



УДК 004.94

ПРИНЦИПЫ СОЗДАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ СИСТЕМЫ «ГЕРМЕС-КНИТ» БелГУ

В. М. МИХЕЛЕВ
В. Е. ХАЧАТРИАН
Д. В. ПЕТРОВ
К. В. КУЗНЕЦОВ

*Белгородский
государственный
университет*

e-mail: Mikhelev@bsu.edu.ru

e-mail: Khachatryan@bsu.edu.ru

e-mail: Dexis@list.ru

e-mail: Chapaev28@ya.ru

Информационная система (ИС) «Гермес-КНИТ» разработана и внедрена в виде Интернет-портала с использованием современных информационно-сетевых технологий и обеспечивает использование инновационных подходов к ИТ-образованию.

Основная проблема, возникающая при создании современных Интернет-порталов – это временные задержки при одновременной работе большого количества пользователей. Современные технологии позволяют перевести Web приложение на уровень систем реального времени, при этом время получения данных составляет миллисекунды. Поэтому преподаватели и студенты могут проводить учебный процесс в реальном времени.

В основу архитектуры программного обеспечения системы заложены технологии от компаний Google и Sun Microsystems с использованием фреймворка GWT (Google Web Toolkit), позволяющие использовать кросс-платформенную технологию J2EE. GWT упрощает создание высокопроизводительных приложений AJAX.

Ключевые слова: ИТ-образование, распределенные веб-приложения, Java, AJAX, Google web toolkit, информационная система «Гермес-КНИТ», система управления обучением.

Введение

Информационная система (ИС) «Гермес-КНИТ» разработана в виде Интернет-портала с использованием современных информационно-сетевых технологий и обеспечивает использование инновационных подходов к ИТ-образованию [1,4]. Работа выполнена на кафедре «Математического и программного обеспечения информационных систем» факультета «Компьютерные науки и телекоммуникации» БелГУ.

Система предназначена для учета и контроля выполнения лабораторных и курсовых работ студентами и учета посещаемости ими практических занятий. Благодаря системному подходу к организации информации, связанной с ведением учебного процесса, на базе информационной системы возможно построение ряда подсистем: тестирование студента, анкетирование, личный кабинет родителя, сбор семестровой и годовой статистики по дисциплине, студенту, группе или другим объектам, задействованным в бизнес-процессе информационной системы.

Основная проблема, возникающая при создании современных Интернет-порталов – это временные задержки при работе пользователей. Применяемые нами технологии позволили перевести Web приложение на уровень систем реального времени, при этом время на получение данных уменьшено до десятков миллисекунд. Поэтому преподаватель может наблюдать без временных задержек процесс выполнения и сдачи на проверку лабораторных и курсовых работ.

В основу архитектуры программного обеспечения системы заложены технологии от компаний Google и Sun Microsystems с использованием фреймворка GWT (Google Web Toolkit), позволяющие использовать кросс-платформенную технологию J2EE. GWT упрощает создание высокопроизводительных приложений AJAX. Интерфейс создается на языке программирования Java, а GWT компилирует исходный код в оптимизированный JavaScript [5].



Анализ информационных систем обеспечения учебного процесса

Рынок программных продуктов обеспечения и проведения учебного процесса в высших учебных заведениях (ВУЗ) насыщен различными решениями. Можно выделить тот факт, что в последнее время почти у каждого университета, использующего инновационные подходы к ИТ-образованию, существуют системы «электронный университет» или дистанционного образования, которые направлены на информатизацию определенных процессов возникающих в ВУЗе, и связанных с учебным процессом. Проанализировав рынок программного обеспечения (ПО), можно выделить следующие типы решений основанные:

- 1) на разработке собственной информационной системы;
- 2) на системе управления обучением (LMS);
- 3) на коммерческих LMS и подобных системах.

Чаще всего используется готовое решение или конструктор для реализации решения под определенные требования, таковыми являются системы управления обучением (LMS). Большинство из них поддерживают стандарт Sharable Content Object Reference Model (SCORM) — стандарт, разработанный для систем дистанционного обучения, основанный на XML и содержащий требования к организации учебного материала и всей системы дистанционного обучения.

Обоснование выбора информационной технологии

В основу архитектуры программного обеспечения информационной системы «Гермес-КНИТ» заложены технологии от компаний Google и Sun Microsystems с использованием фреймворка GWT (Google Web Toolkit), позволяющие использовать кросс-платформенную технологию J2EE. GWT упрощает создание высокопроизводительных приложений AJAX. Google Web Toolkit (GWT) — это транслятор языка Java в язык JavaScript, разработчик пишет код для клиентской и серверной части с использованием одного языка. При этом клиентская часть поддерживает все стандарты спецификации Web 2.0. Асинхронный запрос к серверу, предполагает, что клиент получает не контент страницы с гипертекстовой разметкой, а только необходимые данные, на основе которых страница изменяет свой вид. Благодаря этому можно выделить уровень бизнес-логики не только для сервера, но и для клиентского уровня. Последнее означает, что выполнение JavaScript компонента определяется данными пришедшими с серверной части.

Благодаря использованию полноценной поддержки объектно-ориентированного программирования языком Java, применению модульного тестирования, и использованию готовых модулей и компонентов, время разработки, трудоемкость и экономическая эффективность проекта снизились в разы.

Для разработки программ с использованием выбранных технологий достаточно иметь наборы разработчика JSDK и GWT, в состав которых входят необходимые программы для трансляции, компиляции и сборки приложений. Обычно в составе таких наборов нет специальных редакторов, которые направлены на упрощение и увеличение скорости разработки. Данная задача решается сторонними программными средствами IDE. Это программное средство, направленное на упрощение процесса написания кода и разработки программ. Все они имеют функциональные средства по автоматизации часто повторяемых разработчиком действий, автоматическое завершение кода, и многое другое. Выбор остановлен на Eclipse. Это среда бесплатная и имеет несколько наборов модулей для Java, для Web и J2EE, для C++.

Проектирование информационной составляющей

Основой любой ИС являются данные, которые принимаются, обрабатываются, хранятся, отображаются. Если рассматривать ИС со стороны манипуляции с данными, то можно считать, что работа с данными это основная задача ИС.



В ИС «Гермес-КНИТ» информационной составляющей являются данные, хранимые в базе данных, а также данные в виде файлов, хранимые в файловой системе.

Для организации хранения пользователей работающих в системе в данный момент, используется динамически обновляемая таблица «online cache».

Из БД на рис. 1 можно выделить сущности, относящиеся к пользователям – студент, преподаватель, студент, администратор. Все пользователи объединены в группы, и права у пользователя в ИС зависят от его группы. В плане учебного процесса можно отметить сущности, относящиеся к курсовым работам, к отчетам по лабораторным работам и к текстам лабораторных работ.

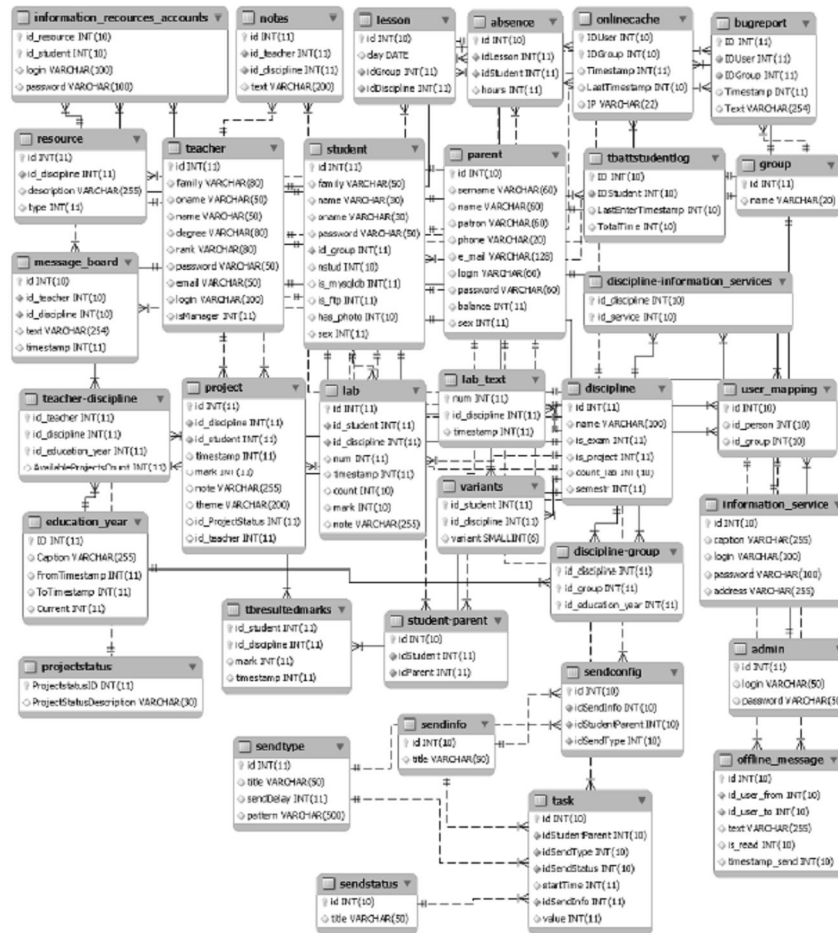


Рис. 1. Физическая модель базы данных

Проектирование программной составляющей

Многоуровневые приложения обработки данных – это приложения с использованием элементов данных, разделенные на несколько логических слоев (или уровней). Другими словами – это приложение, разделенное на несколько проектов: уровень доступа к данным, уровень бизнес-логики и уровень представления имеют собственные

проекты. n-уровневые приложения данных представляют собой приложения, разделенные на несколько уровней. Они называются "распределенными приложениями" или "многоуровневыми приложениями". n-уровневые приложения разделяют обработку на дискретные уровни, распределенные между клиентом и сервером. При разработке приложений, обращающихся к данным, необходимо иметь четкое разделение между различными уровнями, которые составляют приложение.

Типичное n-уровневое приложение включает уровень представления, средний уровень и уровень данных. Самым простым способом разделения различных уровней является создание дискретных проектов для каждого уровня, который требуется включить в приложение. Разделение компонентов приложения на разные уровни увеличивает эксплуатационную надежность и масштабируемость приложения. Это осуществляется за счет возможности применения новых технологий к какому-либо одному уровню без необходимости изменять все решение в целом. Кроме того, n-уровневые приложения обычно хранят важные сведения на среднем уровне, который изолирован от уровня представления.

При проектировании архитектуры приложения выделяются несколько уровней:

- 1) На клиентской стороне:
 - уровень отображения;
 - уровень клиентской бизнес-логики;
 - уровень доступа к сервисам.
- 2) На серверной стороне:
 - уровень предоставления сервисов;
 - уровень бизнес-логики приложения и доступа к данным;
 - уровень данных.

На рис. 2 представлена схема взаимодействия компонентов уровня представления. В основе уровня лежит шаблон проектирования «модель-представление-контроллер». Он позволяет разделить данные, представление и обработку действий пользователя на три отдельных компонента [3,4].

Уровень доступа к сервисам позволяет получать данные со среднего уровня или, другими словами, с серверной стороны. Компоненты данного уровня должны отвечать не только за получение данных, но и за отправку и управление соединением. На базе данного уровня, можно организовывать асинхронные запросы к серверу.

Средний уровень является слоем, который осуществляет связь между уровнями представления и данных. Он включает в себя следующие компоненты:

- бизнес-логику, например, проверку данных и бизнес-правила;
- компоненты и логика доступа к данным и организации транзакций;
- объектные представления данных, например, классы сущностей Hibernate.



Рис. 2. Организация взаимодействия компонентов уровня представления

Общие службы приложений, такие как службы аутентификации, авторизации и персонализации, а также службы, через которые организуется доступ к бизнес-логике среднего уровня и к данным (рис. 3).

Средний уровень обычно подсоединяется к уровню данных с использованием подключения данных. Это подключение данных храниться в компоненте доступа к данным. Уровень данных, в общем случае, является сервером управления базами данных, который хранит данные приложения и организует взаимодействие с данными, хранящимися в базе данных. База данных – организация файлов в файловой системе, направленная на хранение и оперативное управление данными.

Разработка ядра системы

Под ядром системы понимается набор классов и модулей среднего уровня приложения. Это компоненты уровня служб и сервисов, бизнес-логики, а также уровня доступа к данным. На рис. 4 представлена UML диаграмма зависимостей и наследования классов серверной стороны.



Рис. 3. Организация взаимодействия компонентов уровня представления

Серверная сторона (server side) предоставляет интерфейс StudentService доступа, по протоколу GWT-RPC (Google Web Toolkit Remote Procedure Call), к сервисам. Протокол GWT-RPC обеспечивает передачу сериализованных данных поверх протокола HTTP и удаленный вызов процедур. В качестве сериализованных данных могут выступать POJO объекты или объекты классов, реализующие шаблоны проектирования объект-значение (Value object) и объект передачи данных (Data transfer object). Благодаря этому на программном уровне, созданный объект на серверной стороне, может использоваться на клиентской стороне.

Разработка модулей доступа к данным

Доступ к данным осуществляется с использованием библиотеки Hibernate, которая предоставляет программисту большие возможности по работе с различными СУБД. Данная библиотека предоставляет возможность отображения объектов на реляционные сущности (Object Relation Mapping). Это позволяет разработчику писать

минимальное количество SQL кода и оперировать объектами отображения (Domain entity), с помощью которых реализуется CRUD (create read update delete) функциональность манипуляции данными.

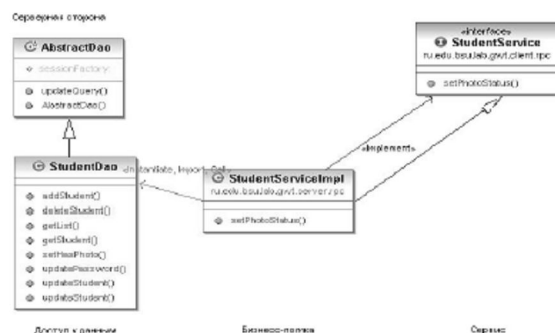


Рис. 4. Архитектура модулей серверной стороны

В качестве источника данных используется сервер управления базами данных MySQL, а доступ к нему осуществляется с помощью MySQL JDBC драйвера.

К ядру системы можно отнести класс `AbstractDao`, а также унаследованные от него уровни доступа к данным, обеспечивающие создание, манипуляцию и удаление данных. Все операции с данными происходят в транзакциях. Этот класс в контексте сессии к базе данных управляет открытием транзакции, ее завершением и откатом. Классы, унаследованные от `AbstractDao`, обеспечивают работу с определенными сущностями в БД. Так, например `StudentDao` обеспечивает доступ к данным студентов, а `Discipline Dao` к данным, которые связаны с дисциплинами.

В результате разработки модулей доступа к данным разработчику ИС предоставлен полный набор функциональных возможностей по работе с сущностями, созданными на этапе проектирования информационно-логической составляющей, хранящейся в базе данных.

Разработка модулей бизнес-логики приложения

Отличие модулей бизнес-логики от модулей доступа к данным заключается в том, что в обязанности методов модулей данных входят только операции создания, выборки, изменения и добавления данных. Модули бизнес-логики применяют множество этих методов для организации более сложной функциональности с участием данных, полученных с нижнего уровня – уровня доступа к данным.

К ядру системы относятся и классы сервисного уровня, реализующие сервисные интерфейсы. Эти классы играют роль плюсов, которые принимают запросы от клиентов с входными данными и вызывают соответствующие методы классов уровня бизнес-логики.

На среднем уровне ИС классы, содержащие бизнес-логику приложений, одновременно являются сервисными классами. Они унаследованы от класса `RemoteHttpServlet` и реализуют интерфейс доступа клиентов на серверную часть.

В результате разработки модулей бизнес-логики приложения, серверная сторона получила полный набор классов, которые обеспечивают основную функциональность ИС и предоставляют клиентской стороне необходимые данные в требуемом виде через GWT-RPC интерфейсы доступа к серверной стороне.



Разработка Web интерфейсов пользователей

Разработка интерфейсов пользователя произведена с использованием технологии GWT. Данная технология позволяет разрабатывать Web приложение на одном языке программирования – Java, как серверную сторону, так и клиентскую. Возможность трансляции Java кода написанного программистом в JavaScript код, исполняемый в браузере, дает разработчику возможность использовать готовые компоненты пользовательского интерфейса. В качестве библиотеки компонентов пользовательского интерфейса была использована Java Script библиотека компонентов Ext-JS. Данная библиотека предоставляет полный набор компонентов для реализации интерфейсов с любым уровнем сложности.

На рис. 5 представлена UML диаграмма классов клиентской стороны, на которой представлены классы, отображающие отношения между классами клиентской стороны. Классы бизнес-логики уровня клиентского приложения получают доступ к сервисам сервера через шаблон "фабрика", который реализован в классе RPCFactory. Данный класс содержит весь набор сервисов доступных на серверной стороне. Отображение данных на клиентской стороне происходит с помощью специальных компонентов – виджетов.

Композиция таких компонентов позволяет строить сложные пользовательские интерфейсы, такие как таблицы с возможностями как локальной, так и удаленной сортировки. Многие компоненты, ориентированы на данные и их отображение. Имеют в своем составе место хранения локальных данных – Store, которое позволяет организовывать ряд локальных операций с данными, таких как сортировка, фильтрация, поиск, удаление записей добавление, редактирование.

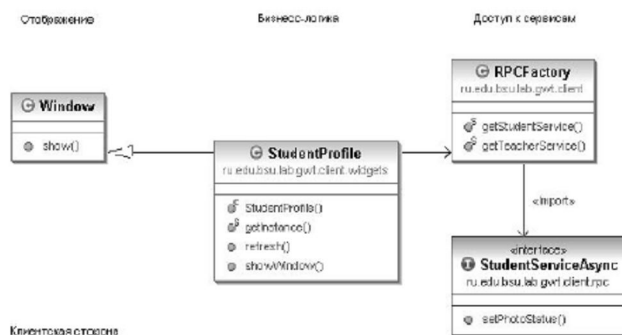


Рис. 5. Архитектура модулей клиентской стороны

На основе такого механизма реализуется возможность кэширования, когда компонент принимает решение об обновлении отображения на основе локальных данных или запроса данных для обновления с серверной стороны. В ИС используются такие компоненты как деревья, которые позволяют реализовывать отображение иерархий и отношений объектов один ко многим. Компоненты деревьев активно используются для отображения основных сущностей базы данных, для отображения иерархий «учебный год», «дисциплина», «группа», «студент», «данные студента» для пользовательского интерфейса преподавателя. Все компоненты аналогичны по отображению системным компонентам, что позволяет пользователю быстро освоиться в интерфейсе.

Пользовательский интерфейс разделяется на три группы (рис. 6). В зависимости от группы пользователю доступны определенные функциональные возможности интерфейса.

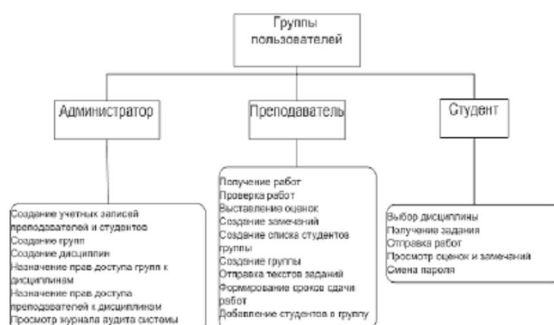


Рис. 6. Группы пользовательских интерфейсов

В результате разработки различных уровней приложения информационная система имеет архитектуру, представленную на рис. 7.

В архитектуре ИС можно выделить три основных приложения, которые обеспечивают функционирование приложения:

- в роли web-сервера выступает приложение Tomcat, оно является J2EE контейнером и позволяет запускать Web приложения, разработанные с использованием Java технологий;

Ftp сервер – специально разработанное в рамках ИС «Гермес-КНИТ» приложение позволяющее производить авторизацию пользователей системы и реализовывать дополнительную бизнес логику совместно с web приложением запущенным в контексте J2EE контейнера;

- в роли системы управления базами данных выступает СУБД MySQL.

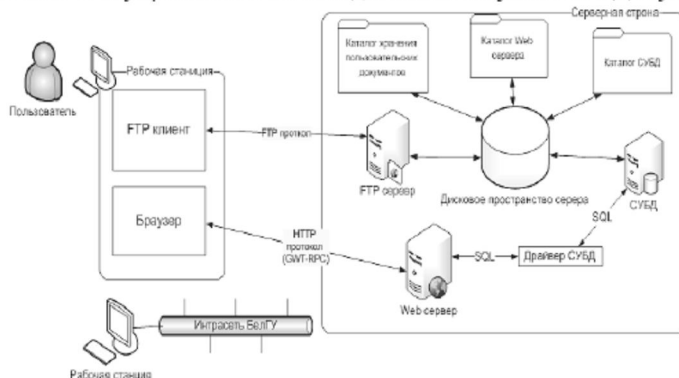


Рис. 7. Архитектура разработанной информационной системы «Гермес-КНИТ»

Перечисленные приложения могут быть запущены физически как на одном узле локальной вычислительной сети (ЛВС), так и быть распределенными между узлами ЛВС.

Информационная система была успешно внедрена в учебный процесс и используется на кафедре «Математического и программного обеспечения информационных систем» факультете компьютерных наук и телекоммуникаций НИУ Белгородского государственного университета.

Используемые в системе программы прошли гос. регистрацию [2].



Литература

1. Сухомлин В.А. ИТ-образование: концепция, образовательные стандарты, процесс стандартизации. М.: Горячая линия-Телеком, 2005.
2. Свидетельство о гос. регистрации программы для ЭВМ № 2010612501. Информационная система Гермес-КНИТ БелГУ / Журавский В.С., Михелёв В.М., Хачатрян В.Е., Михелёв М.В., (RU); заявитель и патентообладатель Белгородский гос. унив. – № – № 2009616819; заявл. 01.12.09; опубл. 12.04.10.
3. Михелев В.М., Журавский В.С., Хачатрян В.Е. Информационная технология системного анализа профессиональных компетенций в рамках освоения учебной дисциплины. Компьютерные науки и технологии: сб. материалов междунар. науч. конф. / БелГУ – Белгород, 2009. – Т I. – С. 226 – 229.
4. В.М. Михелёв, В.Е. Хачатрян. Принципы создания Web-портала для оценки профессиональной компетенции студентов. IV Международная научно-практическая конференция "Современные информационные технологии и ИТ-образование" 14-16 декабря 2009 г. Москва, МГУ им. М. В. Ломоносова, С. 228-232.
5. Cooper, R. T. GWT in Practice. [Текст] В 2 ч А New Kind of Client / R. Cooper. – Manning Publications Co., 2008. – 377 с.: ил.

**PRINCIPLES OF CREATION OF THE SOFTWARE OF INFORMATION SYSTEM
«HERMES-KNIT» BSC**

**V. M. MIKHELEV
V. E. KHACHATRYAN
D. V. PETROV
K. V. KUZNETSOV**

Belgorod State University

e-mail: Mikhelev@bsu.edu.ru

e-mail: Khachatryan@bsu.edu.ru

e-mail: Dexis@list.ru

e-mail: Chapaev28@ya.ru

Information system (IS) "Hermes-Knit" has developed and implemented in the form of an internet portal using advanced information and network technologies and enables the use of innovative approaches to IT education

The main problem that occurs when you create the modern Internet portals – a temporary delay in the work of a large number of simultaneous users. Modern technology allows you to make Web application on the level of real-time systems, and the time of receipt of the data is a millisecond. Therefore, teachers and students can conduct the teaching process in real time.

The basis of the software architecture of the system laid down technology from Google and Sun Microsystems with the framework GWT (Google Web Toolkit), allow the use of cross-platform technology to J2EE. GWT makes it easy to create high-performance application AJAX.

Key words: IT education, distributed web applications, Java, AJAX, Google web toolkit, information system "Hermes-Knit", learning management system.