять блокировку с учётной записи (изменить значение поля `suspended`). Логин генерируется исходя из индивидуального номера студента.

Листинг 3. Генерирование случайного пароля для учётной записи студента.

```
For($i=0;$i<=9;$i++){
    if(mt_rand(1,2)=='1'){
        $password.=chr(mt_rand(48,57));
    }
    else{
        $password.=chr(mt_rand(97,122));
    }
}
```

Как видно из приведённого кода пароль состоит из 9 букв с цифрами, которые генерируются случайно, где символы по коду `ASCII` от 48 до 57 – это цифры, а от 97 до 122 – это латинские буквы.

Подписка студентов

Для удобства и быстроты подписки студентов на курсы было принято решение написать скрипт позволяющий выбирать сразу множество дисциплин для конкретной группы в одном месте. Скрипт позволяет выбрать нужную группу студентов (глобальные группы в СДО), после чего подгружаются все категории с курсами (с использованием рекурсии), как они расположены в СДО. В итоге необходимо отметить галочками нужные курсы и нажать на кнопку подписки. С помощью этого скрипта подписать группу студентов можно за считанные минуты, что позволяет экономить время и избавляет от сёрфинга по курсам в СДО.

Листинг 4. Скрипт вывода категорий и курсов.

```
Function course($category){
    $query=mysql_query("
        SELECT
            `id`,
            `fullname`
        FROM
            `mdl_course`
        WHERE
            `category`='". $category. "' And
            `visible`='1'
    ");
    $id_array=array();
    while($course_row=mysql_fetch_array($query)){
        echo '<p><input type="checkbox" name="course_'.$course_row['id'],'";
        if(group_course($course_row['id'])){
            echo ' checked';
        }
        echo '>',$course_row['fullname'],'</p>';
        $id_array[$i]=$course_row['id'];
    }
    unset($query,$course_row);
}
function categories($parent){
    $query=mysql_query("
        SELECT
            `id`,
            `name`
        FROM
            `mdl_course_categories`
        WHERE
            `parent`='". $parent. "'
    ");
    while($course_categories_row=mysql_fetch_array($query)){
        echo '<p>',$course_categories_row['name'],'</p>';
```

```

        course($course_categories_row['id']);
        categories($course_categories_row['id']); }
unset($query,$course_categories_row); }
categories(0);

```

В этом коде показаны две функции, работа начинается с функции *categories()* с нулевым значением, это значит, что «родителя» у этих категорий нет, то есть они являются системными категориями. Функция *categories()* работает рекурсивно, то есть запускает сама себя, пока функция не переберёт все категории в БД СДО. После вывода категории запускается функция *course()* со значением уникального идентификатора категории, которая выводит курсы внутри категории.

Функция *group_course()* не показана, в ней происходит проверка подписана ли глобальная группа уже на курс или нет, если подписана, то галочка ставится напротив конкретного курса. Скрипт подписывает студентов способом синхронизации глобальных групп. При записи на курс затрагиваются следующие таблицы в БД СДО:

- 1) ``mdl_enrol`` – информация о синхронизации глобальной группы;
- 2) ``mdl_user_enrolments`` – в этой таблице находятся *id* студентов, подписанные на курс;
- 3) ``mdl_role_assignments`` – роль студента в курсе;
- 4) ``mdl_groups`` – в эту таблицу добавляется локальная группа в курс;
- 5) ``mdl_groups_members`` – таблица для членов локальной группы.

Автоматическая подписка студентов

С каждым учебным годом контингент студентов увеличивался и встал вопрос о том, как автоматизировать подписку студентов на курсы. Изначально был модифицирован скрипт простой подписки для подписки глобальных групп по уникальным идентификаторам курсов. Также был написан скрипт, который импортировал планы в формате *xml* (планы сделанные на основе «УП ВПО (универсальный)» (ООО «Лаборатория ММИС»)) в СДО, и на их основе происходила автоматическая подписка всех имеющихся групп. Что позволило разово привязать уникальные идентификаторы курсов СДО к дисциплинам планов. Автоматическая подписка студентов стала занимать меньше одной минуты.

Листинг 5. Созданная таблица для хранения *xml* планов.

```

CREATE TABLE IF NOT EXISTS `mdl_plan_of_xml` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `faculty` varchar(20) NOT NULL,
  `specialty` varchar(255) NOT NULL,
  `course` tinyint(4) NOT NULL,
  `semester` tinyint(4) NOT NULL,
  `discipline` varchar(255) NOT NULL,
  `year_begin` varchar(4) NOT NULL,
  `id_course` varchar(50) NOT NULL,
  `credit_units` tinyint(4) NOT NULL,
  `exam` char(1) NOT NULL,
  `credit` char(1) NOT NULL,
  `control_work` char(1) NOT NULL,
  `coursework` char(1) NOT NULL,
  `count_year_learning` tinyint(4) NOT NULL,
  PRIMARY KEY (`id`))
ENGINE=MyISAM DEFAULT CHARSET=utf8;

```

При загрузке планов заполняются все поля, кроме ``id_course``, это поле заполнялось вручную, и, исходя из этого поля, происходила автоматическая подписка глобальных групп студентов. Позднее автоматическая подписка была доработана, чтобы корректно подписывались все студенты, планы начали подкреплять к глобальным группам студентов в СДО.

Листинг 6. Созданные таблицы для доработанного скрипта для хранения *xml* планов и привязка к группам.

```

CREATE TABLE IF NOT EXISTS `mdl_plan_head` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(100) NOT NULL,
  `institute` varchar(150) NOT NULL,

```

```

`faculty` varchar(150) NOT NULL,
`spec_1` varchar(255) NOT NULL,
`spec_2` varchar(255) NOT NULL,
`year_begin` year(4) NOT NULL,
`amount_year_learning` varchar(10) NOT NULL,
`pulpit` smallint(4) NOT NULL,
`creator` text NOT NULL,
PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
CREATE TABLE IF NOT EXISTS `mdl_plan_structure` (
`id` int(11) NOT NULL AUTO_INCREMENT,
`id_plan` int(11) NOT NULL,
`course` tinyint(4) NOT NULL,
`semester` tinyint(4) NOT NULL,
`discipline` varchar(255) NOT NULL,
`id_course` int(11) NOT NULL,
`credit_units` double NOT NULL,
`exam` tinyint(4) NOT NULL,
`credit` tinyint(4) NOT NULL,
`control_work` tinyint(4) NOT NULL,
`coursework` tinyint(4) NOT NULL,
`block` varchar(15) NOT NULL,
`pulpit` smallint(4) NOT NULL,
`hour_lecture` smallint(6) NOT NULL,
`hour_IWS` smallint(6) NOT NULL,
`hour_control` tinyint(4) NOT NULL,
PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
CREATE TABLE IF NOT EXISTS `mdl_plan_and_cohort` (
`id` int(11) NOT NULL AUTO_INCREMENT,
`id_plan` int(11) NOT NULL,
`id_cohort` int(11) NOT NULL,
`time` bigint(10) NOT NULL,
`deleted` char(1) NOT NULL DEFAULT 'n',
PRIMARY KEY (`id`)) ENGINE=MyISAM DEFAULT CHARSET=utf8;

```

В таблице ``mdl_plan_head`` хранится основная информация о плане, в таблице ``mdl_plan_structure`` хранятся все дисциплины и данные по ним, в таблице ``mdl_plan_and_cohort`` хранятся связи загруженных планов и глобальных групп.

Исходя из вышеописанных таблиц – скрипт берёт информацию со связями групп и планов, вычисляет на каком курсе сейчас находится группа (по её названию), выбирает все дисциплины, исходя из курса группы, и подписывает студентов способом синхронизации глобальных групп.

Автоматическая отписка студентов

После успешной аттестации студента, по какому либо курсу, его необходимо отписывать от этого курса. Отписывать каждого студента по отдельности нецелесообразно.

Был написан скрипт, который проверял наличие оценок в БД ИС «Студент» университета и на основе этих данных он либо отписывал студента от курса (при положительном результате), либо заносил студента в локальную группу задолжников в курсе (при отрицательном результате). Поиск дисциплины в БД ИС «Студент» университета происходит по названию дисциплины, номеру курса и семестра у конкретного студента.

Подписка преподавателей на курсы

Изначально курсов было мало и подписку преподавателей мы производили вручную. По мере увеличения курсов встал вопрос о разграничении прав доступа. Было решено создавать внутри курсов локальные группы для студентов (изолированные) и добавлять туда закрепленных в нагрузку преподавателей. Подписка преподавателей усложнилась из-за того, что приходилось не только подписывать преподавателя на курс, но так же отдельно записывать его в нужные группы студентов, поэтому на основе скрипта подписки студентов был написан скрипт по подписки преподавателей. Отличался он тем, что при выборе курса появ-

лся список локальных групп курса, в которые добавляются преподаватели, а так же назначается роль преподавателя внутри курса.

Электронная зачетная книжка

В СДО в каждом курсе имеется раздел «оценки», где можно просмотреть успеваемость студентов по промежуточным результатам (оценки за тесты, оценки за выполненные работы), а так же есть итоговый результат за дисциплину в целом. На основе этих результатов преподаватель мог выставить зачёт (если дисциплина имела форму аттестации – зачёт), либо оценку за экзамен, но при этом, в случае экзамена, оценка формировалась на основе итоговой оценки за курс, а также на основе сдачи экзамена (очно/заочно) самого студента. С одной стороны СДО не позволяет выставлять оценки за экзамен, либо выводить статус получил студент зачёт или нет, с другой преподаватель обязан заполнить бумажную ведомость и сдать в деканат. Результаты заносятся в ИС «Студент». Поэтому был написан скрипт, который выводил эти данные из БД ИС университета. Ни каких сложностей при формировании таблицы не возникло. Для конкретного студента в таблицу выводятся результаты по курсам и семестрам с наименованием дисциплины, ФИО преподавателя, а также дата выставленного зачёта или оценки (табл.).

Листинг 7. Запрос для получения результата по конкретной дисциплине, конкретного студента с БД ИС университета.

```
$point_row=mssql_fetch_array(mssql_query("
SELECT
    [Stud_Disc_Att].[Date_vAtt],
    [Spr_Level_Att].[NameR_Level],
    [Sotrud].[s_Family] as [surname],
    [Sotrud].[s_Name1] as [firstname],
    [Sotrud].[s_Name2] as [lastname]
FROM
    [Stud_Disc_Att]
INNER JOIN [Spr_Level_Att] ON
[Spr_Level_Att].[Kod_Level_vAtt]=[Stud_Disc_Att].[Kod_Level_vAtt]
INNER JOIN [Sotrud] ON [Sotrud].[Kod_Sotr]=[Stud_Disc_Att].[Kod_Sotr]
WHERE
    [Stud_Disc_Att].[Pers_Kod]='".$id_user[0]."' And
    [Stud_Disc_Att].[Name_DisGos]='".iconv('UTF8', 'cp1251', $discipline_row['discipline'])."' And
    [Stud_Disc_Att].[Num_Sem]='".$semester_for_student."'
));
```

После выполнения данного запроса выводится информация о преподавателе (*surname*, *firstname*, *lastname*), который поставил зачёт, дата оценивания (*Date_vAtt*) и соответственно сама оценка (*NameR_Level*).

Таблица

Фрагмент результатов запроса в системе «Электронный деканат»

Курс	Семестр	Дисциплина	Форма контроля	Преподаватель	ЗЕТ	Дата экзамена (зачёта)	Оценка
1	1	Русский язык и культура речи	Зачёт / Кон. Раб.	Анохина Светлана Анатольевна	3	Feb 11 2013	зачтено
1	1	Безопасность жизнедеятельности	Зачёт / Кон. Раб.	Чернобровкин Владимир Александрович	3	Feb 18 2013	зачтено
1	1	Социология	Зачёт / Кон. Раб.	Тюплина Ирина Анатольевна	3	Jan 11 2013	зачтено
1	1	Физическая культура	Кон. Раб.	-	0.98	-	-
1	1	Математика	Зачёт / Кон. Раб.	Сапрыкина Юлия Викторовна	3	Feb 18 2013	зачтено
1	2	Документная лингвистика	Кон. Раб.	-	2	-	-
1	2	Политология	Экзамен / Кон. Раб.	Тюплина Ирина Анатольевна	3	Jul 3 2013	отлично

Заключение

В итоге разработанный модуль «Электронный деканат» внедрён в действующую СДО (<http://lms.masu.ru>). При разработке, минимизированы ручной ввод информации за счет интеграции с уже имеющимися ИС. Опробование прошло в 2014-2015 учебном году по шести направлениям подготовки в 26 группах. Участники образовательного процесса отметили достоинства: предоставление и прекращение доступа к курсам участников учебного процесса в соответствии учебным планом и нагрузкой преподавателя; управление текущими, итоговыми оценкам, контрольными точками; формирование и отслеживание расписания, хранение и предоставление информации по текущему состоянию и истории учебного процесса в СДО. В результате были сделаны выводы об удобстве внедренного модуля и продолжении работ по его совершенствованию.

Список использованных источников

1. Ильина, Е.А. Организация самостоятельной работы студентов вуза с использованием автоматизированной обучающей системы: дис. канд. пед. наук / Магнитогорск. гос. ун-т. – Магнитогорск, 2010. – 191с.
2. Макашова, В.Н. Информационные системы и технологии / В.Н. Макашова, Г.Н.Чусавитина, А.Н. Старков. – Магнитогорск: МаГУ, 2011. – 188 с.
3. Макашова, В.Н. Развитие творческих способностей студентов вуза в условиях открытого образования: дис. канд. пед. наук / Магнитогорск: МаГУ, 2005. – 190 с.
4. Никуличева, Н.В. Дистанционное обучение с нуля до диплома // Социальный компьютеринг. – Москва, 2013. – № 1(2) – С.41-47
5. Разинкина, Е. М. Порталы как средство сетевого сотрудничества: монография / Е. М. Разинкина [и др.]. – Магнитогорск: Изд-во МаГУ, 2006. – 144 с.
6. Рубан, К.А. Обеспечение интероперабельности системы электронного обучения вуза (на примере Магнитогорского государственного технического университета имени Г.И. Носова) // Информатизация образования и науки. – 2013. – № 3. – С. 177-184.
7. Ильина, Е. А. Проектные решения для разработки программного модуля математической обработки результатов тестирования / Е.А. Ильина, Ю.Б. Кухта, А.В. Дьяконов // Математическое и программное обеспечение систем в промышленной и социальной сферах. –2011. – № 1-2. – С.234-241.
8. Ильина, Е. А. Технология тестирования знаний студентов с использованием системы Moodle / Е.А. Ильина, Л.Г. Егорова, А.М. Сердобинцев // Математическое и программное обеспечение систем в промышленной и социальной сферах. – 2011. – № 1-3. – С. 166-172.
9. Логиновский, О.В. Корпоративная информационная система крупного вуза как эффективный инструмент повышения качества управления / О.В. Логиновский, М.И. Нестеров, А.Л. Шестаков // Известия высших учебных заведений. Уральский регион. – 2013. – № 1. – С. 40-52.
10. Колокольцев, В.М. Университетский комплекс: интеграция и непрерывность / В.М. Колокольцев, Е.М. Разинкина // Высшее образование в России. – 2011. – № 5. – С. 3-10.
11. Колокольцев, В.М. Новые тренды в развитии технического образования // Аккредитация в образовании. – 2011. – № 7 (51). – С. 52-54.
12. Ошурков, В.А. Обзор стандартов в области управления рисками / В.А. Ошурков, В.Н. Макашова // Современные тенденции развития науки и технологий. – 2015. – № 1-5. – С. 79-81.
13. Ошурков, В.А. Методы минимизации ресурсных рисков в проектах разработки программных продуктов // В.А. Ошурков, В.Н. Макашова // Современные научные исследования и инновации. – 2014. – № 10-1 (42). – С. 71-77.
14. Макашова В.Н. Использование электронных образовательных ресурсов для активизации взаимодействия вузов и работодателей // Проблемы и перспективы развития образования в России. – 2011. – № 11. – С. 386-390.

Макашова Вера Николаевна – канд. пед. наук доцент кафедры бизнес-информатики и информационных технологий ФГБОУ ВПО Магнитогорского государственного технического университета им. Г.И. Носова. E-mail: makashova.vera@mail.ru.

Филимошин Вадим Юрьевич – ведущий инженер-программист УИТиАСУ ФГБОУ ВПО «Магнитогорский государственный технический университет им. Г.И. Носова». E-mail: flightofdeath@mail.ru.

Макашова В.Н., Филимошин В.Ю. Опыт разработки и внедрения модуля «Электронный Деканат» в систему дистанционно обучения на основе LMS Moodle // Математическое и программное обеспечение систем в промышленной и социальной сферах. – 2015. – №1. – С. 67-74.

Makashova, V.N. and Filimoshin V.Y. (2015). Experience in the development and implementation of the module "Electronic Dean's office" into distance learning system based on LMS Moodle . Software of systems in the industrial and social fields, 5 (1): 67-74.
