

## ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

УДК 004.4:004.9

В.Л. Капилевич

### АРХИТЕКТУРНОЕ ПРОЕКТИРОВАНИЕ СИСТЕМЫ ИННОВАЦИОННОЙ ОРГАНИЗАЦИИ И ПОДДЕРЖКИ УЧЕБНОГО ПРОЦЕССА В ВЫСШЕМ УЧЕБНОМ ЗАВЕДЕНИИ

Статья посвящена рассмотрению архитектуры программного комплекса, обеспечивающего автоматизацию планирования и организации учебного процесса крупного вуза с учетом современных тенденций интеграции российских высших учебных заведений в общеевропейское и мировое образовательное пространство. Основной задачей описываемых программных средств является обеспечение возможности интерактивного формирования для каждого студента вуза индивидуального учебного плана, систематизации и обобщению сформированных учебных планов с целью составления оптимального расписания учебных занятий и мониторинга их проведения. Приводится обоснование реализации данной системы как облачного сервиса, т.е. предоставления программного обеспечения как услуги (SaaS). Проводится анализ необходимого функционала данной системы, основанный на требованиях Федеральных государственных образовательных стандартов, кредитно-рейтингового подхода, индивидуализации траекторий обучения, концепций CDIO, PBL и др.

**Ключевые слова:** *информационная система, организация учебного процесса, индивидуальная траектория обучения, SaaS (Software as a Service).*

В настоящее время российская система высшего образования переживает серьезные изменения в организационной структуре, методах оценки знаний и методологиях обучения [1]. Данные изменения обусловлены глобализацией образования и требованиями современного бизнеса и производства. В последние годы гармонизация национальной образовательной системы, по крайней мере, с образовательными стандартами европейского сообщества очень актуальна [1, 2]. Заметим, что процесс модернизации образовательных систем в рамках Болонского и Копенгагенского процессов стал выходить за рамки европейского сообщества, охватывая все большее количество стран различных регионов мира [3].

Таким образом, гармонизация образовательных стандартов выдвигает перед всеми ведущими мировыми державами такие цели [1] в области реформации высшего профессионального образования, как создание четких и единых квалификаций, переход на двухуровневую систему подготовки, введение системы кредитов, обеспечение академической мобильности студентов и преподавателей, международное сотрудничество в обеспечении качества высшего образования, образование в течение всей жизни (LLL – Lifelong learning) и др. Немаловажным, говоря языком бизнеса, является ориентация на потребителя, т.е. в нашем случае установка студента в центр образовательного процесса.

На сегодняшний день множество российских высших учебных заведений находятся в процессе перехода к вышеназванным принципам построения процесса обучения и организации работы вуза, но они сталкиваются с множеством проблем. В том числе, с одной из самых трудоемких – полностью управляемый студентом план обучения, который требует тесного взаимодействия студентов, преподавателей курсов, составителей расписания, деканата, учебного управления и других структур учебного заведения. Очевидно, что необходимо внедрение инновационной системы, позволяющей управлять процессом обучения и контролировать его на всех этапах.

При проектировании данной информационной системы был сделан упор на введении принципиально нового системного подхода в организации образовательного процесса и сопутствующих ему управляющих процессов. Особенностью данного подхода является акцентирование внимания на индивидуализацию образовательных траекторий студентов, поддержку современных образовательных концепций (таких, как CDIO и PBL) [4] и обеспечение предельно дружественных в использовании компьютерных технологий для выполнения всех видов организационной деятельности. Кроме того, отдельное внимание при планировании было уделено социальным элементам системы.

### **1. Возможности системы**

Прежде всего, проектируемая система является хранилищем данных обо всех аспектах учебной и организационной деятельности в вузе. После проведения анализа работы типового университета были выделены следующие основные сущности: учетные записи студентов, преподавателей и других сотрудников, аудиторный фонд, набор дисциплин и привязанных к ним материалов, учебные планы (как базовые, так и индивидуальные), структура подразделений, условия составления расписания, индивидуальные расписания и некоторые другие вплоть до данных мониторинга текущей успеваемости.

Очевидно, система должна предоставлять отдельный набор операций в зависимости от роли вошедшего пользователя. Поэтому была имплементирована служба ролей с разграничением доступа к функциям системы. Данная служба позволяет пользователю иметь несколько ролей, что необходимо для внедрения в реальном учебном заведении, так как зачастую один и тот же сотрудник исполняет несколько обязанностей из различных ролей.

Каждая роль имеет свой фронт-энд для удобного доступа ко всем разрешенным функциям через веб-интерфейс. На первом этапе развития системы было принято решение сделать четыре типа фронт-энда: для студента, для преподавателя, для администратора учебного заведения или его подразделения и для свободного просмотра (гостя). В дальнейшем планируется добавить различные фронт-энды для других подразделений и сотрудников, например для ректора, работника деканата, заведующего корпусом и т.д.

Фронт-энд студента привязан к аккаунту студента, который создается и заполняется на этапе его поступления в учебное заведение (или при внедрении системы, в случае, если студент уже находится на обучении), после чего студенту выдаются данные для доступа. Студенту доступны следующие основные функции: редактирование и просмотр личных данных, запись на курсы, отслеживание соответствия индивидуального учебного плана обязательной образовательной про-

грамме и подсчет кредитов, просмотр индивидуального расписания, просмотр материалов по выбранным курсам, оценка курсов и преподавателей, просмотр текущей успеваемости и заданий, связь с преподавателем.

Фронт-энд преподавателя также привязан к аккаунту преподавателя, который создается или при вступлении в должность, или при внедрении системы для уже работающих преподавателей. Было принято решение включить в систему возможность преподавателю заявлять пожелания и требования к составлению расписания и записи на его курсы.

Получив доступ к аккаунту, преподаватель может добавить в систему свои курсы (или отметить себя преподавателем уже существующего в системе курса), добавить описание, рабочую программу, список занятий, учебно-методические материалы и др. Затем он должен определить ограничения для записи на курс. Такими ограничениями могут быть: максимальное количество слушателей, необходимый набор уже пройденных студентом предметов, институт, к которому должен относиться слушатель, и т.п. Кроме того, преподаватель может предъявить требования к аудиториям, в которых будут проходить занятия: наличие экрана с проектором, аудиосистемы, компьютеров для каждого студента, специального лабораторного оборудования, размещение аудитории в конкретном корпусе (корпусах) и т.п. Преподавателю доступны следующие основные функции: редактирование и просмотр личных данных, создание курсов и открытие их для записи студентов, просмотр личного расписания, просмотр списка студентов на курсах, публикация материалов по своим курсам, публикация заданий для самостоятельного выполнения студентами, ведение текущего контроля успеваемости и посещения занятий, связь со студентами (как массовая рассылка, так и индивидуальная).

Администраторы подразделений авторизуются в системе под одноименным фронт-эндом. Основная цель этих пользователей – ввод данных об аудиториях, преподавателях, студентах и т.д. На более поздних этапах разработки данной системы будут добавлены специальные роли для данных задач, такие, как отдел кадров или приемная комиссия. Они примут на себя обязанности по управлению всеми статическими данными (корпуса, аудитории, персонала и т.д.), возлагаемые на этапе внедрения системы на администраторов подразделений.

Также можно выделить отдельно роль и соответствующий фронт-энд технического администратора системы. В нем реализуется различный функционал по контролю работоспособности всех уровней системы, её обслуживанию: проверка базы данных на ошибки, редактирование прав доступа, контроль различных технических характеристик (нагрузка системы, сообщения об ошибках) и т.д. Кроме того, технический администратор осуществляет запуск различных ресурсоемких процедур, таких, как составление общего расписания или подсчета статистики.

Общий фронт-энд предназначен только для просмотра общедоступной информации. Одним из сценариев его применения является предоставление информации о студенте работодателю. Данный функционал присутствует во многих системах профессиональной сертификации, например в сертификационном центре Microsoft. Работодатель может по коду, предоставленному выпускником этого учебного заведения, получить информацию о пройденных курсах, их содержании и успеваемости студента.

Помимо доступа к функциям разрабатываемой информационной системы посредством веб-браузера планируется имплементировать Web API (англ. Web Application Programming Interface – сетевой интерфейс программирования прило-

жений) для решения проблемы доступа к данным и функциям системы с использованием других программ [5]. Данный Web API впоследствии будет использоваться при реализации приложений для мобильных устройств и нативных приложений для различных операционных систем. Такой дополнительный метод взаимодействия с системой, при наличии документации для API в свободном доступе, позволит разработчикам сторонних приложений расширять систему, создавать различные виджеты и т.д.

Важной функцией рассматриваемой информационной системы является непосредственно генерация учебного расписания, удовлетворяющего всему множеству заданных ранее условий. Эта задача является наиболее нетривиальной и ресурсоемкой в рамках разрабатываемой системы [6, 7], так как является NP-полной и в сравнении с распространенными системами генерации расписаний, работающими с небольшим набором групп (классов) и, в большинстве случаев, не берущими в расчет никаких дополнительных условий – данная информационная система работает с большим количеством индивидуальных планов студентов и учитывает различные условия, предъявленные преподавателями при объявлении курсов.

По причине высокой ресурсоемкости было принято решение выделить в сетевой архитектуре системы отдельный вычислительный кластер для задач генерации расписания. Данное решение принято для разгрузки центрального сервера системы, так как он отвечает за взаимодействие с пользователями и может существенно снизить время ответа при генерации расписания на том же сервере. Генерация расписания является полностью автономной задачей и не требует вмешательства в процессе выполнения. Таким образом, эта задача является идеальной для выполнения одним из сервисов облачных вычислений [8]. В настоящий момент выбор в пользу сервисов облачных вычислений оправдан как экономически, так и удобством внедрения и обслуживания [9]. Отсутствие необходимости закупки серверного оборудования, аренды помещения, обеспечения сохранности данных и т.д. делает облачные сервисы наилучшим инструментом для выполнения удаленных вычислений.

В общем случае генерация расписания происходит несколько раз в учебный год. Генерация инициируется администратором системы после проверки наличия всех необходимых компонентов: индивидуальных учебных планов, дисциплин, различных условий генерации и описания аудиторного фонда. В случае, если студент не предоставил индивидуальный учебный план, применяется базовый план для его специальности.

Хранилище данных накапливает данные по дисциплинам, индивидуальным учебным планам и расписаниям. После окончания учебного года, а порой и семестра, данная информация становится неактуальной, но не может быть удалена, так как может понадобиться для статистических или других целей. В данный момент в большинстве высших учебных заведениях такая информация наполняет бумажные архивы в худшем случае и неструктурированные электронные хранилища – в лучшем. Оставлять устаревшие данные в основной базе данных системы тоже является не лучшим решением, так как при малой полезности эти данные замедляют скорость запросов с каждым годом все больше и больше. По этой причине было принято решение выделить в сетевой архитектуре системы отдельное хранилище данных, которое не должно обеспечивать оперативный доступ к данным. Таким образом, будет снижена нагрузка на основную БД и произведена кластеризация данных по признаку их текущей полезности [10].

## 2. Сетевая архитектура системы

Базируясь на заявленном ранее функционале системы, была разработана следующая сетевая архитектура информационной системы (рис. 1).

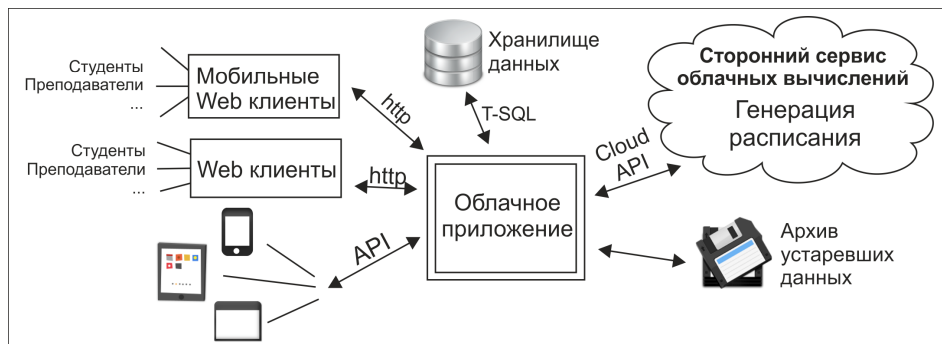


Рис. 1. Схема сетевой архитектуры системы информационной поддержки высшего учебного заведения

Основой архитектуры являются облачные веб-приложения и хранилище данных. Основное веб-приложение выполняет важнейшую функцию распределения прав, предоставления и обработки данных. Хранилище данных не имеет средств разграничения доступа, поэтому вся работа с ним происходит только через веб-приложение. На схеме база данных обозначена как сетевая единица, но при необходимости может быть масштабирована путем внедрения системы репликации без модификации других частей системы.

На левой половине схемы изображены возможные варианты клиентского доступа к системе: мобильные и обычные веб-клиенты (доступ посредством браузера к соответствующей версии веб-сайта), нативные программы для мобильных устройств и различных операционных систем (доступ посредством Web API). Все варианты доступа реализуются через единые механизмы основного веб-приложения для обеспечения идентичного поведения различных подсистем.

Особое внимание в сетевой архитектуре было уделено возможности масштабирования абсолютно всех частей системы, что позволяет реализовать данную систему как сервис облачных вычислений для предоставления клиентам. Являясь одним из популярнейших трендов на сегодняшний день – облачные сервисы являются наилучшим способом предоставления ПО сторонним компаниям (в нашем случае высшим учебным заведениям). Решение предоставлять систему по модели SaaS (Software as a Service, программное обеспечение как услуга) значительно уменьшает затраты на внедрение, полностью исключает затраты на серверное оборудование и построение инфраструктуры для каждого клиента, избавляет от проблемы обновления программного обеспечения каждому клиенту, позволяет имплементировать единые системы резервного копирования и т.д. [11, 12].

## 3. Внедрение и работа с системой

При заключении соглашения с высшим учебным заведением о внедрении данной информационной системы создается профиль университета в хранилище данных и добавляются основные администраторы (ответственные за сопровождение

системы в учебном заведении). Дальнейшая работа по сопровождению делегируется службе информатизации вуза. Следующим этапом является внесение в хранилище данных – аудиторий, студентов, преподавателей, и т.д. На этом, благодаря использованию системы как облачного сервиса внедрение непосредственно программного комплекса заканчивается.

Затем происходит публикация дисциплин, их описаний, материалов, требований преподавателями. После этого студенты выбирают дисциплины для изучения из списков доступных и составляют индивидуальный учебный план на ближайший семестр. Каждый учебный план проверяется на соответствие всем нормам и требованиям как учебного заведения, так и Федеральным государственным образовательным стандартам.

С использованием собранных данных составляется индивидуальное расписание для каждого студента и преподавателя, удовлетворяющее всем поставленным строгим требованиям и максимально подходящее под все нестрогие. Алгоритмически модуль генерации расписания основан на комплексном использовании эволюционных вычислений и метода штрафов. Данная комбинация позволяет включать дополнительные условия в процесс генерации и уменьшить время генерации расписания по сравнению с использованием этих методов отдельно или применения некоторых других методик [13]. Процедура составления расписания может проводиться несколько раз с параллельным изменением настроек или требований к расписанию, пока не будет получен оптимальный результат. В качестве объективного критерия оптимальности принимается минимальное отклонение от функции штрафов, задаваемой массивом требований к расписанию с различными значениями приоритета. Кроме того, возможна (субъективная) ручная оценка отдельных срезов расписания и последующая корректировка.

### Заключение

Следует отметить, что подобные системы существуют, но в основном внедрены за рубежом, например TechAct University Management System или AccelUMS. Использование этих систем в российских учебных учреждениях невозможно, так как они не учитывают особенностей переходного состояния нашей системы образования и имеют жесткую привязку к зарубежным стандартам делопроизводства и организации образовательного процесса.

Представленная в данной статье информационная система позиционируется для последующего внедрения в крупные высшие учебные заведения России, в частности в национальные исследовательские университеты.

### ЛИТЕРАТУРА

1. Волков А., Ливанов Д., Фурсенко А. Высшее образование: повестка 2008–2016 // Эксперт. 2007. № 32 (573).
2. Примерное положение об организации учебного процесса в высшем учебном заведении с использованием системы зачетных единиц: Приложение к письму Министерства образования РФ № 1-55-357 ин/15 от 09.03.2004.
3. Байденко В.И. Болонский процесс: поиск общности европейских систем высшего образования // Исследовательский центр проблем качества подготовки специалистов, 2006. 211 с.
4. Zhu Ming, Meng Li. Exploration and practice in engineering education reform of EE major based on CDIO mode // Engineering Education and Management. V. 112. Berlin, 2011. P. 19–23.

5. *Bianchini D., Valeria De Antonellis.* Semantics-enabled web API organization and recommendation // *Advances in Conceptual Modeling. Recent Developments and New Directions.* Heidelberg, 2011. P. 34–43.
6. *Carter M.W., Laporte G.* Recent developments in practical course timetabling // *Springer Practice and Theory of Automated Timetabling II.* London, 1998. P. 3–19.
7. *Barry McCollum.* University timetabling: bridging the gap between research and practice // *5th International Conference on the Practice and Theory of Automated Timetabling.* Amsterdam, 2006. P. 13–35.
8. *Dong, B., Zheng, Q., Yang, J., Li.* An E-learning ecosystem based on cloud computing infrastructure // *Advanced Learning Technologies, Ninth IEEE International Conference.* Riga, 2009. P. 125–127.
9. *Vinay Chawla, Prenul Sogani.* Cloud computing – the future // *High Performance Architecture and Grid Computing, Communications in Computer and Information Science.* V. 169. Heidelberg, 2011. P. 113–118.
10. *Moore R.W.* Archiving experimental data // *Encyclopedia of Database Systems.* Springer, 2009. P. 132–135.
11. *Иванченко Д.А.* Построение информационной инфраструктуры вуза с применением модели SaaS // *Высшее образование в России.* 2010. № 10. С. 121–126.
12. *Doelitzscher F., Sulistio A., Reich C., Kuijs H., Wolf D.* Private cloud for collaboration and e-Learning services: from IaaS to SaaS // *Computing Magazine* V. 91. No. 1. Wien, 2000. P. 23–42.
13. *Sadaq Naseem Jat, Shengxiang Yang.* A hybrid genetic algorithm and tabu search approach for post enrolment course timetabling // *J. Scheduling.* V. 14. No. 6. Amsterdam, 2011. P. 617–637.
14. *Капилевич В.Л., Ботыгин И.А.* Архитектура системы информационной поддержки инновационной организации и планирования учебного процесса в высшем учебном заведении // *Сборник трудов IX Всероссийской научно-практической конференции студентов, аспирантов и молодых ученых «Молодежь и современные информационные технологии».* Томск, 2011. С. 206–207.

*Капилевич Вячеслав Леонидович*  
Томский политехнический университет  
E-mail: skkapi@gmail.com

Поступила в редакцию 4 ноября 2011 г.

*Kapilevich Viacheslav. L.* (Tomsk Polytechnic University). **Architectural design of the system for university innovation management and educational process supporting.**

Keywords: information system, educational process management, individual studying trajectory, SaaS (Software as a Service).

In design process of this information system, emphasis was placed on the fundamentally new systematic approach for organizing the educational process and associated control processes. A feature of this approach is the emphasis on individualized educational trajectories of students, support of modern educational concepts (such as CDIO, PBL, etc.) and providing user-friendly computer technology for all kinds of organizational activities.

System presented in this article is positioned for future implementation in major universities in Russia, particularly in the national research universities.