

АВТОМАТИЗАЦИЯ УЧЕТА РЕЙТИНГА УСПЕВАЕМОСТИ СТУДЕНТОВ

Обсуждаются подходы к системе автоматизации учета рейтинга успеваемости студентов в рамках университета. Формулируются основные требования, структуры данных рейтинга и алгоритмы его вычисления, дается описание типовых траекторий обучения. Предлагается масштабируемая архитектура локальных баз данных, desktop- и мобильных приложений, дается описание функционала клиентских приложений.

Ключевые слова: успеваемость, рейтинг, автоматизация, Android, desktop, Java, база данных, MySQL.

В течение ряда лет в Новосибирском государственном техническом университете (НГТУ) для оценки успеваемости студентов используется модульно-рейтинговая система [1]. Наряду с традиционной системой оценок в ней применяется 100-балльная шкала, а также 15-уровневая шкала ECTS. С начала учебного 2013/14 года оценка по 100-балльной шкале является обязательной во всех нормативных документах в учебном процессе нового набора. В этом свете становятся весьма актуальными вопросы об автоматизации процессов хранения и обработки данных, связанных с вычислением рейтинга. Отметим основные организационные, методические и психологические аспекты автоматизации:

- уровень формализации процесса вычисления рейтинга. С одной стороны, по максимуму формально должны учитываться разные мелочи, а с другой – оговариваться, где и в каких диапазонах присутствует экспертная (субъективная) оценка преподавателя;

- ведение архива отчетных документов. Текущий документооборот по учебному процессу достаточно велик: отчеты по лабораторным работам, рефераты, архивы программных проектов и т.п. Все это желательно сохранять для исключения возможного плагиата, разрешения конфликтных ситуаций, подбора материалов для аттестаций, конкурсов, проверок и т.п.;

- очевидность оценки в стандартных случаях. Если раньше должник, сдавший все семестровые задания не в срок, с низким качеством, не имел возможности получить выше «тройки» по критерию «само собой разумеется», то сейчас такой здравый смысл должен обеспечиваться рейтингом;

- мобильность и автономность клиента. При проведении занятий преподавателю приходится постоянно перемещаться между стационарными рабочими местами для просмотра результатов

работ, поэтому для собственных нужд удобно было бы пользоваться мобильным приложением, а при возможных проблемах со связью – иметь локальные копии необходимых данных;

- доступность данных о текущем рейтинге для студентов;

- возможность дальнейшего использования (экспорта) накопленных данных.

Структуры данных рейтинга и алгоритмы его вычисления. Естественно, что предлагаемая ниже схема вычисления рейтинга не может охватить всех форм организации и контроля учебного процесса. Она рассчитана на традиционную форму ведения учебных дисциплин, довольно проста и учитывает только ограниченное количество факторов. Но прежде чем ее излагать, определимся в терминах:

- **предмет** – оригинальная дисциплина учебного плана, читаемая в одном семестре и имеющая одну итоговую оценку. Если дисциплина учебного плана читается в нескольких семестрах, в каждом из них она оформляется в системе отдельной дисциплиной. Аналогично курсовая работа или проект оформляется отдельной дисциплиной;

- **единица контроля дисциплины** – составная часть дисциплины, по которой выносится оценка, включающаяся в рейтинг;

- **рейтинг** – набор данных для одной учебной группы и одной дисциплины (читаемой в одном семестре и имеющей одну итоговую оценку);

- **хранилище** – область данных ограниченного объема, предназначенная для временного хранения документов (отчеты, архивы программных проектов, собранные данные и т.п.). Преподаватель периодически скачивает файлы из хранилища в собственный архив.

Описание структуры дисциплины представляет собой список учебных единиц (лабораторные работы, их защита, семинары, контрольные, зачет, экзамен и т.п.), за каждую из которых

устанавливается максимальный балл, их сумма должна составлять в рейтинге 100 баллов, так как это определяет уровень итоговой оценки. В текущей версии системы способ подсчета баллов по каждому виду учебной единицы жестко задан (т.е., например, для всех лабораторных работ он одинаковый). Имеется ряд вариаций этого процесса:

- **способ подсчета** – ручной (экспертный, субъективный) – исходный балл ставится преподавателем в пределах указанного максимума, система в формировании оценки участия не принимает (экзамен, зачет, индивидуальное задание, курсовая работа), автоматический – исходный балл устанавливается по нормативу и снижается / увеличивается по формальным критериям;

- **формальные критерии качества** – при автоматическом подсчете балла введен список критериев (например, небрежность оформления, повышенная сложность), по которым производится увеличение / снижение балла на фиксированный процент по каждому из них (например, на 10%). Признаки можно выбирать группой;

- **учет сроков сдачи** – для ряда учебных единиц (лабораторная работа, защита работы, контрольная работа) вводится линейная шкала снижения балла на заданный процент за каждую просроченную неделю относительно установленного срока сдачи. Срок сдачи каждой учебной единицы устанавливается при создании рейтинга в соответствии с расписанием (например, срок сдачи лабораторной работы – следующее занятие за ее выполнением). Число недель, в течение которых балл снижается, ограничено, т.е. вводится предельное снижение (например, 50%). При досрочной сдаче производится симметричное увеличение;

- **учет посещаемости** – для некоторых видов учебных единиц предусмотрен контроль посещаемости (лабораторные работы), они вносятся в отдельный список при создании рейтинга, этот список может дополняться преподавателем. Поскольку фиксируется факт пропуска занятия, за который в рейтинге предусмотрено фиксированное снижение рейтинга (например, 1 балл за пропуск), то учет посещаемости не является обязательным.

Чтобы варьировать степень влияния разных факторов, вводится ряд весовых коэффициентов, из них составляются типовые наборы и назначаются отдельно каждому рейтингу:

- процент снижения (увеличения) балла по сроку сдачи (7–10% за неделю);

- количество недель, в течение которых действует предыдущее снижение (увеличение) – (5–7 недель, т.е. максимум 50% в обе стороны);

- процент изменения балла за каждый параметр качества (10%);

- количество баллов, снимаемых за каждый пропуск (0,5–1 балл).

Рейтинг и типичные траектории обучения. Используемый способ вычисления рейтинга позволяет объективно оценить типовые траектории обучения, которых обычно придерживаются студенты. Но если при неформальной системе оценки преподаватель стремится к некоторому компромиссу между законом (качеством выполненной работы) и справедливостью (отношением студента к ее выполнению), то формальный подход должен внести этот принцип в технологические рамки. Итак, какие возможны варианты накопления рейтинга?

- *«Двоек не ставим».* Если вся учебная нагрузка выполнена в срок и без претензий (но и без «экслюзива»), то студент зарабатывает рейтинг, достаточный для получения «тройки» на экзамене. При этом средний и высокий уровень ответов на экзамене оценивается соответственно.

- *«Досрочно и автоматом».* Если студент сдает все досрочно, то ему достаточно небольшого индивидуального задания для получения «автомата». Можно также сдавать часть экзамена, например без теории.

- *«Так себе, на троечку».* При наличии небольших долгов (невыполнение или сдача не в срок) студент может быть допущен к экзамену, но по общему рейтингу выше «четверки» не получит, а реально может претендовать только на «тройку».

- *«Оптом не дешево».* Если студент в течение семестра не занимается, то, сдав в конце семестра полностью все задания, он получает максимум 50% баллов от исходного, т.е. необходимый минимум для допуска к экзамену. При этом все должно быть выполнено без претензий к качеству. На экзамене он фактически получает на 1 традиционный балл ниже, т.е. вместо 5 – 4 и т.п. и ниже 3 сдавать не имеет права (справедливость торжествует, порок наказан).

Состояние вопроса. Автоматизация «сверху» или «снизу»? Центр автоматизации университета

(ЦИУ) НГТУ выполняет широкий спектр работ по автоматизации различных сторон деятельности университета [2]. Что касается непосредственно учебного процесса, сюда входят документационное сопровождение (учебные планы, рабочие программы, приказы, протоколы ГЭК, аттестация и т.п.), а также ведение баз данных (БД) успеваемости и текущий ее контроль (контрольные недели). Информационная система имеет открытые и закрытые (корпоративные) ресурсы, все преподаватели и студенты – аккаунты и личные кабинеты. Большинство подсистем имеют стандартный web-интерфейс и доступны через обычный браузер.

В то же время текущий документооборот, связанный непосредственно с проведением учебного процесса, осуществляется подручными средствами – лабораторные журналы, отчеты и пояснительные записки в электронном виде или в твердой копии, в лучшем случае – Excel с доработанными формулами расчета.

На наш взгляд, имеются важные сдерживающие факторы внедрения централизованной информационной системы на этом уровне:

- информационная безопасность. Обычно доступ к корпоративным ресурсам осуществляется с рабочего места преподавателя, здесь же речь идет об учебной аудитории или лаборатории, в которой есть вероятность оставить без контроля «залогиненную» программную компоненту;
- объем архивных данных и текущий трафик;
- отсутствие постоянного доступа к компьютеру. Несмотря на тотальную компьютеризацию, возможны ситуации, когда компьютера просто нет под рукой, например при проверке посещаемости в начале лекции либо при приеме заданий в учебной аудитории и т.п.;
- разделение ответственности и обязательный регламент. До сих пор ответственность за текущий документооборот по учебному процессу несли преподаватели, разделение этой ответственности с централизованными службами в массовом масштабе может породить ненужные трения и психологический дискомфорт;
- свобода выбора преподавателя – нормативные документы НГТУ определяют рамочные конструкции для вычисления рейтинга по основным схемам учебных дисциплин, а преподаватель в рабочей программе дисциплины наполняет их

конкретным содержанием. С этой позиции введение единообразия не имеет под собой необходимых оснований.

В целом же автоматизация «сверху» не смотрится по общесистемным соображениям. Она может породить неэффективного, слабо управляемого монстра. Альтернативным подходом является автоматизация «снизу» путем разработки автономных и полуавтономных, в том числе и мобильных приложений, использующих локальные БД и собственные копии данных.

«Видение» проекта. Система учета рейтинга, организованная «снизу», реализуется в виде множества локальных БД, каждая из которых обслуживает ограниченную группу преподавателей и предметов (например, по циклу дисциплин, кафедре, образовательной программе). Организация БД достаточно проста и допускает совмещение функций администратора БД и куратора цикла (архивирование данных, ввод метаданных, касающихся структуры рейтингов). Хостинг и администрирование самих БД осуществляются либо системными администраторами структурных подразделений (кафедр), либо централизованными службами университета (центр информатизации). Возможен вариант автономной работы без централизованного хостинга с периодической синхронизацией данных. Клиентские приложения могут использовать как свои собственные логины/пароли, так и логины/пароли личного кабинета центра автоматизации.

Источники данных и варианты организации системы. Клиентские приложения позволяют работать как в режиме общей БД для нескольких клиентов, так и в режиме монопольного использования локальной БД или локальных копий данных рейтингов. Система естественным образом масштабируется: любой клиент может иметь **список входов** для подключения к различным **источникам данных**. В качестве источников выступают:

1. Серверная БД MySQL в режиме разделения (удаленный доступ использованием библиотек JDBC).
2. Web-сервер с протоколом http.
3. Локальная БД SQLite.
4. Локальный каталог файлов данных рейтингов, загруженных из перечисленных выше источников.



Рис. 1. Архитектура системы

Разнообразие источников позволяет реализовать два режима работы группы преподавателей с БД:

- при наличии хостинга БД или web-хостинга – множественный доступ в режиме разделения (источники 1,2);
- при отсутствии постоянного и общедоступного хостинга – работа ведущего преподавателя с локальной БД (источник 3) и раздача файлов данных рейтингов (источник 4) для автономной работы с периодической синхронизацией с БД-источником.

Архитектура системы. Все компоненты разработки выполнены на Java (рис. 1). Реализована многослойная архитектура с общим программным кодом для всех вариантов серверных и клиентских приложений, включающая следующие компоненты:

- уровень представлений – реализация внешних форм клиентских приложений;
- контроллер бизнес-логики – алгоритмы поведения для всех клиентов;
- модель бизнес-логики (классы бизнес-объектов) – внутреннее представление рейтингов, групп, студентов, оценок и т.п. и их основных алгоритмов;
- средства сборки бизнес-объектов из табличных объектов БД;
- конвертеры бизнес-объектов в двоичные файлы и в XML-формат;
- табличные объекты БД (DAO – Data Access Objects);

- интерфейс к библиотекам JDBC и SQLite;
- серверные компоненты web-сервиса (сервелеты).

Система включает 6 основных видов компонентов и приложений.

Сервер базы данных (MySQL) со стандартными средствами сетевого доступа удаленных Java-клиентов с использованием библиотек JDBC.

Web-сервер с API обмена данными рейтингов.

Клиентское приложение «Администратор» обеспечивает весь набор функций администрирования, кроме непосредственного хостинга БД MySQL (рис. 2):

- редактирование списка входов, подключение к БД, инициализация БД, экспорт и импорт данных;
- ведение списков групп, студентов, дисциплин, учебных единиц и их параметров (структура дисциплины), управление рейтингами;
- ведение учетных записей преподавателей, включая списки разрешений на редактирование рейтингов;
- редактирование наборов параметров расчета рейтинга;
- переход к клиентскому приложению «Преподаватель» с полным списком рейтингов.

Клиентское приложение «Преподаватель» (рис. 3) имеет следующий функционал:

- настройку списка входов БД;
- подключение к БД и авторизацию (собственный логин и логин ЦИУ НГТУ);

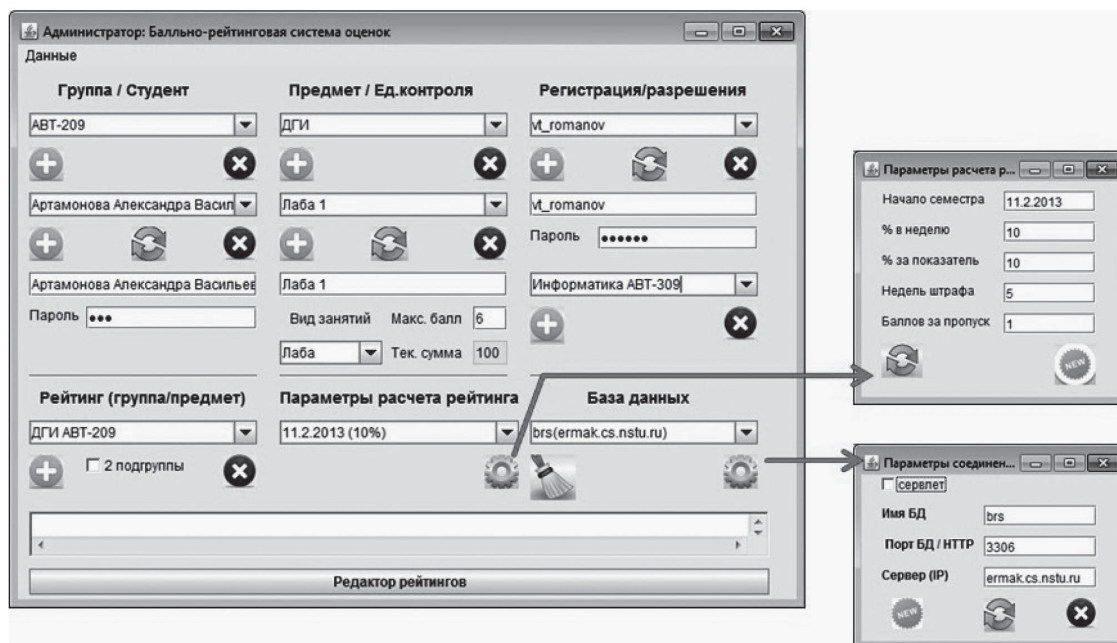


Рис. 2. Клиентское приложение «Администратор»

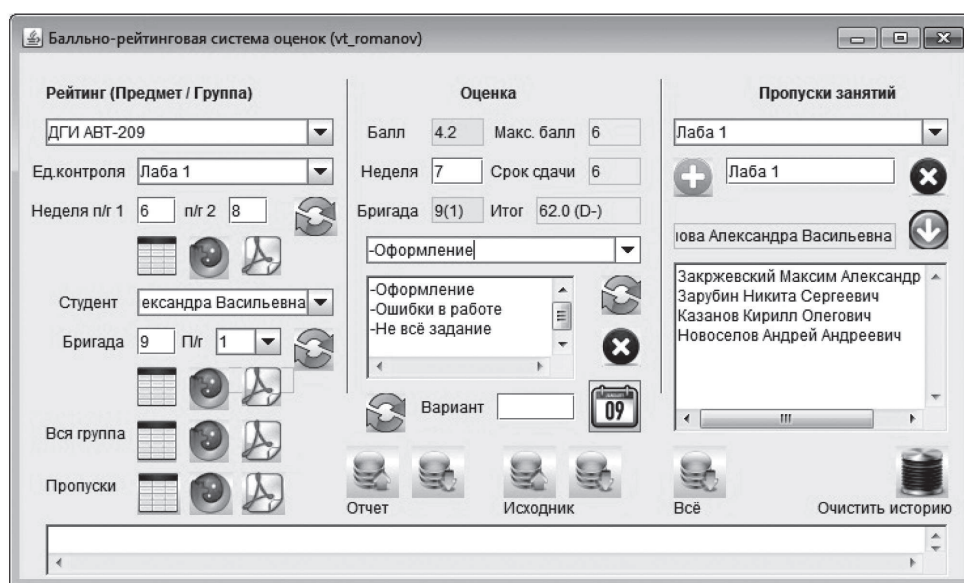


Рис. 3. Клиентское приложение «Преподаватель»

- выбор рейтинга;
- редактирование данных рейтинга – сроки сдачи, номера бригад и вариантов заданий, оценки, посещаемость;
- генерацию отчетов в форматах pdf, html и виде окон приложения – по учебной единице, студенту, пропускам и успеваемости группы;

- загрузку и просмотр файлов в хранилище;
- скачивание файлов из хранилища;
- просмотр и очистку истории редактирования.

Android-приложение «Преподаватель» рис. 4 может работать в режиме online-подключения к выбранной БД и в режиме offline – автономной

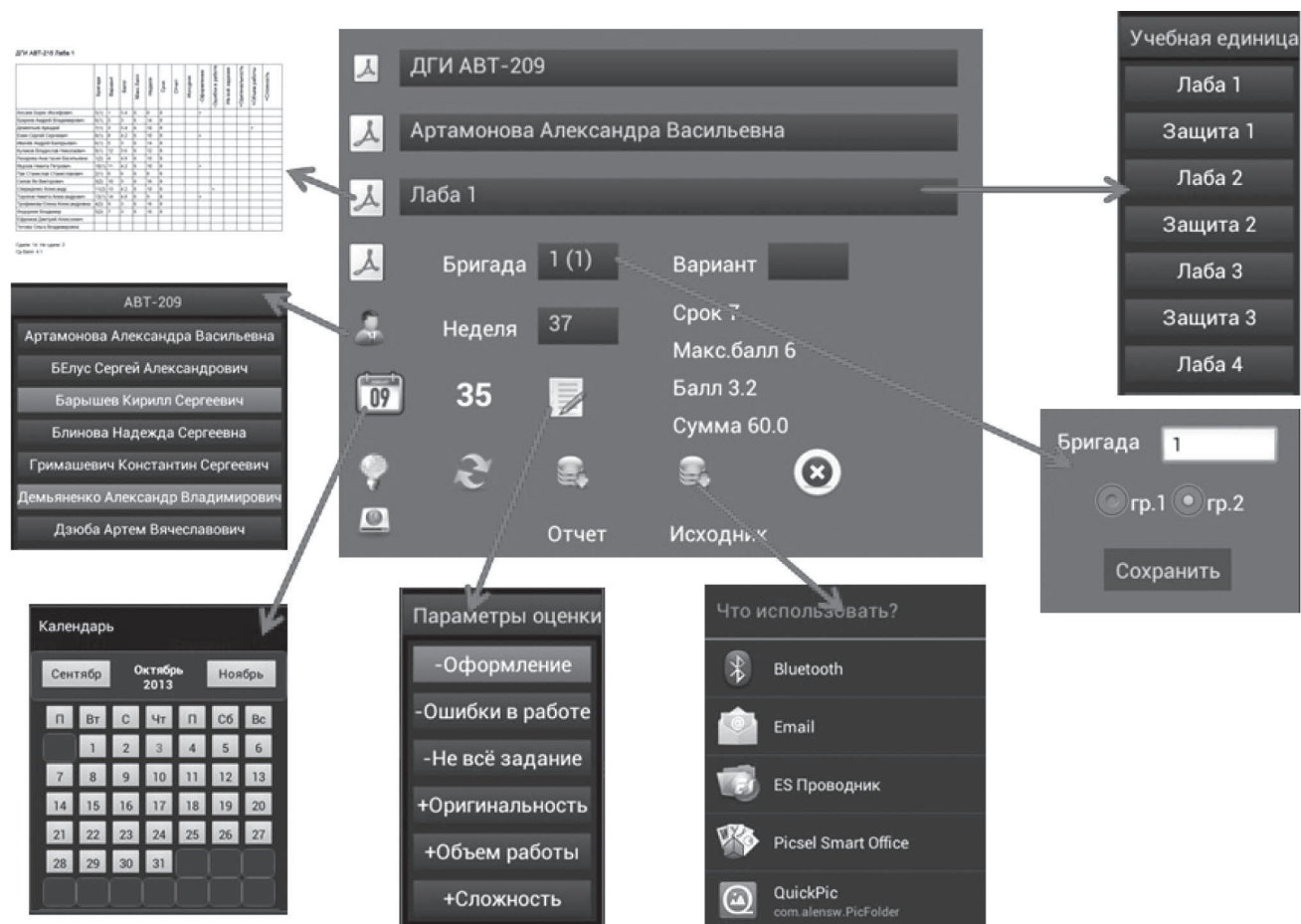


Рис. 4. Android-приложение «Преподаватель»

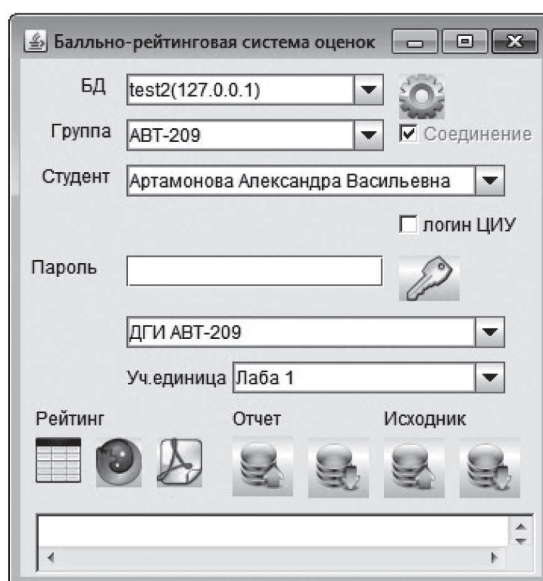


Рис. 5. Клиентское приложение «Студент»

работы с локальными копиями данных рейтингов (рейтинги могут быть скачаны из разных БД). Функционал приложения включает в себя:

- настройку списка входов БД;
- подключение к БД и регистрацию (online);
- выбор рейтинга;
- редактирование данных рейтинга – номер бригад и вариантов заданий, оценки, посещаемость;
- автономную работу (offline) с локальными копиями рейтингов (загрузка, редактирование, синхронизация с источником);
- генерацию отчетов в формате pdf – по учебной единице, студенту, пропускам и успеваемости группы;
- скачивание и просмотр файлов из хранилища.

Клиентское приложение «Студент» предназначено для просмотра данных рейтинга студента, а также для загрузки файлов в хранилище (рис. 5). Система поддерживает два вида авторизации: по собственным логинам/паролям и по логинам/паролям ЦИУ НГТУ, обращаясь к соответствующему серверу.

Перспективы интеграции с информационной системой НГТУ. Основное направление интеграции – интеграция по данным, в ближайшей перспективе включает в себя:

- использование данных ЦИУ для проверки логинов / паролей;
- вычисление и экспорт баллов контрольных недель на основе текущего рейтинга;
- экспорт итогового рейтинга в систему ввода экзаменационных / зачетных оценок;
- импорт списков учебных групп, студентов, дисциплин.

В качестве ресурсной поддержки необходимы хостинг локальных БД MySQL, а также сопровождение и текущий консалтинг. Возможно также свободное распространение программ для автономного использования без централизованного хостинга.

ЛИТЕРАТУРА

1. *Рейтинговая система* [Электронный ресурс] / Новосибирский государственный технический университет, 1994–2013. – Электрон. дан. – Режим доступа: <http://www.nstu.ru/education/rating>, свободный
2. *Информационная система университета* [Электронный ресурс] / Новосибирский государственный технический университет, 1994–2013. – Электрон. дан. – Режим доступа: http://ciu.nstu.ru/isu/isu_about

3. Романов Е.Л. Система с мобильными клиентами для учета рейтинга успеваемости студентов // Сборник трудов VII Международной научно-практической конференции «Информационные и коммуникационные технологии в образовании, науке и производстве», Протвино, 24–28 июня 2013 г. – Протвино: Управление образования и науки администрации, 2013. – С. 332–335.

E.L. Romanov
Novosibirsk State Technical University, Novosibirsk, Russia

AUTOMATION OF STUDENTS' PROGRESS RATING REGISTRATION

Keywords: progress, rating, automation, Android, desktop, Java, database (DB), MySQL.

The set of applications for the automatic registry university system of students' progress rating is considered. The hundred-point and ECTS grading scales are used. The students' progress rating takes into account both subjective estimations and formal control of delivery terms, parameters of quality of tasks performance as well as absence from classes. The traditional typical learning paths are presented. The rating calculating algorithms produce "fair" estimations for these traditional typical learning paths.

The technological, organizational and psychological means inculcations control automation 'from above' and 'from below' are discussed. The set of compatible applications for decentralized system is proposed.

The applications can use varied data sources: MySQL server DBs, web-server with its own APIs, SQLite local DBs and files with particular group-theme rating data. Variety of sources implements two work regimes with DB for teachers:

multiple access in share regime (if DB-hosting or web-hosting is available);

teacher's access to local DB and sharing rating data files for local access and periodic timing of files with DB (if hosting is not available).

All applications have multilayer architecture with shared program Java-code for all server and client applications. The architecture consists of the conception layer, business logic controller, business logic model, tools for extraction and linkage of DB table business objects, binary and XML converters for business objects, support of DB table objects, JDBC and SQLite library interfaces, web-service server components.

The system provides two kinds of authorization: according to its own logins/passwords storing in DB and according to university informatization centre logins/

passwords. Every DB provides repository for temporary document storage and change. Repository is an area of data limited capacity.

The system includes six main components and applications.

The DB server (MySQL) with remote Java-clients' standard network access means (using JDBC libraries).

The web-server with own APIs and supporting all levels of business logic. Client applications work with web-server in "thin client" mode.

Client application "Manager" provides all manager functions except for DB MySQL direct hosting: editing of data source list, connecting to DB, DB initialization, DB backup, editing of rating structure, maintaining of teacher accounts.

Client application "Teacher" provides editing of data source list, connecting to DB and authorization, editing of rating structure, making pdf, html and window application reports, uploading, loading and viewing of repository files, making files – copies of rating data and timing file-copies with DB-source.

Android-application "Teacher" has similar functionality with some limitations.

Teacher's client applications can work with all data sources: in the regime of web-server and server DB multi-access mode and local DB and rating data file-copy off-line editing.

Client applications "Student" is intended for student's rating data viewing and for files uploading into repository.

In conclusion the system integration possibilities with university informatization centre are discussed.

REFERENCES

1. *The rating system* [Electronic resource] / Novosibirsk state technical University, 1994-2013. – The electron. Dan. – Mode of access: <http://www.nstu.ru/education/rating> free
2. *Information system of the University* [Electronic resource] / Novosibirsk state technical University, 1994-2013. – The electron. Dan. – Mode of access: <http://ciu.nstu.ru/isu/about>
3. *Romanov E.L.* System with mobile clients for accounting ranking of students // proceedings of the VII International scientific practical conference «Information and communication technologies in education, science and production», Protvino, June 24-28, 2013 – Protvino: Department of education and science of the administration, 2013. - S. 332-335.