

**САНКТ-ПЕТЕРБУРГСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ
им. проф. М. А. БОНЧ-БРУЕВИЧА**

Факультет СС, СК и ВТ

Дипломная работа

на тему

**Разработка системы для дистанционного обучения
студентов СПбГУТ им. проф. М.А. Бонч-Бруевича**

Дипломник **Бычковский К. М.**

Руководитель работы **Доронин Е. М.**

Санкт-Петербург

2011 г.

РЕФЕРАТ

В дипломной работе рассматривается тема «Разработка системы для дистанционного обучения студентов СПбГУТ им. проф. М.А. Бонч-Бруевича.

Дипломная работа содержит 76 страниц, 46 рисунков, 4 таблицы.

Ключевые слова: система дистанционного обучения (СДО), дистанционные технологии, Интернет, обучение студентов, серверное программирование, html, php, технология AJAX, безопасность web-приложений.

Цель работы – исследование возможности внедрения современных технологий Интернет в процесс обучения студентов высших образовательных учреждений, создание платформы для организации учебного процесса в СПбГУТ им. проф. М.А. Бонч-Бруевича.

В результате дипломного проектирования была разработана система дистанционного обучения на основе сайта в сети Интернет, которую впоследствии можно использовать для организации образовательных процессов в высших учебных заведениях.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
1. НЕОБХОДИМОСТЬ ВНЕДРЕНИЯ ДИСТАНЦИОННОГО ОБУЧЕНИЯ В ВЫСШИХ УЧЕБНЫХ ЗАВЕДЕНИЯХ. ПОСТАНОВКА ЗАДАЧИ	9
1.1. АНАЛИЗ СОВРЕМЕННОГО ПРОЦЕССА ОБУЧЕНИЯ	9
1.2. ОБОСНОВАНИЕ НЕОБХОДИМОСТИ ВНЕДРЕНИЯ ТЕХНОЛОГИЙ ДИСТАНЦИОННОГО ОБУЧЕНИЯ В УЧЕБНЫЙ ПРОЦЕСС ВЫСШЕГО ОБРАЗОВАНИЯ.....	12
1.3. ПОСТАНОВКА ЗАДАЧИ.....	16
2. МОДЕЛИРОВАНИЕ СИСТЕМЫ ДИСТАНЦИОННОГО ОБУЧЕНИЯ.....	19
2.1 ОБЩЕЕ ТЕХНИЧЕСКОЕ ЗАДАНИЕ НА СИСТЕМУ	19
2.1.1 ОБЩИЕ ТРЕБОВАНИЯ.....	19
2.1.2 ТРЕБОВАНИЯ К ПОЛЬЗОВАТЕЛЯМ СИСТЕМЫ.....	20
2.1.3 ТРЕБОВАНИЯ К ОРГАНИЗАЦИИ МЕХАНИЗМА ВЗАИМОДЕЙСТВИЯ ПОЛЬЗОВАТЕЛЕЙ.....	20
2.1.4 ТРЕБОВАНИЯ К МОДУЛЮ НАКОПЛЕНИЯ ДАННЫХ.....	20
2.1.5 ТРЕБОВАНИЯ К ПРОВЕДЕНИЮ ОБРАЗОВАТЕЛЬНЫХ ПРОЦЕССОВ В СИСТЕМЕ.....	21
2.1.6 ТРЕБОВАНИЕ К ИНТЕРФЕЙСАМ ПОЛЬЗОВАТЕЛЕЙ.....	21
2.1.7 ПРИМЕНЕНИЕ СТРОРОННЕГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ.....	21
2.1.8 ТРЕБОВАНИЯ К ВОЗМОЖНОСТЯМ МОДЕРНИЗАЦИИ	22
2.1.9 ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ К РЕАЛИЗАЦИИ СИСТЕМЫ	22
2.2 ОБЗОР ПЛАТФОРМЫ ДЛЯ СОЗДАНИЯ СИСТЕМЫ ДИСТАНЦИОННОГО ОБУЧЕНИЯ.....	23
2.3 РАЗРАБОТКА МОДЕЛИ СИСТЕМЫ ДИСТАНЦИОННОГО ОБУЧЕНИЯ.....	26
2.3.1 ФЕНОМЕНОЛОГИЯ СИСТЕМ ДИСТАНЦИОННОГО ОБУЧЕНИЯ.....	26
2.3.2 ФУНКЦИИ ПОЛЬЗОВАТЕЛЕЙ СИСТЕМЫ.....	28
2.3.3 УЧЕБНЫЙ ПРОЦЕСС.....	29
2.3.4. ДЕКОМПОЗИЦИЯ МОДЕЛИ ОБУЧЕНИЯ.....	31
3. РАЗРАБОТКА СИСТЕМЫ ДИСТАНЦИОННОГО ОБУЧЕНИЯ	35
3.1 РАЗРАБОТКА ИНТЕРФЕЙСА СИСТЕМЫ.....	35
3.2 РАЗРАБОТКА БАЗЫ ДАННЫХ СИСТЕМЫ.....	38
3.3 РАЗРБОТКА ГЛАВНОГО ПРОГРАММНОГО МОДУЛЯ СИСТЕМЫ.....	46
3.3.1 КЛАСС ИНТЕРФЕЙСА СИСТЕМЫ.....	46
3.3.2 ПОЛИТИКА БЕЗОПАСНОСТИ СИСТЕМЫ. ОРГАНИЗАЦИЯ СОЕДИНЕНИЯ С БАЗОЙ ДАННЫХ	48
3.3.3 ФУНКЦИОНАЛ ПОЛЬЗОВАТЕЛЯ	49
3.3.4 ФУНКЦИОНАЛ АДМИНИСТРАТОРА	52
3.3.5 ФУНКЦИОНАЛ ПРЕПОДАВАТЕЛЯ.....	58
3.3.5.1 СОЗДАНИЕ КУРСА.....	58
3.3.5.2 АДМИНИСТРИРОВАНИЕ КУРСОВ	66
3.3.6 ФУНКЦИОНАЛ СТУДЕНТА	67
3.4 ОРГАНИЗАЦИЯ БЕЗОПАСНОСТИ В СИСТЕМЕ ДИСТАНЦИОННОГО ОБУЧЕНИЯ.....	71
ЗАКЛЮЧЕНИЕ.....	74
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	75

ВВЕДЕНИЕ

Из основных тенденций развития современного образования следует отметить тот факт, что из-за большого объема учебных программ студентам высших образовательных учреждений для успешного освоения большинства дисциплин требуется изучать часть учебного материала самостоятельно. В случае с очной формой обучения самостоятельная подготовка в настоящее время занимает не менее 45% от всего учебного процесса. Причем объем самостоятельной подготовки растет с каждым годом.

Для самостоятельного изучения учебных дисциплин студенты используют различные ресурсы: печатные издания ВУЗов, материалы, взятые из библиотек или купленные в книжных магазинах, ресурсы сети Интернет. При этом с развитием современной науки у студента, использующего для самостоятельной подготовки печатные ресурсы (книги, учебники, методические пособия), всегда возникает проблема актуальности информации в выбранном материале. К тому же получение нужного материала из книг и пособий требует довольно больших временных затрат, а в современном процессе обучения эта проблема является достаточно критичной.

Самый большой информационный обмен, как известно, происходит сегодня в сети Интернет. Здесь можно быстро и легко получить практически любую информацию по различным темам. Этим, в свою очередь, и пользуется большинство учащихся по всему миру. На заседании коллегии Министерства связи и массовых коммуникаций, 11 мая 2010 года, заместитель руководителя «Роспечати» Андрей Романченко заявил, что Интернетом в России пользуются более 43 миллионов человек. Из них 37% пользователей являются молодые люди в возрасте от 18 до 25 лет. Очевидно, что Интернет является самым востребованным источником информации для российской молодежи.

К сожалению, при получении информации из глобальной сети возникают некоторые проблемы. Сеть Интернет может содержать ресурсы, компетентность которых может вызывать сомнения. Информация может быть неточной, а иногда даже совсем неверной или непригодной для изучения. Поэтому при изучении

требуется осуществлять тщательный отбор информации и использовать проверенные ресурсы Сети.

Пользователи Интернет имеют свободный доступ к наиболее популярным системам сбора информации. Примером такой системы служит «Wikipedia». Создатели позиционируют свой проект как «Свободная энциклопедия». Это означает, что любой посетитель сети Интернет может пополнять содержание сайта. На сегодняшний день пользователями сайта уже создано более 532 000 статей на русском языке. Такой подход имеет ряд недостатков. По словам Ларри Сенджера, одного из создателей «Wikipedia», руководители проекта не задумываются о достоверности публикуемой информации. Компетентность авторов статей не проверяется, и правильность полученной информации можно поставить под сомнение. В свою очередь, Ларри Сенджер собирается создать новое интернет-сообщество под названием «Citizendium». Этот проект будет являться «зеркалом Wikipedia», что допускается условиями ее лицензии использования, но впоследствии авторы будут редактировать статьи в соответствии с правилами новой системы. Результатом должно стать появление массива более качественной информации, и его будут использовать вместо сведений, получаемых из «Wikipedia». К сожалению, на данный момент сайт «citizendium.org» работает в режиме бета-тестирования и только на английском языке. По этой причине пользоваться этим ресурсом для осуществления самостоятельной подготовки сложно.

На основе «wikipedia.org» создано множество сайтов, посвященных более узким тематикам. Осуществить поиск в системе проектов, созданных по технологии «Wiki» можно на сайте «wiki.com». Но, как уже было отмечено выше, доверять информации, полученной с помощью этих ресурсов, не стоит. Понимая проблематичность поиска информации в Интернет, преподавательский состав учебного заведения может создавать собственные ресурсы и публикации в Сети. Большинство образовательных учреждений уже имеют свои сайты, на которых публикуется информация для самостоятельной подготовки студентов. Но эта информация не достаточно систематизирована для качественного

обучения. Для более качественной организации обучения и самостоятельного освоения учебного материала создаются, так называемые, «системы дистанционного обучения». Вариантов исполнения таких систем огромное множество, так как для разных задач требуется разный подход к обучению.

В использовании технологий дистанционного обучения заинтересованы не только образовательные учреждения. Большое количество коммерческих компаний по всему миру, занимающихся разработкой различного оборудования, выгодно отличают возможность обучения своих сотрудников и клиентов по всему миру посредством Интернет. Одной из таких компаний является американская компания «Cisco» — мировой лидер в области сетевых технологий. Так как компания «Cisco» занимается непосредственно созданием сетевого оборудования, используемого инженерами связи по всему миру, то обучение сетевым технологиям компании выгодно управлять дистанционно. Для этого организована Сетевая Академия CISCO (CNA). Студентам академии выдаются логин и пароль, с помощью которого они получают доступ к информационным ресурсам (лекциям), выполняют лабораторные работы и сдают экзамены. Обучение состоит из четырех ступеней. После успешного завершения обучения студентам выдается сертификат, подтверждающий квалификацию студента. Система дистанционного обучения CNA разработана самой компанией CISCO и организована через веб-интерфейс, что делает ее доступной для всех пользователей сети Интернет, желающих стать специалистом в области сетевой инженерии.

Существуют также готовые решения для организации дистанционного обучения. К таким системам можно отнести проект «Moodle». Он распространяется по бесплатной лицензии. К плюсам «Moodle» можно отнести то, что процесс обучения в этой системе, как и в «Сетевой Академии CISCO», организован через веб-интерфейс, что позволяет использовать систему большому количеству пользователей. Эта система создана инициативной группой программистов, которая пыталась объединить потребности всех образовательных учреждений, желающих проводить обучение дистанционно.

Именно поэтому проект при первом знакомстве может показаться немного сложным и непонятным. Существует так же множество модулей, которые могут и не понадобиться в процессе работы системы.

Преимущество системы дистанционного обучения, созданной на основе протокола «http» очевидно. Но в некоторых случаях требуется непосредственный контакт преподавателя и студента. При проведении краткосрочных образовательных курсов требуется проводить занятия в форме семинара, когда после доклада преподавателем учебного материала слушатели могут проконсультироваться и обсудить тему семинара. В сети Интернет такие семинары носят название «вебинары». Компания «Softline» – лидирующая компания в сегменте рынка информационных технологий и продажи программного обеспечения, использует «вебинары» для ознакомления со своими продуктами. Участники «вебинаров» могут ознакомиться с новейшими технологиями в режиме конференции. Это создает большое конкурентное преимущество компании, использующей такой вид дистанционного обучения. Обеспечение учебного процесса с использованием онлайн-конференций требует большой работы программистов, а так же постоянной поддержки проекта. Существует готовое решение, позволяющее обеспечить качественное обучение через сеть Интернет – Adobe Acrobat Connect, позволяющее проводить веб-конференцию, на которой пользователи помимо визуального доклада могут просматривать дополнительные материалы (презентации) онлайн. Организовать дистанционное обучение возможно путем запуска сервера Adobe Connect, на котором преподавательский состав размещает учебные материалы. Студенты же получают возможность посредством специального программного обеспечения иметь доступ к виртуальному классу, просматривают лекции, общаются друг с другом, задают вопросы преподавателю. Adobe Connect Pro предоставляет широкие возможности для эффективных коммуникаций и коллективной работы в режиме онлайн: инструменты для создания и управления контентом, управление программами тренинга и тестирования. Программное решение отличается доступностью и простотой использования: для работы клиентской части

необходим только браузер и Flash Player, что предоставляет пользователям уникальную гибкость в выборе платформы – Windows, MacOS, Linux, Solaris.

К недостаткам Adobe Connect Pro можно отнести большую стоимость программного обеспечения (73 774 рублей для каждого пользователя), что для высшего учебного заведения, где работают сотни преподавателей, является проблемой.

Анализируя возможности всех систем обучения, можно сформулировать основные проблемы при проектировании системы дистанционного обучения студентов высшего учебного заведения:

- выбор платформы для создания системы;
- размещение учебного материала;
- контроль учебного процесса;
- организация общения и консультаций между студентами и преподавателями;
- администрирование ресурса;
- возможность расширения и модернизации системы.

Решению всех этих актуальных проблем, а так же непосредственной разработке системы дистанционного обучения и посвящена дипломная работа.

1. НЕОБХОДИМОСТЬ ВНЕДРЕНИЯ ДИСТАНЦИОННОГО ОБУЧЕНИЯ В ВЫСШИХ УЧЕБНЫХ ЗАВЕДЕНИЯХ.

ПОСТАНОВКА ЗАДАЧИ

1.1. АНАЛИЗ СОВРЕМЕННОГО ПРОЦЕССА ОБУЧЕНИЯ

Для создания системы дистанционного обучения требуется понимание современных процессов обучения. Государственные образовательные учреждения используют для организации обучения закон Российской Федерации «Об образовании». Процесс обучения можно разделить на три части: получение теоретических знаний, получение практических навыков, контроль полученных знаний и навыков. Обучение включает аудиторные занятия (лекции, семинары, упражнения, практические и лабораторные занятия) и самостоятельную подготовку. Контроль полученных знаний и умений осуществляется путем сдачи зачетов и экзаменов. В качестве промежуточного контроля проводится тестирование.

Важным этапом в развитии образования является оптимизация времени, отводимого на самостоятельную подготовку. Современные студенты во время подготовки к занятиям все меньше времени проводят в библиотеках и читальных залах учебного заведения, где бы они имели возможность общения с преподавателями. Все чаще учащиеся используют Интернет для получения информации по различным темам. Задачей разработчика системы дистанционного обучения является совмещение традиционных способов обучения с современными средствами получения информации. Этим можно облегчить работу, как студентов, так и преподавателей. Но для этого необходимо разобраться с основами современного образования.

Для начала рассмотрим структурное деление. Во всем мире высшие учебные заведения состоят из факультетов (департаментов). Факультет – это подразделение, осуществляющее подготовку студентов и аспирантов по одной или нескольким родственным специальностям, повышение квалификации специалистов, а также руководство научно-исследовательской деятельностью кафедр, которые он объединяет. Факультеты делятся на кафедры,

осуществляющие подготовку в рамках определённой специализации. В зависимости от изучаемого курса студенты объединяются в группы, за которые отвечают кафедры факультета. Так же существуют подразделения, отвечающие за управление обучением и за контроль учебного процесса (администрация, учебные отделы). Имеются вспомогательные и обслуживающие подразделения. Обобщенная структура образовательного учреждения представлена на рис. 1.1.

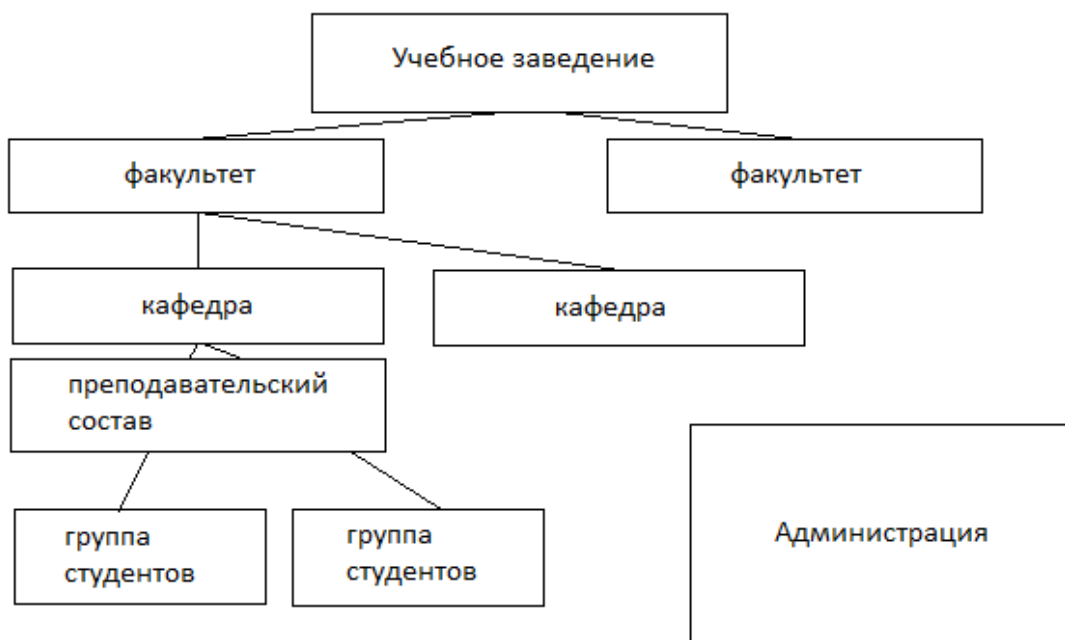


Рис. 1.1. Структурно-административное деление учебного заведения

Из рисунка становится ясно, как именно связаны подразделения университета. Очевидно, что при внедрении системы дистанционного обучения нужно учитывать административное деление.

Для правильного моделирования системы следует учитывать процесс организации обучения. Содержание курсов, которые требуется изучить студенту для того, чтобы получить определенную специальность, строго регламентированы государственными стандартами. Существует множество учебников и учебных пособий, используемых для изучения различных дисциплин, которые одобрены министерством образования. Для хранения и

предоставления их студентам в каждом учебном заведении организованны библиотеки. Библиотека – это постоянно пополняемый ресурс. Для хранения печатных материалов требуются довольно большие помещения. Ведь библиотеки больших университетов содержат десятки тысяч единиц учебной и научной литературы. Многие ВУЗы создают электронные версии библиотек, где материалы представлены в виде цифровых копий. Это очень удачная идея, позволяющая облегчить доступ к учебным ресурсам. К тому же практику использования электронных библиотек при традиционном обучении можно использовать и для дистанционного обучения.

При анализе современного образования требуется рассмотреть процесс обучения студентов и формирования учебных курсов. Учебные курсы создаются преподавателями учебного заведения на основе учебных планов и программ. Курс, как уже было упомянуто выше, состоит из лекций, семинаров, практических занятий и лабораторных работ. Изучение дисциплины может считаться завершенным только после сдачи студентом экзамена или зачета. Почти все материалы, необходимые для успешного обучения, разрабатываются преподавателями кафедр.

Последнее, что нужно рассмотреть в данном вопросе, – это формы обучения. В высших учебных заведениях практикуют использование четырех форм обучения: очную, заочную, вечернюю и экстернат. Разработчиков системы дистанционного обучения в первую очередь интересует отношение часов аудиторных занятий к часам самостоятельной подготовки студентов при той или иной форме обучения. Очевидно, что применение современных телекоммуникационных технологий в большей мере необходимо использовать для заочной и вечерней формы обучения и при экстернате.

1.2. ОБОСНОВАНИЕ НЕОБХОДИМОСТИ ВНЕДРЕНИЯ ТЕХНОЛОГИЙ ДИСТАНЦИОННОГО ОБУЧЕНИЯ В УЧЕБНЫЙ ПРОЦЕСС ВЫСШЕГО ОБРАЗОВАНИЯ

В последние десятилетия появилась новая проблема развития образования в России. Знания стареют каждые три, пять лет, а технологические знания – каждые два, три года. Объем знаний выпускников высших учебных заведений удваивается каждые три, четыре года. Если не менять образовательных технологий, то качество подготовки специалистов будет объективно отставать от требований рынка труда. Усвоение знаний студентами с помощью информационных и коммуникационных технологий по самым нижним оценкам на 40-60% быстрее, или больше в единицу времени, чем с использованием обычных технологий.

Из-за постоянного роста конкуренции молодых специалистов на рынке труда каждое высшее учебное заведение нуждается в развитии инновационных методов обучения, чтобы поддерживать высокий статус и значимость в среде работодателей. Инновации характерны для любой профессиональной деятельности человека и поэтому естественно становятся предметом изучения, анализа и внедрения. Инновации сами по себе не возникают, они являются результатом научных поисков, передового педагогического опыта отдельных преподавателей и целых коллективов. Этот процесс не может быть стихийным, он нуждается в управлении.

Учебно-методическая база, образовательно-информационные технологии любого ВУЗа таковы, что они вообще не зависят от того, дневная ли это форма обучения, или заочная, или удаленная. Во многих российских университетах существует тенденция проводить унификацию учебных материалов путем создания их электронных версий. Если знания, весь учебный материал, вся его дидактическая составляющая оформлены и находятся в формализованном виде, на электронных источниках, не важно, куда подать эти знания: то ли в аудиторию (с соответствующим оснащением), то ли находящемуся за пределами учебного заведения пользователю. Это особенно полезно, если запланирован

большой объем самостоятельной подготовки студентов, например, при таких формах обучения как вечерняя, заочная и экстернат.

Одним из видов инновационной деятельности в области образования является применение технологий дистанционного обучения. Министерство образования за последние годы приняло и всячески поддерживает применение таких технологий. В приказе №137 Министерства образования и науки РФ от 06.05.2005 г. «Об использовании дистанционных образовательных технологий» отмечено: «Целью использования ДОТ (дистанционных образовательных технологий) образовательным учреждением является предоставление обучающимся возможности освоения образовательных программ непосредственно по месту жительства обучающегося или его временного пребывания (нахождения)».

При применении дистанционного обучения в образовательном процессе роль преподавателя все больше будет иметь консультативное значение, тем самым освобождая педагогу больше времени для научной деятельности. Это дает преимущество учебным заведениям, использующим такого рода инновации, так как курс развития образования, заданный правительством Российской Федерации, предполагает интенсивное развитие научной и опытной деятельности в ВУЗах страны.

Современные компьютерные телекоммуникации способны обеспечить передачу знаний и доступ к разнообразной учебной информации наравне, а иногда и гораздо эффективнее, чем традиционные средства обучения. Следует так же отметить, что применение новых технологий повышает интерес молодежи к процессу обучения.

Необходимость развития дистанционного обучения как самостоятельного или вспомогательного метода подтверждается также интересом к нему мирового сообщества. По данным зарубежных экспертов к 2000 году минимальным уровнем образования, необходимым для выживания человечества, стало высшее образование. Обучение такой массы студентов по очной (дневной) форме вряд ли выдержат бюджеты даже самых благополучных стран. Поэтому не случайно

за последние десятилетия численность обучающихся по нетрадиционным технологиям растет быстрее числа студентов дневных отделений. Мировая тенденция перехода к нетрадиционным формам образования прослеживается и в росте числа ВУЗов, ведущих подготовку по этим технологиям. За период 1900-1960 гг. их было создано 79, за 1960-1970 гг. - 70, а только за 1970-1980 гг. - 87. Долговременная цель развития СДО в мире – дать возможность каждому обучающемуся, живущему в любом месте, пройти курс обучения любого колледжа или университета. Это предполагает переход от концепции физического перемещения студентов из страны в страну к концепции мобильных идей, знаний и обучения с целью распределения знаний посредством обмена образовательными ресурсами.

Еще одним немаловажным преимуществом дистанционных технологий является стоимость обучения. Эксперты считают, что телекоммуникационное интерактивное преподавание обходится на 20-25% дешевле традиционного. Компания «Microsoft» считает, что стоимость сетевого обучения может снизиться как минимум вдвое по сравнению с традиционным обучением, поскольку преподаватель в состоянии давать уроки, находясь где угодно. При этом особого компьютерного оборудования не требуется. Это особенно важно в особых ситуациях, таких как болезнь преподавателя, командировка либо просто невозможность присутствия на занятиях. Экономия может быть достигнута и за счет других факторов. Взяв на вооружение систему дистанционного обучения, учебный отдел может быть уверен, что все учащиеся пользуются одними и теми же и, кроме того, самыми свежими учебно-методическими материалами. Ведь обновлять учебные пособия с помощью Интернет гораздо легче.

Может показаться, что применение СДО не эффективно потому, что без визуального контакта с преподавателем нельзя с точностью сделать вывод о подготовленности студента. Но это не так. К примеру, при внедрении электронных семинаров в процесс обучения, студентам на таких занятиях вести себя пассивно труднее, чем на обычных занятиях. Ведь активность студентов

контролирует программа, позволяя преподавателям легко сделать вывод о способностях студента.

Несомненное преимущество индивидуального подхода при дистанционном обучении иллюстрируется исследованиями, проведенными американскими разработчиками стандарта «SCORM». Ученые сравнивали занятия в классе и индивидуальное обучение. Были отмечены следующие особенности:

- в среднем, в час на студента группы при работе в классе приходится 0.1 вопроса. При частном обучении студент может спросить или ответить на 120 вопросов в час;
- для 98% студентов эффективность индивидуальной работы повышается на 50%.

Следует отметить, что под «индивидуальной работой» подразумевается способ обучения, когда со студентом работает один преподаватель или обучающая программа.

Индивидуальная работа, несомненно, дает лучший результат. Но в условиях массового образования такой способ обучения становится очень дорогим. Использование информационных технологий для подачи материала и инструкций может решить эту проблему. Поэтому электронное обучение, благодаря развитию технологий, становится все более эффективным и доступным.

Подводя итог, можно сделать вывод о необходимости применения технологий дистанционного обучения, выделив их преимущества перед традиционными методами:

- дистанционное обучение позволяет существенно повысить эффективность процесса самоподготовки студентов;
- организация хранения учебных материалов в электронном виде позволяет облегчить доступ студентам и преподавателям к необходимой информации;
- современные технологии СДО помогают улучшить качество обучения, особенно при заочной и вечерней форме;

- автоматизация обучения облегчает работу преподавателей;
- современные компьютерные технологии привлекают студентов к процессу обучения;
- дистанционные образовательные технологии поддерживаются Министерством образования Российской Федерации;
- дистанционные технологии дают огромные возможности роста и развития учебного заведения в мировой образовательной системе;
- применение СДО удешевляет процесс обучения;
- применение СДО улучшает процесс обучения.

Именно благодаря этим преимуществам системы дистанционного обучения используются повсеместно и являются неотъемлемой частью образовательного процесса в лучших высших учебных заведениях мира. Безусловно, для поддержания статуса образовательного учреждения и повышения качества образования требуется взять на вооружение данные технологии.

1.3. ПОСТАНОВКА ЗАДАЧИ

При правильном подходе к разработке любых систем непосредственно перед моделированием требуется выделить основные требования к проекту. В случае с разработкой дистанционного обучения необходимо понять, какие задачи ставятся перед системой, каким образом интегрировать ее в процесс обучения и каковы ресурсы и возможные затраты на проектирование и поддержку. Но прежде всего, важно проанализировать состояние образовательного процесса, на базе которого строится система, а именно выделить исходные данные, требуемые для разработки.

Разобрав в пункте 1.1 современные образовательные процессы, можно сделать вывод, что обучение студентов проходит по сценарию, показанному на рис. 1.2.

Учащийся посещает лекции, семинары, практические и лабораторные занятия под руководством преподавателей, изучает дополнительный материал в

часы самостоятельной подготовки. Завершает весь процесс итоговый экзамен по дисциплине.

Важной задачей разработчика СДО является организация процесса обучения именно по такому сценарию.



Рис. 1.2. Обобщенный сценарий учебного процесса

На сегодняшний день любое образовательное заведение имеет достаточно учебного материала для проведения того или иного вида занятия. Учебная литература хранится в библиотеках ВУЗа. При дистанционном обучении студенты должны иметь доступ к учебным материалам непосредственно с рабочего места.

Занятия с учащимися проводит преподавательский состав, который также должен привлекаться при дистанционном обучении.

Исходя из этого, можно предположить, что задачей системы дистанционного обучения является предоставление доступа к учебным материалам ВУЗа и обеспечение непосредственного общения преподавателя и студента.

Для больших учебных заведений главной задачей модернизации обучения является *обеспечение доступности к системе образования всем студентам и преподавателям*. Использование Интернет при взаимодействии с учащимися, конечно же, очевидно. Но не стоит забывать о *безопасности передаваемых данных*. При этом вопросы безопасности носят главенствующий характер, нежели доступность источника информации. Так же нужно помнить о том, что

процесс оценки знаний студентов должен быть наиболее защищен. Иначе дистанционное обучение не будет иметь смысла и не сможет конкурировать с традиционными методами.

Процесс создания учебного материала в системе дистанционного обучения требует знаний языков программирования хотя бы высокого уровня. Эти знания могут быть недоступны для некоторых преподавателей. Поэтому разработчик ядра системы должен применять более сложные технические средства программирования, создавая шаблоны и структуры. Такие шаблоны должны позволить пользователям системы создавать свои материалы, не углубляясь в программирование.

С ростом интереса к использованию электронного образования и усовершенствования современных технологий связи возможны ситуации, когда требования к функциональному состоянию системы повысятся. При этом *нужно будет добавлять новые функции и возможности*. Это так же следует учесть при построении.

Подводя итог, можно выделить основные требования к системе дистанционного обучения:

- идеология и методы дистанционного обучения не должны отличаться от традиционных;
- система дистанционного обучения должна предоставлять доступ к учебным материалам ВУЗа;
- интерфейс системы должен быть доступен любому пользователю;
- система должна быть устойчива при работе и безопасна;
- работа в системе должна быть удобна для использования;
- должна быть предусмотрена возможность расширяемости и масштабирования системы.

По сути, самым главным требованием к системе дистанционного обучения является то, что процесс дистанционного обучения должен как можно меньше отличаться от традиционного обучения.

2. МОДЕЛИРОВАНИЕ СИСТЕМЫ ДИСТАНЦИОННОГО ОБУЧЕНИЯ

2.1 ОБЩЕЕ ТЕХНИЧЕСКОЕ ЗАДАНИЕ НА СИСТЕМУ

2.1.1 ОБЩИЕ ТРЕБОВАНИЯ

Система дистанционного обучения предназначена для обучения студентов Санкт-Петербургского государственного университета телекоммуникаций им. проф. М.А. Бонч-Бруевича по формам очного, заочного, вечернего образования и экстернатуры. Подразумевается использование системы в целях повышения качества обучения, создания краткосрочных курсов для повышения квалификации учащихся, а так же для организации и накопления знаний в различных областях науки, изучаемых и университете.

Реализация должна представлять собой программный продукт, позволяющий создать среду для разработки образовательных программ на основе готового учебного плана университета. Взаимодействие с системой должно осуществляться через глобальную сеть с использованием протоколов прикладного уровня.

Система дистанционного обучения должна отвечать следующим требованиям:

- доступность: способность предоставлять доступ к учебным компонентам из точки удаленного доступа;
- адаптируемость: способность адаптировать учебную программу согласно индивидуальным потребностям образовательных программ;
- эффективность: способность увеличивать эффективность и производительность, сокращая время и затраты на доставку учебных материалов;
- долговечность: способность соответствовать новым технологиям без дополнительной и дорогостоящей доработки;
- интероперабельность: способность использовать учебные материалы вне зависимости от платформы, на которой они созданы;

- возможность многократного использования: способность использовать систему в разных контекстах.

2.1.2 ТРЕБОВАНИЯ К ПОЛЬЗОВАТЕЛЯМ СИСТЕМЫ

Пользователи системы должны быть разделены по обязанностям и иметь определенные права доступа к элементам системы. Рекомендуется создать три вида пользователей: администраторов системы управления обучением, преподавателей и студентов.

Администраторы должны иметь возможность редактировать всю введенную в систему информацию, иметь доступ ко всем модулям, а так же добавлять пользователей в систему дистанционного обучения.

Преподаватели должны иметь возможность создавать интерактивные курсы, а так же добавлять в систему информационный материал. Также должна быть осуществлена возможность контроля успеваемости студентов.

Студенты должны иметь доступ к курсам, проходить тесты и иметь возможность консультироваться с преподавателями.

2.1.3 ТРЕБОВАНИЯ К ОРГАНИЗАЦИИ МЕХАНИЗМА ВЗАИМОДЕЙСТВИЯ ПОЛЬЗОВАТЕЛЕЙ

Организация взаимодействия между пользователями должна осуществляться путем введения модуля сообщений. Благодаря этой части системы осуществляется общение студентов и преподавателей, а так же организовывается справочная поддержка.

2.1.4 ТРЕБОВАНИЯ К МОДУЛЮ НАКОПЛЕНИЯ ДАННЫХ

В системе необходимо организовать механизмы хранения учебных материалов, способы их описания, обмена и поиска. Все введенные данные должны быть четко структурированы и представляться в виде курса. Курс должен состоять из лекций и заканчиваться тестированием учащихся.

Модуль накопления данных необходимо унифицировать и организовать поддержку введения данных из других баз хранения данных, распространенных в сети Интернет. Требуется продумать последующий процесс модернизации данных с использованием метаязыков.

2.1.5 ТРЕБОВАНИЯ К ПРОВЕДЕНИЮ ОБРАЗОВАТЕЛЬНЫХ ПРОЦЕССОВ В СИСТЕМЕ

Доступ к образовательным ресурсам, созданным преподавателями, должен осуществляться через администраторов. Студентов, участвующих в обучении, необходимо объединять в группы. Администратору необходимо сопоставлять созданные курсы с необходимыми группами студентов, назначив при этом ответственного преподавателя. Изучив лекционный материал, учащиеся обязаны пройти тестовое задание. Тест должен состоять из вопросов и вариантов ответов на них. После прохождения результаты выполнения тестов сохраняются в системе. Преподавателю, ответственному за изучение курса, предоставляется доступ к результатам группы. На всем протяжении курса у студентов должна быть возможность консультации с преподавателем.

Поддержка изменения тестового модуля системы обязательна.

2.1.6 ТРЕБОВАНИЕ К ИНТЕРФЕЙСАМ ПОЛЬЗОВАТЕЛЕЙ

Система дистанционного обучения должна иметь дружелюбный интерфейс. Подход к разработке визуального оформления должен осуществляться с учетом требований всех возрастных групп возможных пользователей.

Требуется организовать модуль поддержки пользователей, по возможности используя элементы технологий искусственного интеллекта. Возможно создание платформы для разработки справочной информации по системе и по технологии дистанционного обучения в целом.

2.1.7 ПРИМЕНЕНИЕ УЖЕ СОЗДАННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

При разработке системы возможно применение готового ПО, распространяющегося в рамках лицензии GNU GPL. Это подразумевает предоставление пользователю права копировать, модифицировать и распространять (в том числе на коммерческой основе) программы (что по умолчанию запрещено законом об авторских правах), а также гарантировать, что и пользователи всех производных программ получают вышеперечисленные права.

Так же подразумевается использование платных программ, но в целях упрощения разработки системы, внедрения проверенных технологий и повышения безопасности работы.

2.1.8 ТРЕБОВАНИЯ К ВОЗМОЖНОСТЯМ МОДЕРНИЗАЦИИ

В системе дистанционного обучения должна быть реализована возможность модернизации и замены модулей. Модель системы должна позволять адаптировать ее к изменяющимся в процессе ее эксплуатации условиям и допускать поэтапную модернизацию отдельных компонентов.

Для достижения этих целей в процессе программирования необходимо использовать объектно-ориентированный стиль и соблюдать требования к процессу программирования.

2.1.9 ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ К РЕАЛИЗАЦИИ СИСТЕМЫ

Физическую реализацию требуется организовать на ЭВМ с операционной системой Linux. Рекомендуемые сборки операционных систем – Ubuntu Server, MacOS. Так же требуется поддержка следующих основных сервисов:

- сервер «Apache 2»;
 - сервер баз данных «MySQL4»,
- а так же дополнительных сервисов:
- сервер DNS «bing9»;
 - сервер FTP.

Компьютер должен быть подключен к сети Интернет по широкополосному каналу. Осуществить это надо по возможности с использованием межсетевого экрана.

Программную реализацию требуется создать на языках HTML, PHP5, SQL, Javascript. Возможно использование различных Фреймворков типа «jQuery». Модули программ должны быть понятными – достаточно короткими для простоты, но слишком маленькие модули порождают слишком много ветвей в структуре дерева, и иерархия становится сама по себе трудной для понимания. Обязательно использование классов, функций и процедур, вместо повторения кода.

Базы данных необходимо реализовывать с использованием драйвера «InnoDB».

Права пользователей должны быть организованы в контексте базы данных, а так же файловой системы. Для каждого курса рекомендуется организовать собственную директорию для хранения файлов, загруженных разработчиком.

2.2 ОБЗОР ПЛАТФОРМЫ ДЛЯ СОЗДАНИЯ СИСТЕМЫ ДИСТАНЦИОННОГО ОБУЧЕНИЯ

Для создания программного компонента системы дистанционного обучения основным языком программирования выбран скриптовый язык PHP. Этот язык широко применяется для разработки веб-приложений и создания динамических веб-сайтов. Благодаря своей простоте, богатой функциональности, кроссплатформенности и бесплатному распространению исходных кодов этот язык хорошо применим при разработке больших систем, работающих в сети Интернет. К крупнейшим сайтам, использующим PHP, относятся «Facebook», «В контакте», «Wikipedia», «YouTube».

К достоинствам языка относится наличие таких средств как:

- автоматическое извлечение POST и GET-параметров, а также переменных окружения веб-сервера в предопределённые массивы;
- файловые функции успешно обрабатывают как локальные, так и удалённые файлы;
- автоматическая отправка HTTP-заголовков;
- работа с cookies и сессиями;
- обработка файлов, загружаемых на сервер;
- работа с HTTP заголовками и HTTP авторизацией;
- работа с удалёнными файлами и сокетами.

Сценарии языка PHP выполняются на стороне сервера и не требуют от клиента установки дополнительного программного обеспечения, за исключением web-браузера. В качестве ответа клиенту посылается текстовый документ, содержащий инструкции языка HTML, интерпретируемого любым

современным браузером. Благодаря этому достигается доступность системы дистанционного обучения любому пользователю персонального компьютера с подключением к сети Интернет.

Важным достоинством языка PHP являются широкие возможности использования баз данных. Хранение данных средствами баз данных весьма эффективно при большом количестве хранимых данных. Самой распространенной системой управления базами данных является «MySQL», разработанная компанией «Sun Microsystems» и принадлежащая мировому лидеру в этой области – компании «Oracle Corporation». Обычно «MySQL» используется в качестве сервера, к которому обращаются локальные или удалённые клиенты.

Важной чертой баз «MySQL» является возможность репликации данных, то есть создание резервных копий.

Гибкость СУБД «MySQL» обеспечивается поддержкой большого количества типов таблиц. Техническое задание проекта предписывает использование «InnoDB» таблиц, и на это существует ряд причин. Во-первых, драйвер «InnoDB» поддерживает транзакционную модель. Транзакционный запрос позволяет сохранить целостность данных при изменении или удалении. Во-вторых, возможность резервирования и восстановления лучше реализована в таблицах именно этого типа.

Другим языком программирования, применение которого возможно при разработке системы, является «JavaScript». «JavaScript» обычно работает как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности web-страницам.

В статье «Самый непонимаемый язык программирования в мире стал самым популярным в мире языком программирования» Дуглас Крокфорд утверждает, что лидирующую позицию «JavaScript» занял в связи с развитием «AJAX», поскольку браузер стал преобладающей системой доставки приложений. Именно применение технологии «AJAX» очень полезно при

разработке системы дистанционного обучения. И на это существуют свои причины.

Web-приложения – это приложения, функциональные возможности которых обеспечиваются сервером и доставляются по сети Интернет. Классическая модель Web-приложения действует следующим образом. Клиентское приложение отправляет на сервер HTTP-запрос. Сервер проводит необходимую обработку и затем выдает HTML-страницу клиенту (рис. 2.1).

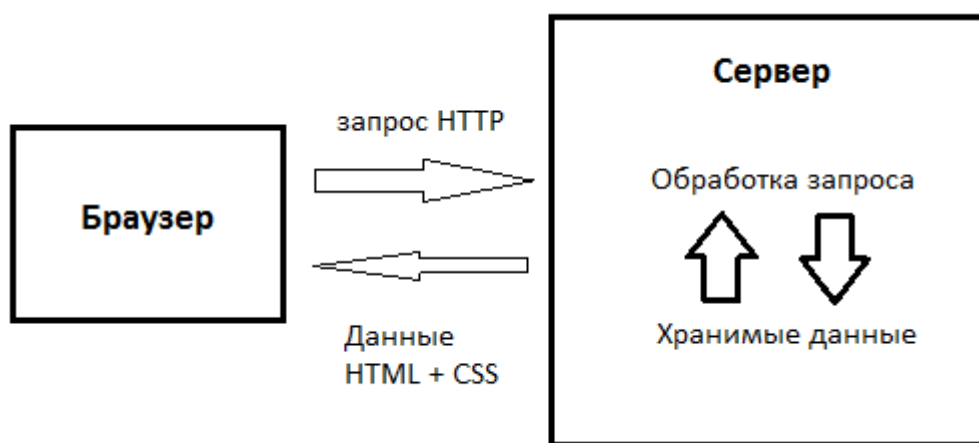


Рис. 2.1. Классическая схема работы Web-приложения

Существенным недостатком такого алгоритма является то, что клиенту приходится ждать загрузки каждой последующей страницы.

Джесси Гарретт, изложивший идею AJAX, предложил сократить ожидание или сделать его совсем незаметными. В расшифровке аббревиатуры AJAX и кроится смысл технологии. *Asynchronous JavaScript and XML* – технология, позволяющая осуществить передачу HTTP запроса без перезагрузки страницы. Такая передача происходит в результате работы сценария JavaScript (рис. 2.2).

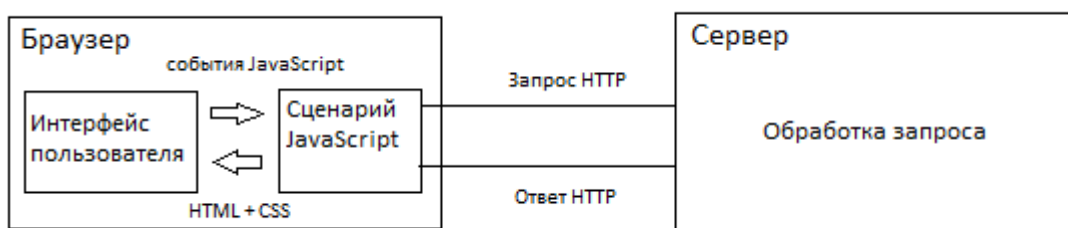


Рис. 2.2

Разумное использование этой технологии может сделать систему дистанционного обучения более дружественной к пользователю. AJAX можно применить для решения следующих задач:

- проверка правильности заполнения форм с привлечением возможностей сервера;
- подсказки при заполнении форм;
- создание динамических средств изменения содержимого базы данных;
- разработка модулей, в которых требуется динамическое обновление содержимого.

Применение всех вышеуказанных средств программирования позволит создать качественную, удобную и безопасную систему дистанционного обучения для студентов Санкт-Петербургского государственного университета телекоммуникаций им. проф. М.А. Бонч-Бруевича.

2.3 РАЗРАБОТКА МОДЕЛИ СИСТЕМЫ ДИСТАНЦИОННОГО ОБУЧЕНИЯ

2.3.1 ФЕНОМЕНОЛОГИЯ СИСТЕМ ДИСТАНЦИОННОГО ОБУЧЕНИЯ

Для того чтобы дальнейшее изложение разработки было более понятным и четким, требуется определить некоторую феноменологию процессов и объектов, используемых в процессе моделирования.

Система дистанционного обучения, СДО – программно-аппаратный комплекс, позволяющий осуществить процесс дистанционного обучения студентов посредством сети Интернет. Разработка СДО является целью данной дипломной работы.

Администратор СДО – лицо, являющееся пользователем системы и выполняющее функции контроля над процессом обучения.

Преподаватель – лицо, являющееся пользователем системы и выполняющее функции создания учебных материалов для курса, а так же контролирующее процесс обучения групп студентов.

Студент – лицо, являющееся пользователем системы и проходящее обучение в ней по различным курсам.

Группа – множество студентов, объединенных по признаку выпускных специальностей. Используется при назначении курса для обучения студентов.

Кафедры, Департаменты – множество преподавателей, объединенных по признаку определённой специализации. Используется для простоты организации и поиска.

Учебные материалы – информационные ресурсы, созданные преподавателями системы и используемые для обучения студентов, а так же для публикации их в библиотеке системы. К учебным материалам относятся лекции и тесты.

Курс – совокупность учебных материалов, объединенных по определенной тематике. Курс, как и его содержание, создается преподавателем.

Библиотека СДО – база знаний, основанная на лекциях преподавателя системы, разрешенных для публикации. Библиотека доступна любому пользователю, не требуя при этом регистрации в системе.

Лекция – набор информационных материалов в виде текста, цифровых изображений, flash-клипов. Лекции используются преподавателем при создании курсов. Преподаватель может публиковать лекции в библиотеке СДО для общего доступа к ним любых пользователей Интернет.

Тест – информационный ресурс системы, созданный преподавателем для осуществления контроля знаний студентов после изучения учебного материала. Тест является неотъемлемой частью курса, состоит из вопросов и вариантов ответа на них и представляет собой интерактивный опрос студентов по определенным темам, осященным в лекциях курса.

Свободные информационные ресурсы – ресурсы системы, доступные любому пользователю сети Интернет. К таким ресурсам относятся материалы библиотеки СДО.

Создатель курса – преподаватель, который разработал определенный курс.

Администратор курса – преподаватель, назначенный для контроля результатов успеваемости студентов, изучающих определенный курс, а так же для консультации в рамках этого курса.

2.3.2 ФУНКЦИИ ПОЛЬЗОВАТЕЛЕЙ СИСТЕМЫ

Для успешного осуществления процессов обучения, прежде всего, требуется разобраться с функциями пользователей системы. Техническое задание определяет три вида пользователей: администраторов, преподавателей и студентов. При организации общедоступного информационного ресурса (библиотеки) следует так же учесть, что системой может пользоваться любой пользователь сети Интернет. В табл. 2.1 представлены требуемые функции для каждого типа пользователей.

Таблица 2.1

Функции пользователей системы

Тип пользователя	Необходимость авторизации пользователя	Функции пользователя в системе
Преподаватель	Да	1. Создание и управление курсами 3. Создание лекций и методических материалов 4. Разработка контрольных заданий 5. Контроль успеваемости 6. Консультация студентов
Студент	Да	1. Изучение материалов курса 2. Прохождение контрольных заданий 3. Консультация с преподавателями 4. Контроль собственных результатов
Тип пользователя	Необходимость авторизации пользователя	Функции пользователя в системе
Администратор	Да	1. Создание и управление преподавателями 2. Администрирование учебного процесса 3. Управление курсами и

		информационными ресурсами системы
Пользователь сети Интернет	Нет	1. Получение доступа к свободным информационным ресурсам системы

2.3.3 УЧЕБНЫЙ ПРОЦЕСС

Исходя из функций пользователей системы, можно смоделировать процесс обучения в среде СДО. Обучение в системе можно разделить на несколько этапов. Прежде всего, необходимо создать штатную структуру системы. Это является обязанностью администратора, который создает пользователей и наделяет их определенными правами.

В любом высшем образовательном учреждении преподаватели объединяются в кафедры, а студенты в группы. Очевидно, что для интеграции традиционных методов обучения в систему дистанционного обучения, требуется деление студентов и преподавателей по такому же признаку.

Одной из функций преподавателя, как видно из табл. 2.1, является создание курсов. Курс создается не для определенной группы студентов. Он может быть назначен нескольким группам, а может быть и вовсе не введен в процесс обучения или опубликован в библиотеке системы. Если курс будет назначен большому количеству студентов, управлять результатами учащихся становится сложно. Особенно, если этим занимается только создатель курса. Функцию назначения курсов группам студентов осуществляет администратор. Становится ясным, что для обучения группы в рамках одного курса требуется назначать ответственного преподавателя, который будет контролировать результаты и проводить консультации. При этом создатель курса исполняет роль лектора, разрабатывая новые и совершенствуя старые учебные материалы. Для него нет необходимости следить за результатами курса. Если студенту требуется помощь, он обращается к администратору курса. Тот, в свою очередь, имеет возможность общаться с создателем курса, если ему что-то не ясно. Это поможет разделить и облегчить работу преподавателей.

Еще одним важным элементом в модели обучения является проверка курса. В некоторых ситуациях создатель курса может не понять требований к разработке курса и оформить его не правильно. Для того чтобы исключить такую возможность, вводится функция подтверждения курсов. При создании нового курса администратору приходит уведомление. Он проверяет учебный материал и подтверждает их право на публикацию и возможность использования в процессе обучения. При этом решение о помещении курса в библиотеку СДО принимает преподаватель, создавший курс, а назначает его группам для прохождения – администратор. Модель учебного процесса представлена на рис. 2.3.

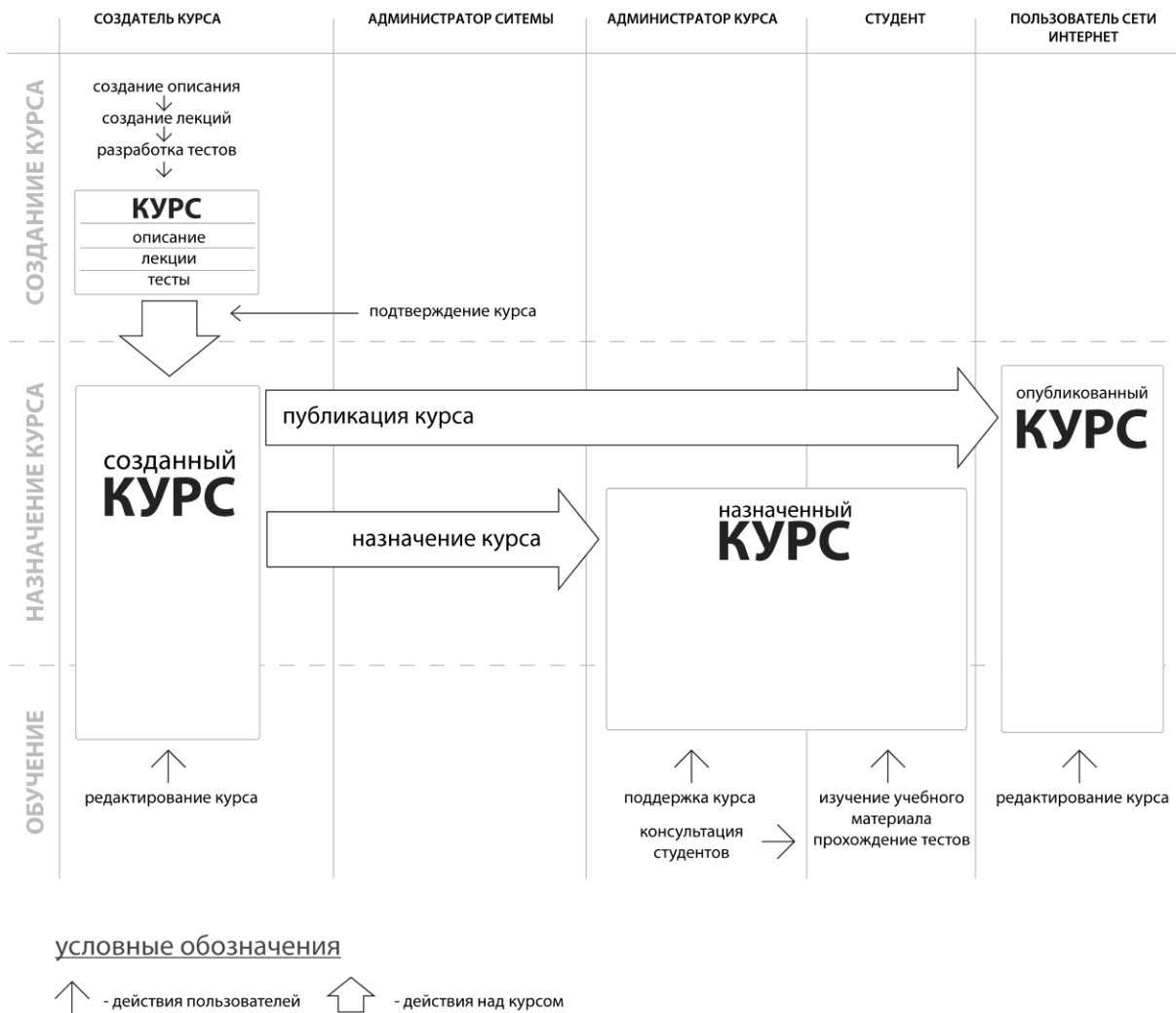


Рис. 2.3. Модель учебного процесса

Необходимо пояснить, что один и тот же курс в модели может выполнять несколько функций. Он может использоваться и как учебный материал для обучения студентов системы, так и как свободный информационный ресурс.

2.3.4 ДЕКОМПОЗИЦИЯ МОДЕЛИ ОБУЧЕНИЯ

При проектировании сложной системы ее разбивают на части, каждая из которых затем рассматривается отдельно. Возможны два различных способа такого разбиения на подсистемы: структурное (или функциональное) разбиение и объектная (компонентная) декомпозиция.

При *функциональной декомпозиции* программной системы ее структура может быть описана блок-схемами, узлы которых представляют собой «обрабатывающие центры» (функции), а связи между узлами описывают движение данных. В разрабатываемой системе дистанционного обучения можно выделить две подсистемы: подсистема накопления учебных данных и подсистема обучения. В технологии программирования процессы, проходящие в таких системах, называют *back-end* и *front-end*, соответственно. *Front-end* (фронт-энд) и *back-end* (бэк-энд) — это обобщенные термины, которые отражают начальное и конечное состояния процесса. *Front-end* отвечает за получение ввода (входной информации) в любых формах от пользователя и обработку полученной информации в ту форму, которую *back-end* способен использовать. *Front-end* — это интерфейс между пользователем и *back-end*'ом. Соотношения функций системы с процессами *front-end* и *back-end* приведены в табл. 2.2.

Таблица 2.2

Функции пользователей системы

Процессы	Функции
Front-end	<ol style="list-style-type: none"> 1. Создание пользователей 2. Создание и управление курсами 3. Создание лекций и методических материалов 4. Разработка контрольных заданий 5. Назначение курсов

Back-end (подсистема обучения)	1. Обучение студентов 2. Доступ к свободным информационным ресурсам
--------------------------------	---

Совокупность работы этих двух подсистем сможет обеспечить качественное обучение.

Дальнейшее разбиение подсистем приведет к модульной структуре, где за каждое действие отвечает отдельный модуль. Это позволит осуществить легкую возможность модернизации системы. Чтобы добавить новую функцию или исправить недочеты в системе потребуется исправить лишь один модуль.

Наряду к функциональной, к системе можно применить объектную декомпозицию. В качестве объектов разумно использовать пользователей системы. У каждого пользователя существуют свойства, описывающие его и методы, используемые для организации функционала. Общая диаграмма зарегистрированного пользователя системы приведена на рис. 2.4.

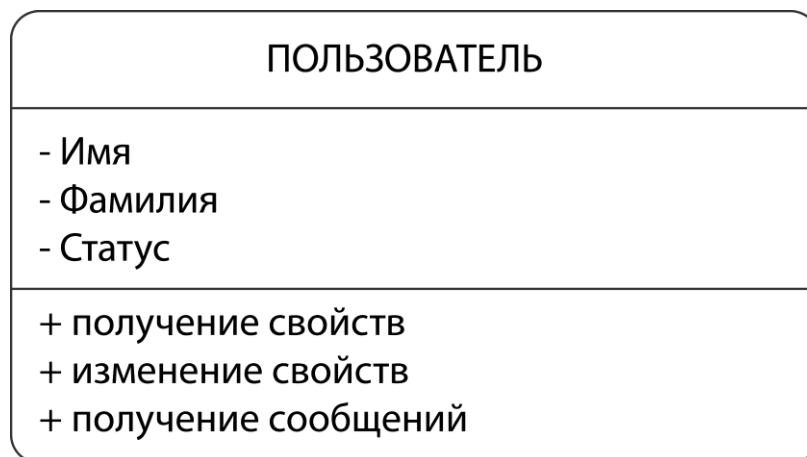


Рис. 2.4. Общая диаграмма пользователя системы

Как видно из рис. 2.4, любой пользователь имеет такие свойства как имя, фамилия и статус. Свойство статус определяет принадлежность пользователя к типу: администратор, преподаватель, студент. Для получения и изменения собственных свойств существуют свои методы. Так же у каждого пользователя есть метод получения сообщений. Здесь под получением сообщения

подразумевается, как и функции посылки, так и приема сообщений от пользователей системы.

К диаграммам других пользователей добавлены новые свойства и методы, требуемые для организации функциональных возможностей. Диаграмма администратора представлена на рис. 2.5.

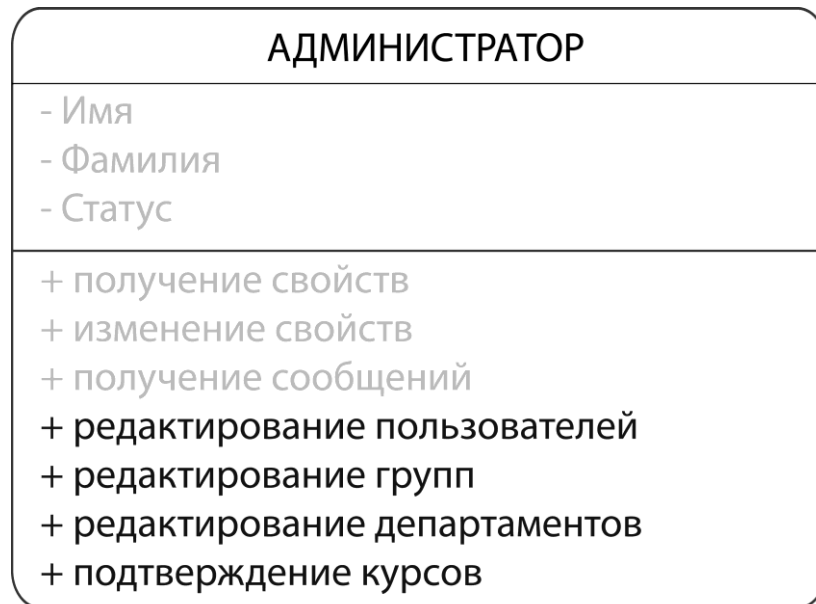


Рис. 2.5. Диаграмма администратора системы

Серым цветом на диаграмме показаны общие для всех пользователей свойства и методы. У администратора так же существуют методы для создания штатной структуры системы. Так же есть метод «подтверждение курсов», что так же подразумевает их редактирование.

У студентов и преподавателей в отличие от администраторов появляется свойство структурной принадлежности. У преподавателей это свойство носит название «департамент». Диаграмма преподавателей показана на рис. 2.6.

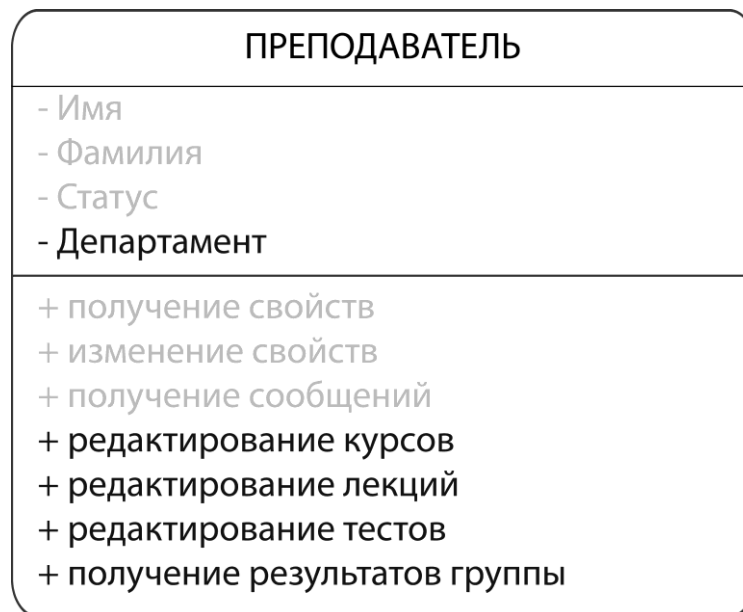


Рис. 2.6. Диаграмма преподавателей системы

К тому же добавлены методы редактирования курсов (создание и редактирование), редактирование лекций, редактирование тестов. Для администраторов курса есть метод получения результатов групп, проходящих курс.

У студентов, помимо свойства, определяющего его принадлежность к группе, добавлены методы получения материалов лекций и выполнения тестов. На рис. 2.7 изображена диаграмма студентов.

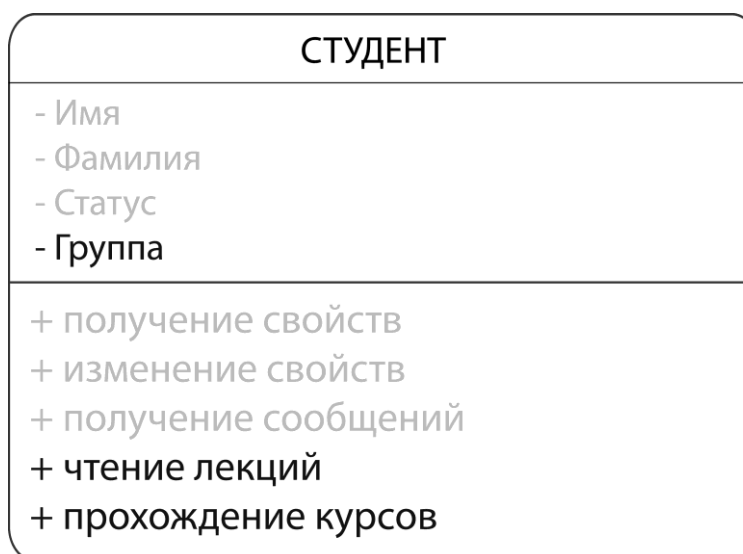


Рис. 2.7. Диаграмма студентов системы

Эти диаграммы после некоторой модернизации можно использовать в качестве UML диаграмм для создания абстрактных классов в программном коде и для объектного описания системы.

С помощью полученной структуры можно приступить к разработке системы дистанционного обучения.

3. РАЗРАБОТКА СИСТЕМЫ ДИСТАНЦИОННОГО ОБУЧЕНИЯ

3.1 РАЗРАБОТКА ИНТЕРФЕЙСА СИСТЕМЫ

Важным элементом любой системы, имеющий непосредственный контакт с пользователем, является интерфейс. Интерфейс – это совокупность средств и методов взаимодействия между элементами системы. Этот термин используется практически во всех областях науки и техники. Его значение относится к любому сопряжению взаимодействующих сущностей.

Программный компонент системы дистанционного обучения представляет собой сайт, имеющий графический интерфейс. При проектировании графических интерфейсов вводится понятие «юзабилити». Юзабилити (usability — дословно «возможность использования», «полезность») — понятие, обозначающее итоговый уровень удобства интерфейса для использования в заявленных целях. На сегодняшний день проведено множество исследований в области разработки интерфейсов сайта. Чтобы в них разобраться и правильно использовать, требуется выделить основные структурные элементы сайта.

Итак, каждый сайт, чтобы обеспечить удобную работу пользователей, должен включать следующие элементы.

Блок идентификации сайта – часть графического интерфейса, позволяющая посетителю однозначно определить, что представляет ресурс, которым он пользуется. Представляет собой логотип (знак) сайта, его название и, возможно, краткое описание.

Гиперссылка – часть документа, ссылающаяся на другой элемент (текст, заголовок, примечание, изображение) в самом документе, на другой объект (файл, директория, приложение), расположенный на локальном компьютере или в компьютерной сети, либо на элементы этого объекта.

Навигация – элемент интерфейса, позволяющий выполнять переход пользователя к определенным структурным частям (страницам) сайта. Состоит из главного и дополнительного меню и представляет собой набор гиперссылок.

Контент – собственно само содержание сайта. Является совокупностью текстовой и графической информации.

Оперируя этими понятиями, можно описать внешний вид системы, соответствующий требованиям удобства использования. Собственно проектирование графического интерфейса для сайта представляет собой организацию всех необходимых элементов на странице, а так же разработку их внешнего вида.

Современные достижения в области эргономики позволили создать, так называемую, «карту кликов». Эта карта показывает, на что именно посетитель сайта обращает свое внимание. Пример «карты кликов» поисковой системы «Google» представлен на рис. 3.1.

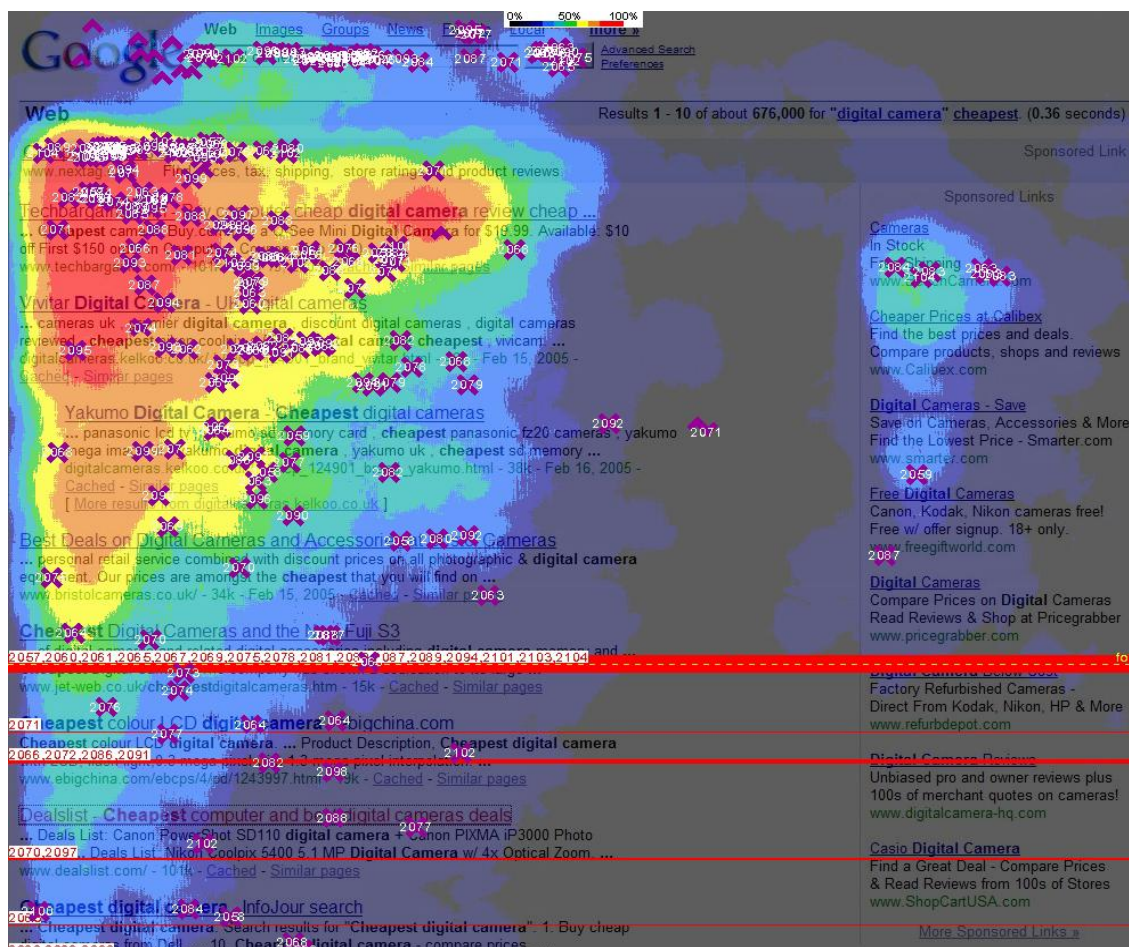


Рис. 3.1. «Карта кликов» сайта google.com

Треугольник, образованный в месте самых частых «кликов» посетителей сайта, специалисты по «юзабилити» называют «золотым треугольником». Именно в этом месте на странице сайта требуется расположить основные структурные элементы, чтобы привлечь внимание пользователей.

Схема сайта системы дистанционного обучения представлена на рис. 3.2.

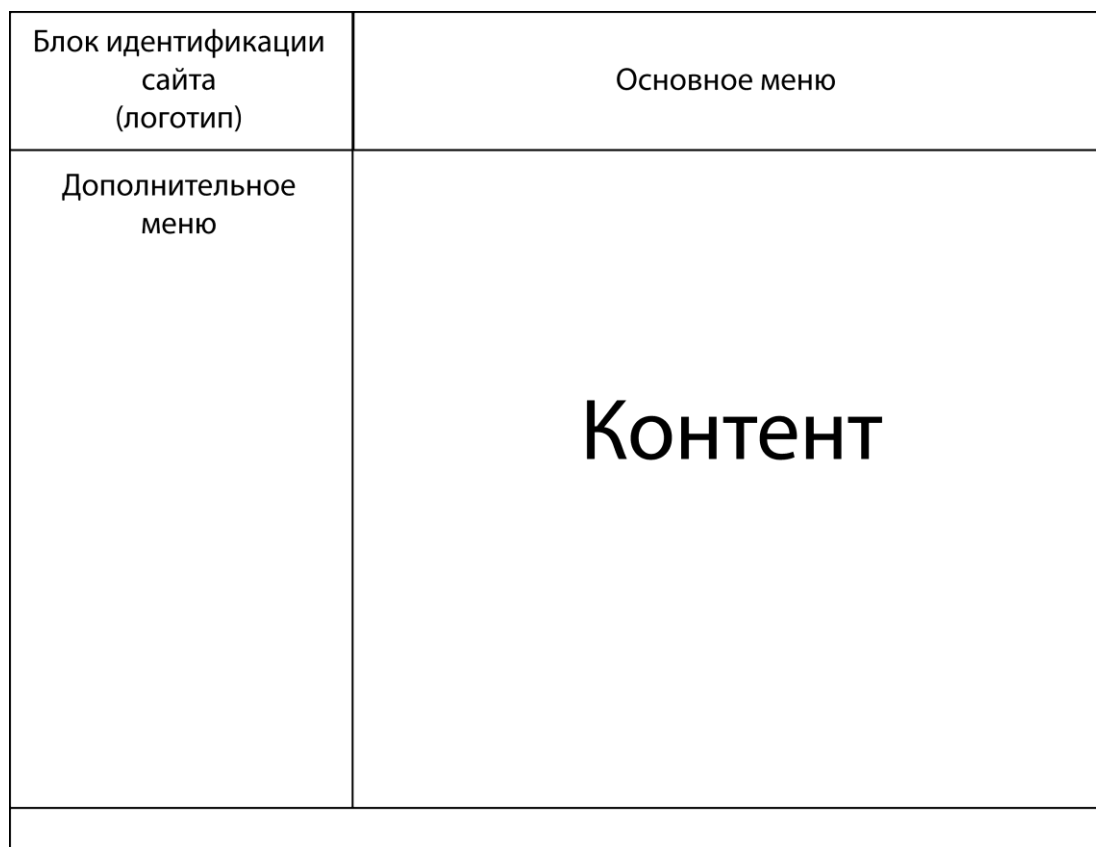


Рис. 3.2. Структура интерфейса сайта

Блок идентификации сайта представляет собой логотип сайта. Основное меню доступно всем пользователям. Поэтому оно будет состоять из ссылок на страницы новостей, библиотеки системы, справку о пользовании системой и информации о способах связи с администратором системы. К этим частям открыт доступ любому пользователю, зашедшему на сайт. Готовый графический интерфейс изображен на рис. 3.3.

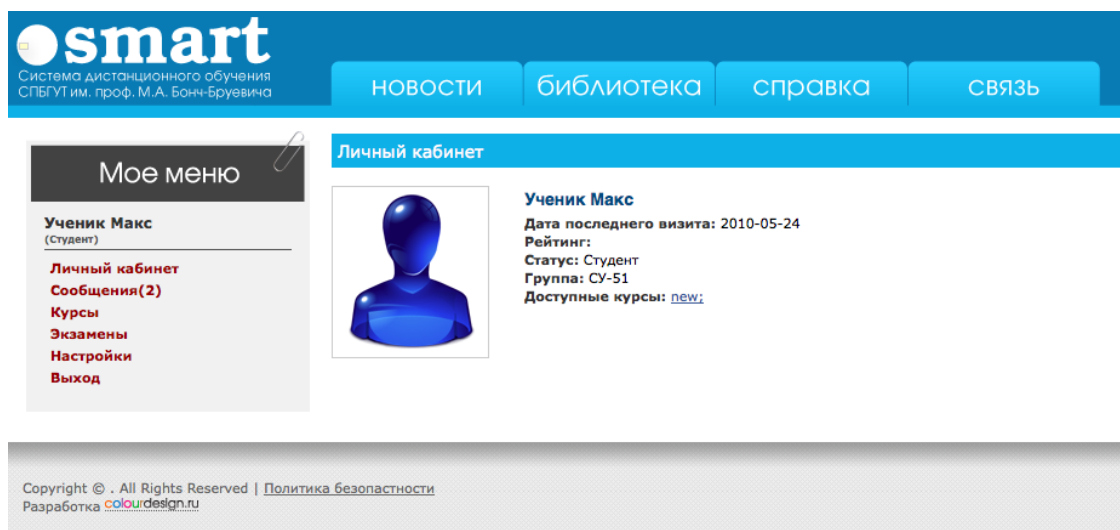


Рис. 3.3. Пример графического интерфейса системы

Дополнительное меню состоит из имени пользователя, его класса и ссылок на доступные ему части сайта.

Необходимо отметить, что, так как зарегистрированным пользователям системы необходимо пройти процесс регистрации, дополнительное меню может быть заменено на форму для ввода логина и пароля на стартовой странице сайта. Система дистанционного обучения в рамках данной работы будет иметь название «Smart». Любое дальнейшее упоминание в тексте пояснительной записки этого названия следует воспринимать как синоним системы дистанционного обучения.

3.2 РАЗРАБОТКА БАЗЫ ДАННЫХ СИСТЕМЫ

Для хранения информации в системе дистанционного обучения предполагается использование баз данных MySQL. Требуется разработать структуру базы данных, чтобы начать процесс программирования. База данных будет иметь имя «smart» и состоять из 12 таблиц. Структура и подробное описание приведено в табл. 3.1.

Таблица 3.1

Структура базы данных *smart*

Название таблицы	Описание таблицы	Название полей	Описание полей
users	Информация о пользователе	id	Идентификатор пользователя. Значение является уникальным и однозначно определяет пользователя.
		email	Для всех типов пользователей, кроме администратора, значением этого поля является контактный адрес электронной почты. Так же используется как логин в процессе авторизации. Значение является уникальным и однозначно определяет пользователя.
		password	Пароль пользователя. Используется в процессе авторизации.
		status	Идентификатор принадлежности к типу пользователя. Имеет три допустимых значения: «1» - администратор, «2» - преподаватель, «3» - студент
		name	Имя пользователя
		surname	Фамилия пользователя

Продолжение таблицы 3.1

Название таблицы	Описание таблицы	Название полей	Описание полей
		group	Идентификатор принадлежности пользователя к группе. Значение поля актуально только для студентов. Является внешним ключом к таблице <i>groups</i>
		departament	Идентификатор принадлежности пользователя к факультету (департаменту). Значение поля актуально только для преподавателей. Является внешним ключом к таблице <i>departaments</i>
		rate	Рейтинг пользователя. Позволяет определить важность вклада пользователя в развитие системы
		lastdate	Дата последнего посещения
		photo	Идентификатор наличия личной фотографии в профиле пользователя
groups	Описание групп студентов	id	Идентификатор группы. Значение является уникальным и однозначно определяет группу. Является ключом к значению поля <i>group</i> в таблице <i>users</i>
		name	Имя группы
		department	Идентификатор принадлежности к факультету. Является внешним ключом к таблице <i>departaments</i>

Продолжение таблицы 3.1

Название таблицы	Описание таблицы	Название полей	Описание полей
departaments	Описание факультетов (департаментов)	id	Идентификатор факультета. Значение является уникальным и однозначно определяет факультет. Является ключом к значению поля <i>departement</i> в таблице <i>users</i> и поля <i>department</i> в таблице <i>groups</i>
		name	Название факультета
messages	Сообщения пользователей	id	Идентификатор сообщения. Значение является уникальным и однозначно определяет сообщение
		from	Идентификатор отправителя. Является внешним ключом к полю <i>id</i> таблицы <i>users</i>
		to	Идентификатор получателя. Является внешним ключом к полю <i>id</i> таблицы <i>users</i>
		date	Дата отправки сообщения
		message	Текст сообщения
		showed	Идентификатор прочтенного сообщения. Если значение равно 1, то сообщение прочитано, если же 0 – то сообщение не прочитано.

Продолжение таблицы 3.1

Название таблицы	Описание таблицы	Название полей	Описание полей
news	Новости проекта	id	Идентификатор новости. Значение является уникальным и однозначно определяет новость
		header	Заголовок новости. Используется для сортировки новостей
		text	Текст новости
		date	Дата создания новости. Используется для сортировки
curses	Информация и настройки курса	id	Идентификатор курса. Значение является уникальным и однозначно определяет курса.
		name	Название курса
		numTQ	Количество вопросов, требуемых для прохождения теста
		qorder	Порядок сортировки вопросов теста. Если значение равно 1 – порядок произвольны, если же 0 – вопросы следуют в порядке создания.
		description	Описание курса
		admin	Идентификатор создателя курса. Является внешним ключом к полю <i>id</i> таблицы <i>users</i>

Продолжение таблицы 3.1

Название таблицы	Описание таблицы	Название полей	Описание полей
curses	Информация о курсах и их свойствах	confirm	Идентификатор подтверждения курса администратором
curses_match	Сопоставление курсов и групп студентов. Является элементом отношения «многие ко многим» языка SQL	id	Идентификатор сопоставления. Значение является уникальным и однозначно определяет сопоставление.
		curse	Идентификатор курса. Является внешним ключом к полю <i>id</i> таблицы <i>curses</i>
		group	Идентификатор группы. Является внешним ключом к полю <i>id</i> таблицы <i>groups</i>
		start_date	Дата начала прохождения курса
		finish_date	Дата окончания прохождения курса
		admin	Администратор курса. Является внешним ключом к полю <i>id</i> таблицы <i>users</i>
lectures	Информация о лекциях и их свойствах	id	Идентификатор лекции. Значение является уникальным и однозначно определяет лекцию
		name	Название лекции
		author	Идентификатор автора лекции. Является внешним ключом к полю <i>id</i> таблицы <i>users</i>

		course	Идентификатор курса. Является внешним ключом к полю <i>id</i> таблицы <i>curses</i>
		update	Дата последнего обновления лекции

Продолжение таблицы 3.1

Название таблицы	Описание таблицы	Название полей	Описание полей
pages	Страницы лекций. Является элементом отношения «многие к одному» языка SQL	source_id	Идентификатор лекции, к которой принадлежит страница. Является внешним ключом к полю <i>id</i> таблицы <i>lectures</i>
		content	Текст лекции
		id	Идентификатор страницы. Значение является уникальным и однозначно определяет страницу
		number	Порядковый номер страницы в лекции
questions	Вопросы тестов	id	Идентификатор вопроса. Значение является уникальным и однозначно определяет вопрос
		course	Идентификатор курса. Является внешним ключом к полю <i>id</i> таблицы <i>curses</i>
		question	Текст вопроса
		variants	Варианты ответа
		answers	Количество правильных ответов
		pic	Идентификатор наличия поясняющего изображения.

Продолжение таблицы 3.1

Название таблицы	Описание таблицы	Название полей	Описание полей
tests_result	Результаты прохождения пользователем теста	user_id	Идентификатор студента, выполнившего тест. Является внешним ключом к полю <i>id</i> таблицы <i>users</i>
		curseid	Идентификатор пройденного курса. Является внешним ключом к полю <i>id</i> таблицы <i>curses</i>
		result	Результат прохождения теста
		date	Дата прохождения теста
tips	Подсказки пользователям системы. Таблица неизменяема пользователями системы. Редактирование производит разработчик текущей версии СДО	id	Идентификатор подсказки. Значение является уникальным и однозначно определяет подсказку
		theme	Тема подсказки. Используется при поиске подсказки.
		tip	Текст подсказки
		header	Заголовок подсказки. Содержит ключевые слова. Является уникальным и однозначно определяет подсказку

Необходимо так же отменить непосредственную связь некоторых таблиц. Это осуществляется благодаря указанным ключам, и с помощью их поддерживается целостность структуры база данных. При удалении, скажем, курса из системы, удалятся и соответствующие строки в таблицах *curses_match* и *lectures*, так как в них заданы внешние ключи к таблице *curses*. По этой же причине удалятся строки в таблице *pages*, имеющие непосредственное отношение к удаленным лекциям. Это и есть основное преимущество таблиц InnoDB перед другими типами таблиц.

Оперируя таблицами этой базы данных можно с успехом осуществлять процесс обучения в СДО.

3.3 РАЗБОТКА ГЛАВНОГО ПРОГРАММНОГО МОДУЛЯ СИСТЕМЫ

Главный модуль системы дистанционного обучения представляет собой совокупность web-приложений, с помощью которого осуществляется поддержка основных функциональных возможностей:

- графический интерфейс;
- авторизация пользователя;
- получение информации о пользователе;
- настройка пользователя;
- выход из системы.

Чтобы обеспечить требование последующей модернизации системы, разумно использовать объектно-ориентированный подход к программированию моделей. Все объекты в системе должны представлять собой экземпляры классов. Все функции системы будут являться методами в определенных классах.

3.3.1 КЛАСС ИНТЕРФЕЙСА СИСТЕМЫ

Для построения графического интерфейса в системе служит класс *html*. Он состоит из методов, требуемых для создания *html* верстки сайта.

Класс *html* имеет два свойства – *title* и *page*. При создании экземпляра задается заголовок страницы (свойство *title*) и раздел сайта (новости,

библиотека, справка или связь, свойство *page*). С помощью этих свойств методом *header()* генерируется часть страницы до пользовательского меню.

Рассмотрим часть кода, осуществляющую проверку возможности доступа к внутренним ресурсам системы:

```
if(empty($_SESSION['id'])) {  
    $UI->aunt();  
} else {  
    $UI->menu($_SESSION['id']);  
}
```

Механизм безопасности системы требует инициализации сессии при просмотре каждой страницы сайта. При этом создается глобальный массив *_session*, где хранятся переменные сессий. Механизм сессий используется в таких социальных сетях, как *facebook.com*, *vkontakte.ru* и *twitter.com*, и является самым надежным средством для осуществления доступа к определенным страницам сайта. При успешном прохождении авторизации пользователем в системе создается переменная *id* в массиве *_session*, куда сохраняется идентификатор пользователя из базы данных. Соответственно, если такой переменной в массиве нет, то пользователь не прошел процесс авторизации. В этом случае вызывается метод *aunt()* класса *html*, который добавляет на страницу форму для авторизации (рис. 3.4).

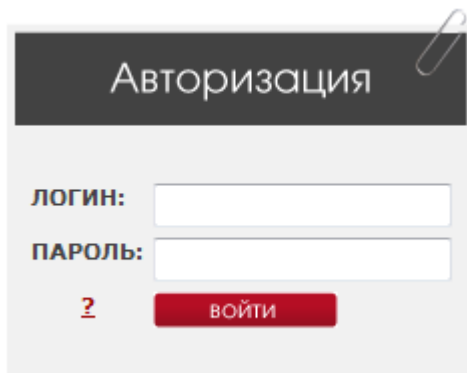
The image shows a user authentication form. At the top, there is a dark grey header with the word "Авторизация" in white text. Below the header, there are two input fields: the first is labeled "ЛОГИН:" and the second is labeled "ПАРОЛЬ:". Below the password field, there is a small red question mark icon and a red button with the text "ВОЙТИ" in white. The entire form is set against a light grey background.

Рис. 3.4. Форма для авторизации пользователей

Если же программа находит переменную *id* в массиве, то выполняется метод *menu* класса *html*, добавляющий дополнительное меню на страницу. Для каждого пользователя пункты меню отражают их функциональные возможности.

Необходимо упомянуть о методе *mesCount()*, который возвращает количество непрочитанных сообщений пользователя, хранимых в таблице *messages* базы данных *smart*.

В процессе работы системы могут возникнуть ошибки. О некоторых из них необходимо сообщать пользователю. Для этого служит метод *itsOk()*, который выводит сообщения об удачно или неудачно законченных процессах.

Для того чтобы разделить меню пользователей и смысловое содержимое страницы, используется функция *change_col()*. Для правильного завершения html верстки существует метод *footer()*.

3.3.2 ПОЛИТИКА БЕЗОПАСНОСТИ СИСТЕМЫ. ОРГАНИЗАЦИЯ СОЕДИНЕНИЯ С БАЗОЙ ДАННЫХ

База данных *smart* является основным хранилищем данных системы. Кроме структуры требуется продумать методы безопасного подключения к базе данных для каждого типа пользователя. В табл. 3.2 представлена структура политики безопасного доступа к таблицам *smart* с помощью ключей (r – возможность чтения данных таблицы, w – возможность добавления записей в таблицу, m – возможность редактирования записей).

Таблица 3.2

Права доступа к таблицам базы данных

Таблица базы данных	Права пользователей		
	Администратор	Преподаватель	Студент
users	rwm	rm	rm
messages	rw	rw	rw
groups	rwm	r	r
departaments	rwm	r	r
curses	r	rwm	r
curses_match	rwm		r

Продолжение таблицы 3.2

Таблица базы данных	Права пользователей		
	Администратор	Преподаватель	Студент
lectures	rwm	rwm	r
pages	rwm	rwm	r
questions		rwm	r
tests_result		r	r
news	rwm	r	r
tips	r	r	r

Доступ к таблицам устанавливается в настройках базы данных и зависит от используемого при соединении имени и пароля. Для разграничения прав пользователей требуется использовать класс *connect()*. При этом от входных параметров, заданных при создании экземпляра, зависят права доступа к базе.

3.3.3 ФУНКЦИОНАЛ ПОЛЬЗОВАТЕЛЯ

После прохождения регистрации у любого пользователя появляется доступ к дополнительному меню. У каждого типа пользователя есть доступ к личному кабинету, где он может получить информацию о своей учетной записи (аккаунте). Пример страницы пользователя приведен на рис. 3.5.

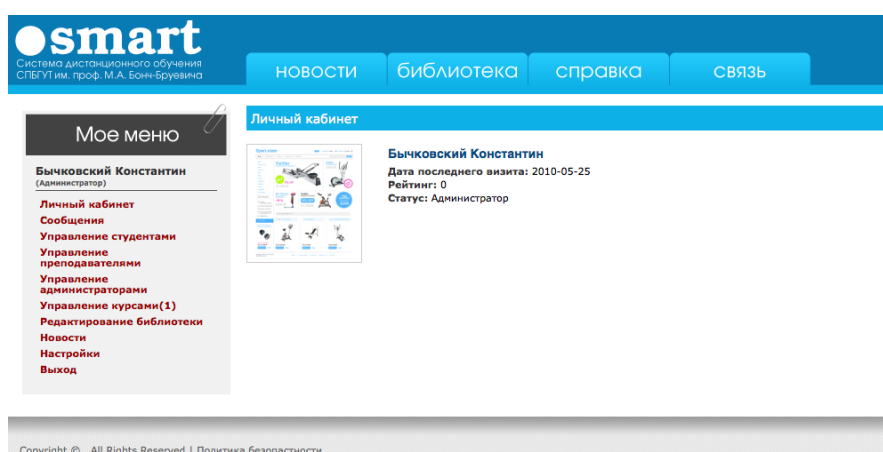


Рис. 3.5. Пример страницы пользователя

На этой странице представлена фотография пользователя, его фамилия и имя, статус, рейтинг и дата последнего визита. Для этого служит метод *getinfo()* класса *users*.

Еще одной возможностью, доступной для всех типов пользователей, является способность принимать и получать сообщения. Для этого требуется выбрать пункт меню «Сообщения». Количество непрочитанных сообщений показано рядом со ссылкой на страницу сообщений. Это сделано с помощью метода *mesCount()* класса *html*. Страница сообщений приведена на рис. 3.6.

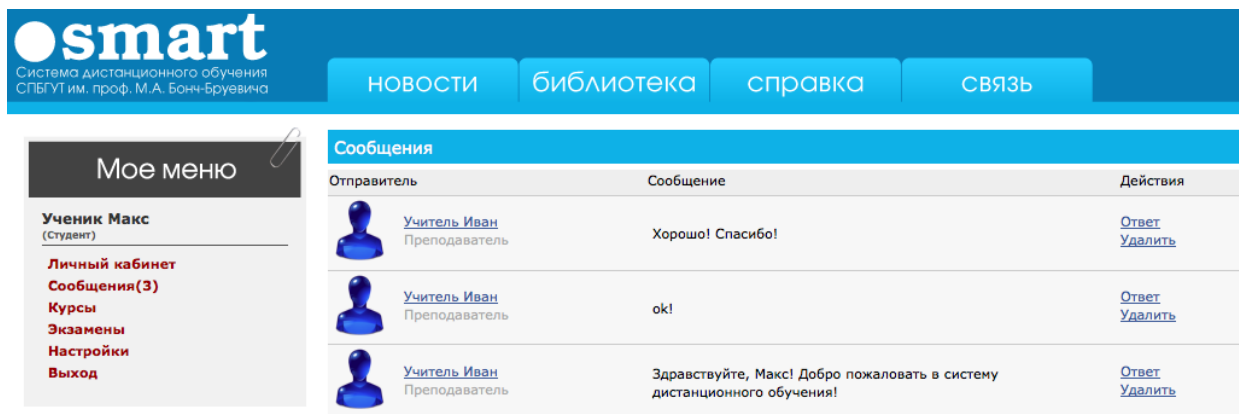


Рис. 3.6. Страница сообщений

На этой странице показана вся информация о сообщениях пользователя: имя и фамилия отправителя, текст сообщения. Так же осуществлена возможность управления. Чтобы ответить на сообщение, нужно выбрать соответствующую ссылку. После этого пользователь попадет на страницу адресата (рис. 3.7).

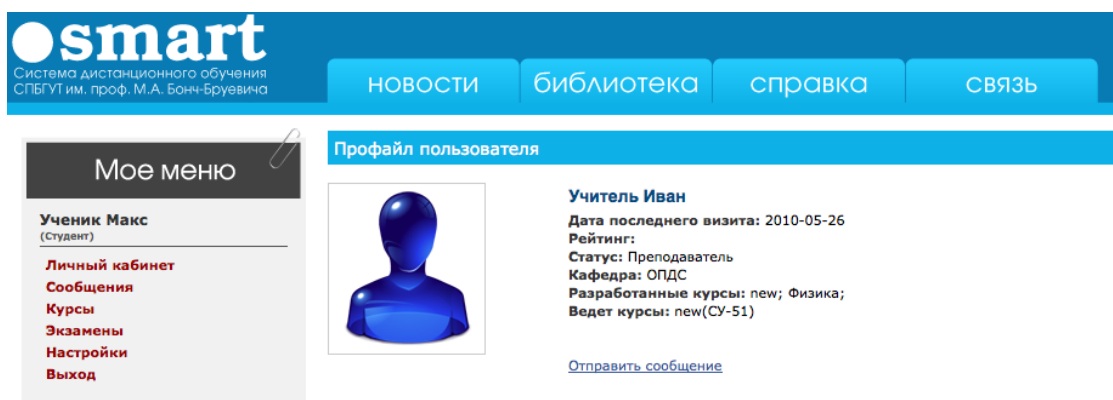


Рис. 3.7. Страница пользователя

Такая страница в социальных сетях носит название «профиль пользователя». Здесь можно получить всю информацию об отправителе, а так же ответить на его сообщение. Хотя профиль пользователя похож на личную страницу, он генерируется отдельным методом *getProfile()* класса *service*. Этот

класс доступен любому пользователю системы и служит для выполнения служебных функций, таких как получение информации о пользователе. При этом используется самый безопасный метод соединения с базой.

У каждого пользователя системы помимо имени, фамилии и прочей информации в профиле есть личная фотография, так называемый «аватар» или «юзерпик» (от англ. «userpic», «user picture» – изображение пользователя). Как видно из рис. 3.7 у пользователя с именем Иван фотография заменена на стандартное изображение. Что бы изменить это, требуется воспользоваться пунктом меню «Настройки» (рис. 3.8).

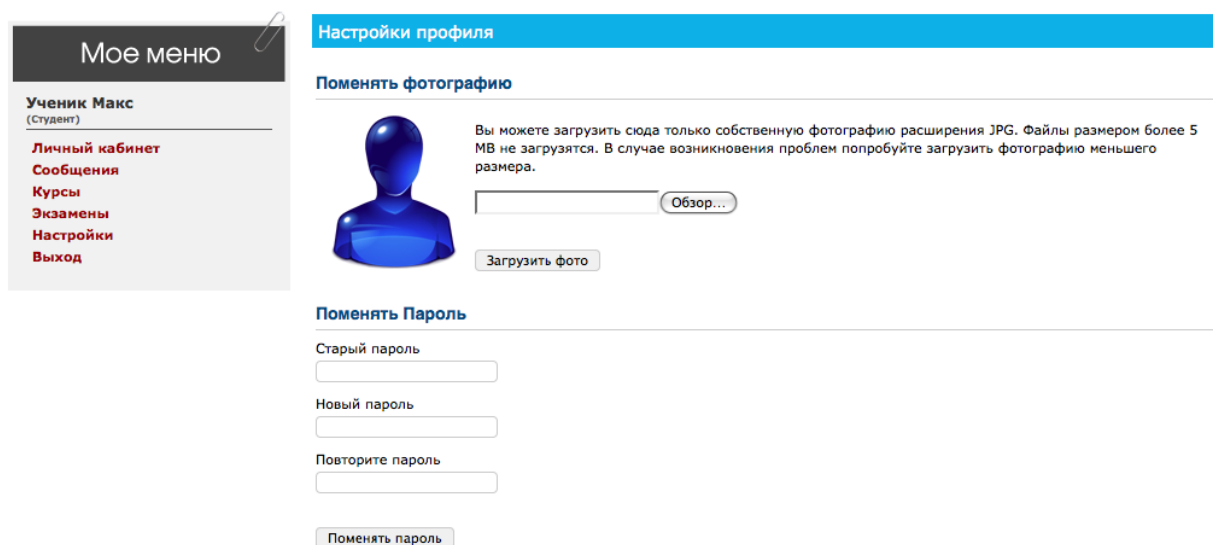


Рис. 3.8. Страница настроек пользователя

Так же на этой странице можно изменить пароль для входа в систему.

Сессия пользователя хранится на сервере постоянно. То есть, если пользователь прошел процесс авторизации в системе однажды, ему не придется это делать при каждом посещении сайта. Но существуют также ситуации, когда из-за соображения безопасности, необходимо завершить работу с системой. Для этого служит пункт меню «Выход», который осуществляет переход на страницу, где сценарий завершает сессию.

Для описания системы далее представлены функциональные возможности пользователей, выполняемые при организации процесса обучения в системе

«Smart», согласно описанной в п. 2.1 и изображенной на рис. 2.3 модели учебного процесса.

3.3.4 ФУНКЦИОНАЛ АДМИНИСТРАТОРА

Функции администратора осуществляются экземпляром класса *admin*. Одной из основных задач администратора является создание пользователей. Пользователи в системе, как упоминалось выше, делятся на три типа. Процессы добавления студентов, преподавателей и администраторов схожи, поэтому рассмотрим эту функцию на примере студентов. Для создания каждого типа у администратора есть отдельный пункт дополнительного меню (рис. 3.9).



Рис. 3.9. Меню администратора

Для создания студента системы нужно выбрать пункт «Управление студентами». Студенты в системе объединяются в группы. Следовательно, прежде чем создать пользователя такого типа, необходимо создать группу. Для этого на странице управления группами нужно нажать на соответствующую кнопку. Пример создания группы изображен на рис. 3.10.

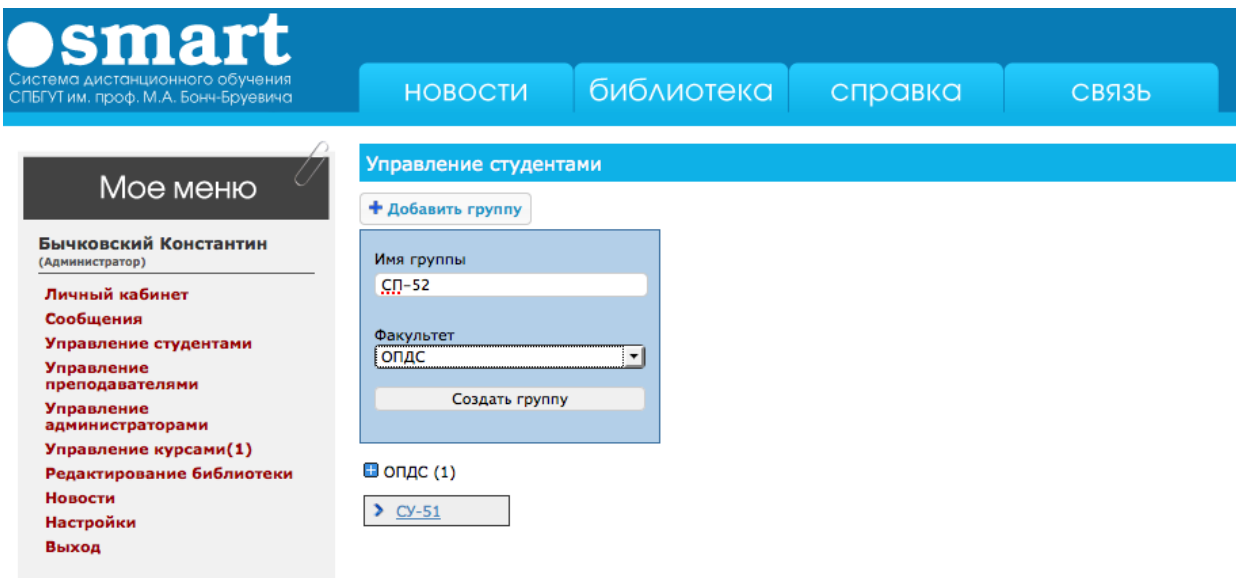


Рис. 3.10. Создание группы

При создании вводится имя группы и выбирается факультет, к которому она принадлежит. После создания группы возможно добавление в нее пользователей. Нажав на название, администратор откроет свиток информации о группе. В нем представлены все студенты этой группы. Так же здесь реализована возможность удаления группы и добавления новых пользователей. Если администратор решит удалить группу, то и студенты этой группы тоже удалятся из системы. При создании вводится имя и фамилия студента, его контактная электронная почта (рис. 3.11).

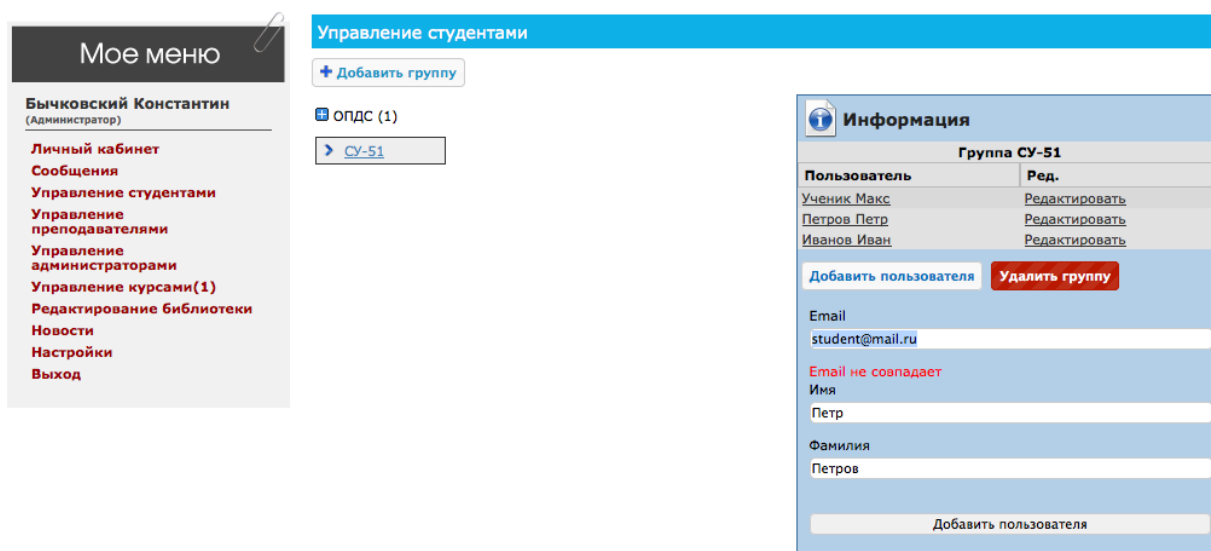


Рис 3.11. Процесс создания студентов

Следует отметить, что в процессе добавления требуется указывать адрес рабочей почты, так как после создания пользователя, на этот адрес система пошлет сообщение, в котором будет указан временный пароль для входа в систему. В случае неполадок с почтовым клиентом сервера, администратору предлагается лично сообщить пользователю необходимую информацию для входа в систему (рис. 3.12).

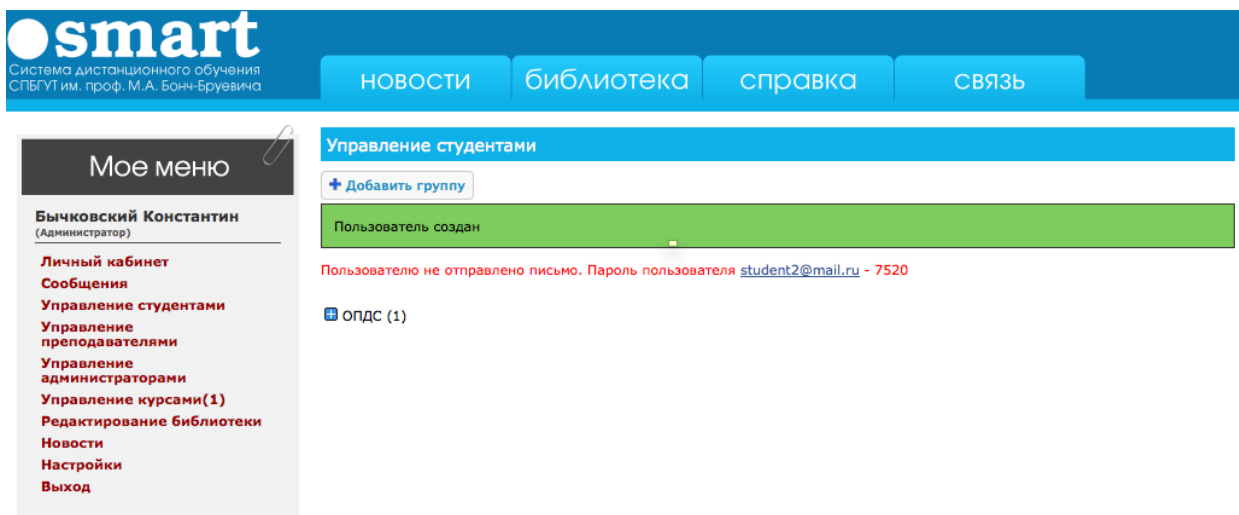


Рис. 3.12. Пример сообщения о созданном пользователе

Еще одной основной функцией администратора является добавление новостей. Просмотр новостей доступен каждому посетителю сайта. Любая новость состоит из заголовка, содержания и даты публикации (рис. 3.13).

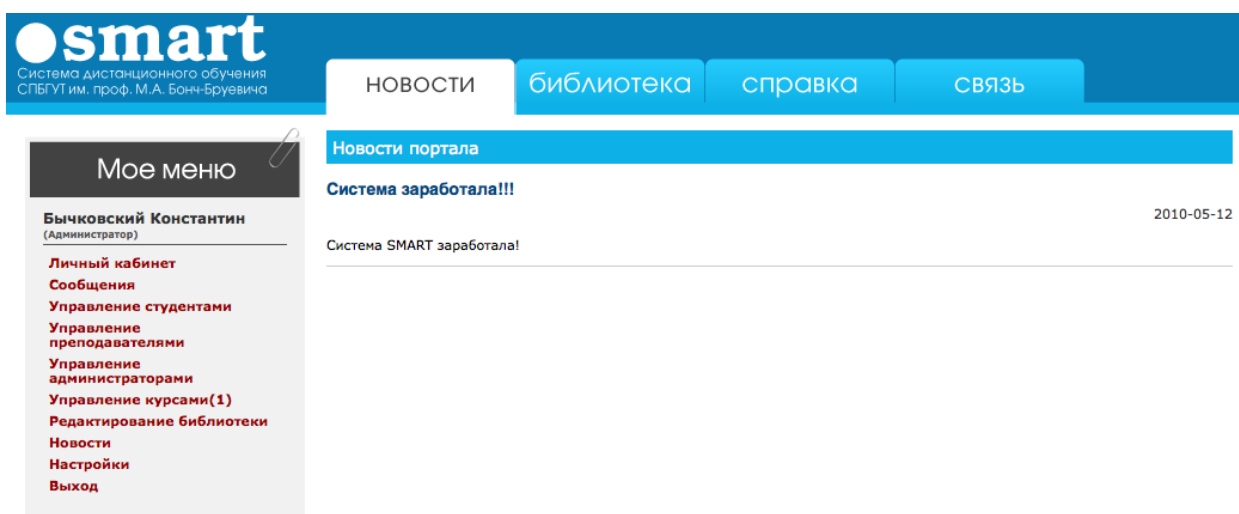


Рис. 3.13. Новости системы дистанционного обучения

Чтобы создать новость, администратор использует модуль новостей (пункт меню «Новости»). При редактировании новостей происходит асинхронная посылка http-запроса с помощью технологии AJAX. Такие запросы используются почти во всех модулях системы. Следует разобрать подробно возможности этой технологии на примере модуля редактирования новостей, чтобы далее не затрагивать этот вопрос. Рассмотрим этот процесс с помощью специального плагина Firebug браузера Mozilla Firefox. Он позволяет следить за сетевой активностью в процессе работы web-приложения.

Итак, администратор заходит на страницу редактирования новостей и нажимает кнопку «Новая новость». Указав заголовок и текст новости, пользователь нажимает кнопку «Добавить новость» (рис. 3.14).

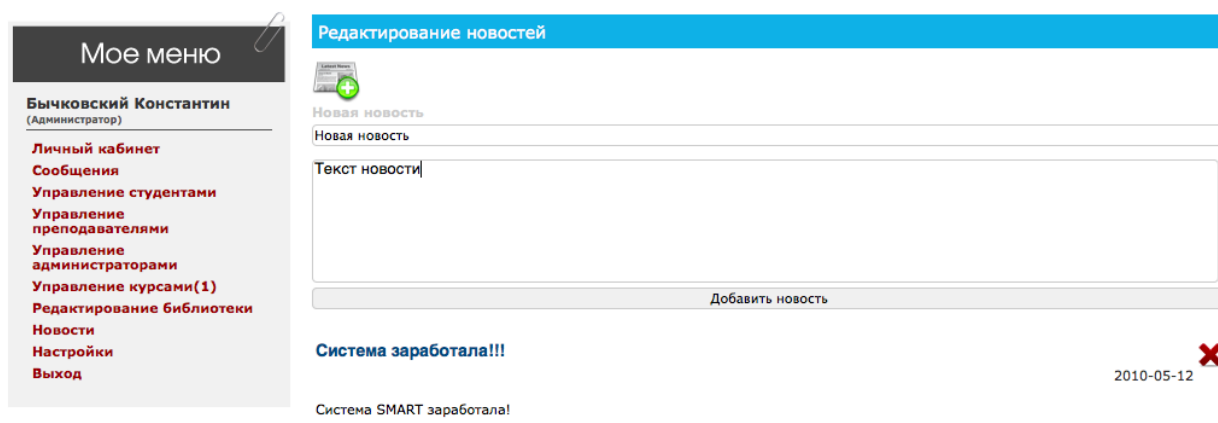


Рис. 3.14. Добавления новости

Функционал модуля сообщений осуществляется методом *getNewsToEdit()* класса *edit*. Этот метод генерирует на странице форму добавления новостей. При нажатии кнопки «Добавить новость» страница перезагружается, при этом отправляются данные формы на сервер. Сервер обрабатывает полученную информацию и создает новую запись в таблице *news* базы данных *smart*. Такой режим работы с серверов является наиболее распространенным в интернете и называется синхронным.

Если возникает необходимость изменить новость, администратору предлагается использовать интерактивный метод редактирования новостей. Для этого он должен просто нажать на элемент новости, который нужно исправить.

На рис. 3.15 представлена ситуация, когда администратор нажал на заголовок новости.



Рис. 3.15. Пример интерактивного изменения новостей

Теперь текст заголовка доступен для редактирования. После успешного исправления пользователь нажимает кнопку «ОК». Как видно на рис. 3.16 клиент пользователя послал http-запрос на сервер, сценарию *addToDB.php* без перезагрузки страницы.

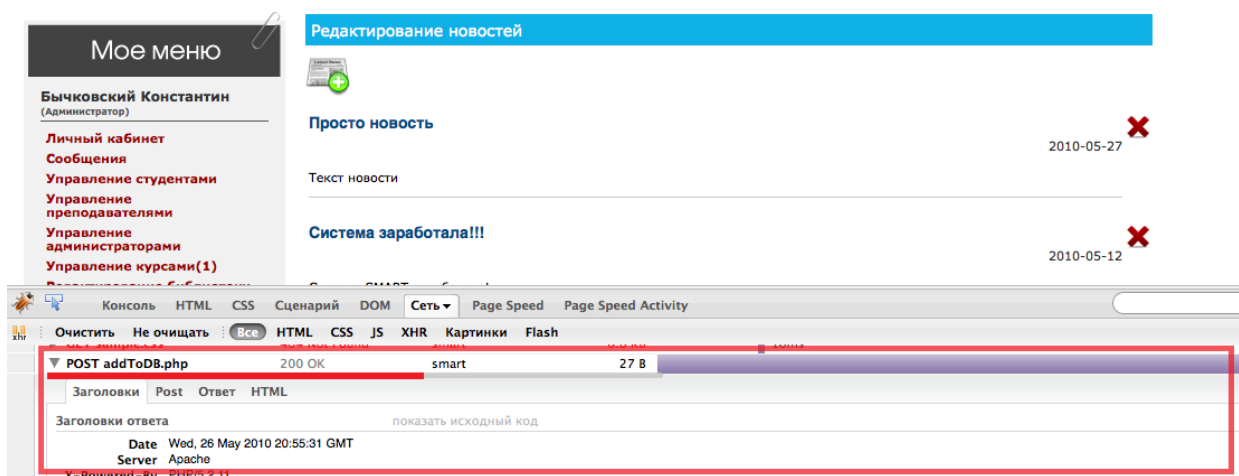


Рис. 3.16. Асинхронный запрос к серверу

Для полноты восприятия процесса рассмотрим непосредственно сам запрос:

```
value=Просто новость&id=h6
```

Как отлично видно из исходного кода запроса, на сервер послано две переменных: *value*, имеющая значение «Просто новость» и *id*, равная «h6».

Сценарий *addToDB.php* анализирует переменную *id* стандартной функцией *substr* интерпретатора ядра php. Первый символ в значении этой переменной означает, что требуется обновить заголовок новости, а второй обозначает

идентификатор поля новости в базе данных. После этого сценарий обновляет поле *header* в таблице *news*, в строке, где поле *id* равно 6. Далее отправляется ответ клиенту в виде текста «Просто новость». Именно этот текст с помощью функций языка *javascript* и заменяет элемент редактируемой новости.

Как видно, процесс асинхронного запроса прост и незаметен для пользователя, что добавляет интерактивность в работу над системой дистанционного обучения.

Последней функцией администратора является возможность администрирования курсов системы (рис. 3.17).

Курс	Автор	Свойства
new	Учитель Иван	свойства курса

Рис. 3.17. Администрирование курсов

На этом рисунке представлена вся информация о курсах, разработанных преподавателями. Любой курс после создания требует проверки. Заявки на утверждения курсов находятся наверху страницы. Администратор может посмотреть материалы курса, при необходимости исправить их и после проверки подтвердить его возможность использования в процессе обучения. Если у администратора нет времени на проверку, он может скрыть этот блок, нажав на ссылке «позже».

Чтобы назначить курс группе студентов, необходимо выбрать ссылку «Свойства курса» (рис. 3.18).

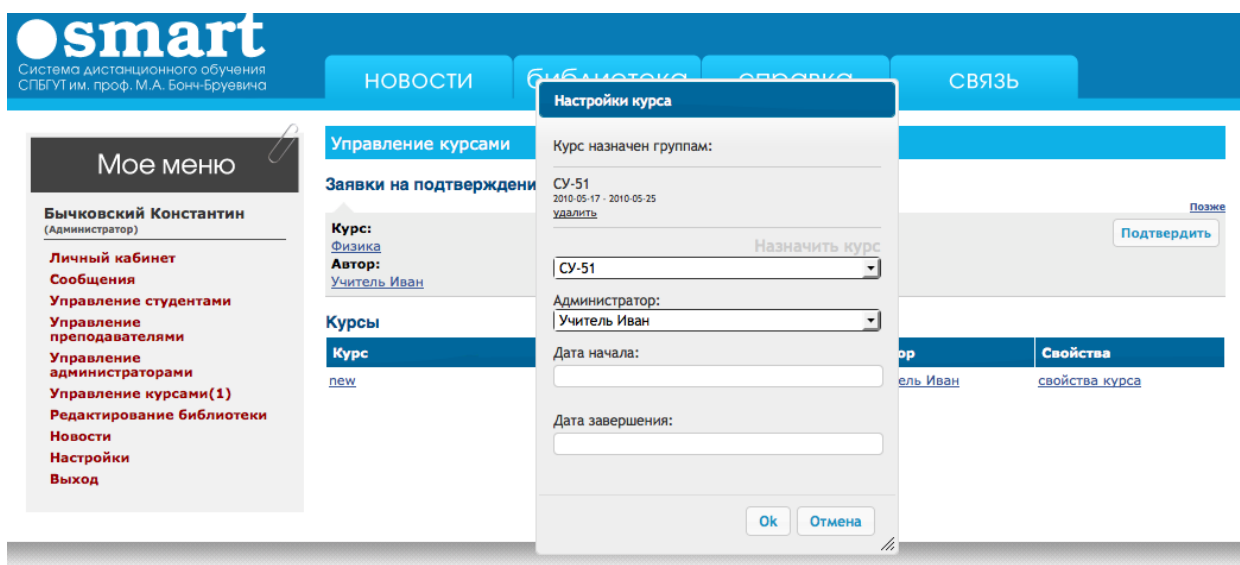


Рис. 3.18 Настройки курса

В появившемся окне можно ознакомиться с группами, изучающими этот курс, а так же назначить его новым группам.

3.3.5 ФУНКЦИОНАЛ ПРЕПОДАВАТЕЛЯ

3.3.5.1 СОЗДАНИЕ КУРСА

Важным шагом при организации обучения в среде СДО «Smart» является создание курсов. Это обязанность преподавателей. Для того чтобы приступить к разработке курса, необходимо выбрать пункт «Мои курсы» в меню преподавателя. Загруженная после этого страница позволяет управлять уже созданными курсами, а так же разрабатывать новые (рис. 3.19).

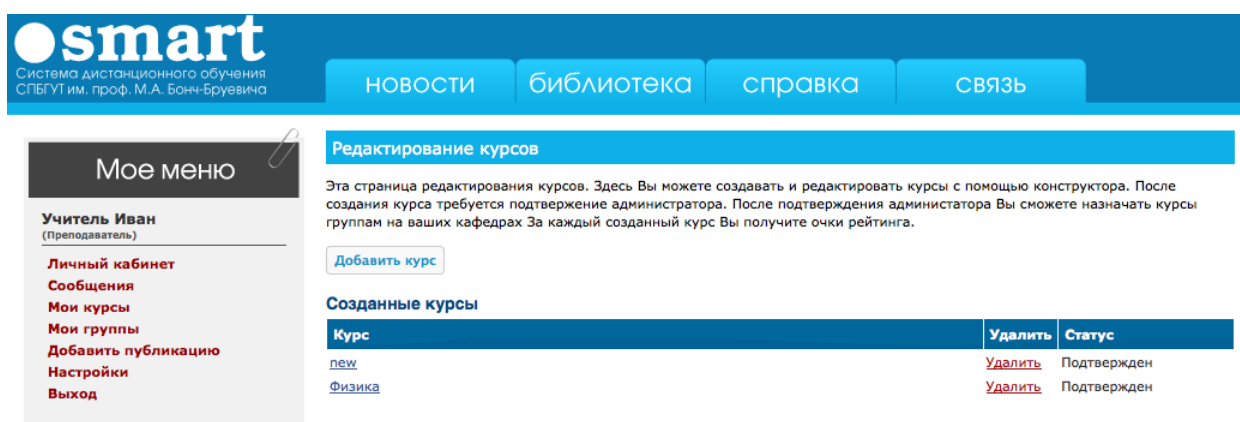


Рис. 3.19. Страница «Мои курсы»

При создании курса, прежде всего, требуется ввести название (рис. 3.20).

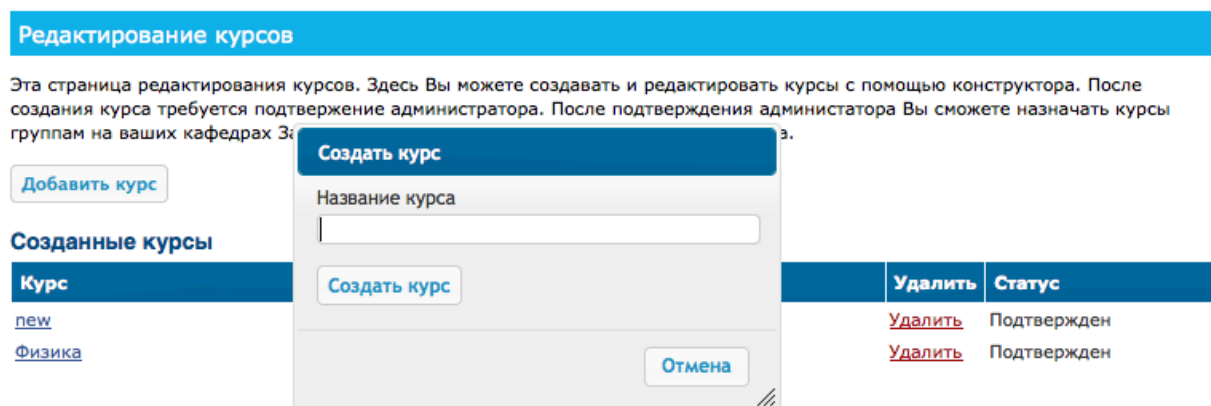


Рис. 3.20. Создание курса

После этого преподавателю необходимо выбрать курс и добавить учебный материал. Для этого служит модуль разработки курса, изображенный на рис. 3.20.

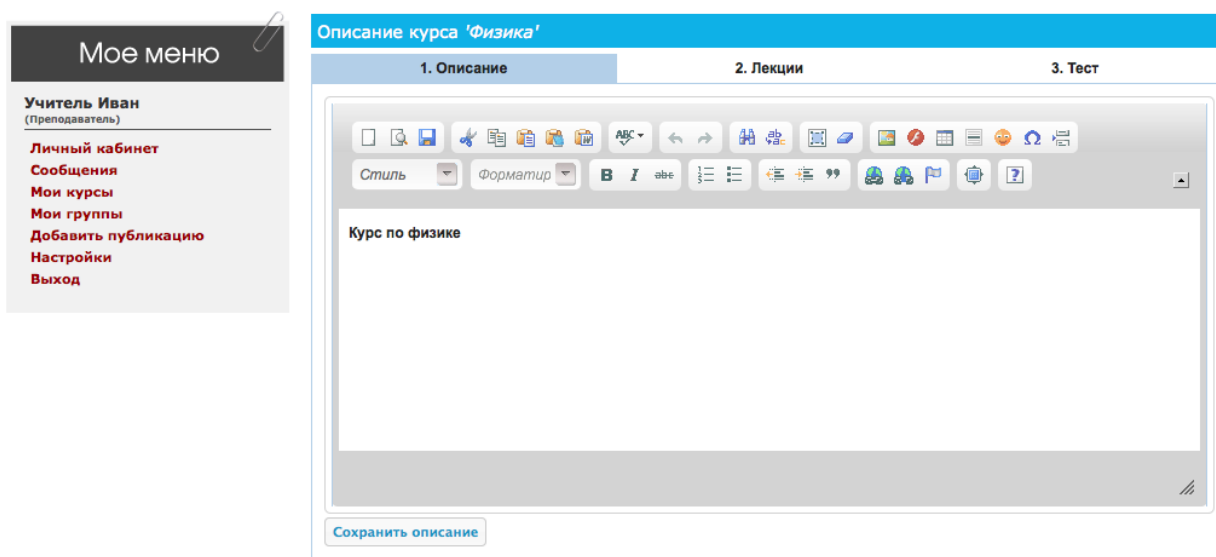


Рис. 3.20. Модуль разработки курса

Разработка курса состоит из трех частей. Сначала преподаватель добавляет описание курса, потом создает необходимое число лекций, после этого разрабатывает тестовое задание. Для создания учебного материала разработчику предлагается использовать встроенный редактор. Он имеет большой инструментарий для работы с текстом.

Важной функцией в работе текстового редактора является возможность добавления в содержимое лекций или описание курса собственных изображений. Для этого необходимо воспользоваться кнопкой «Свойства изображения» на панели управления редактором (рис. 3.21).

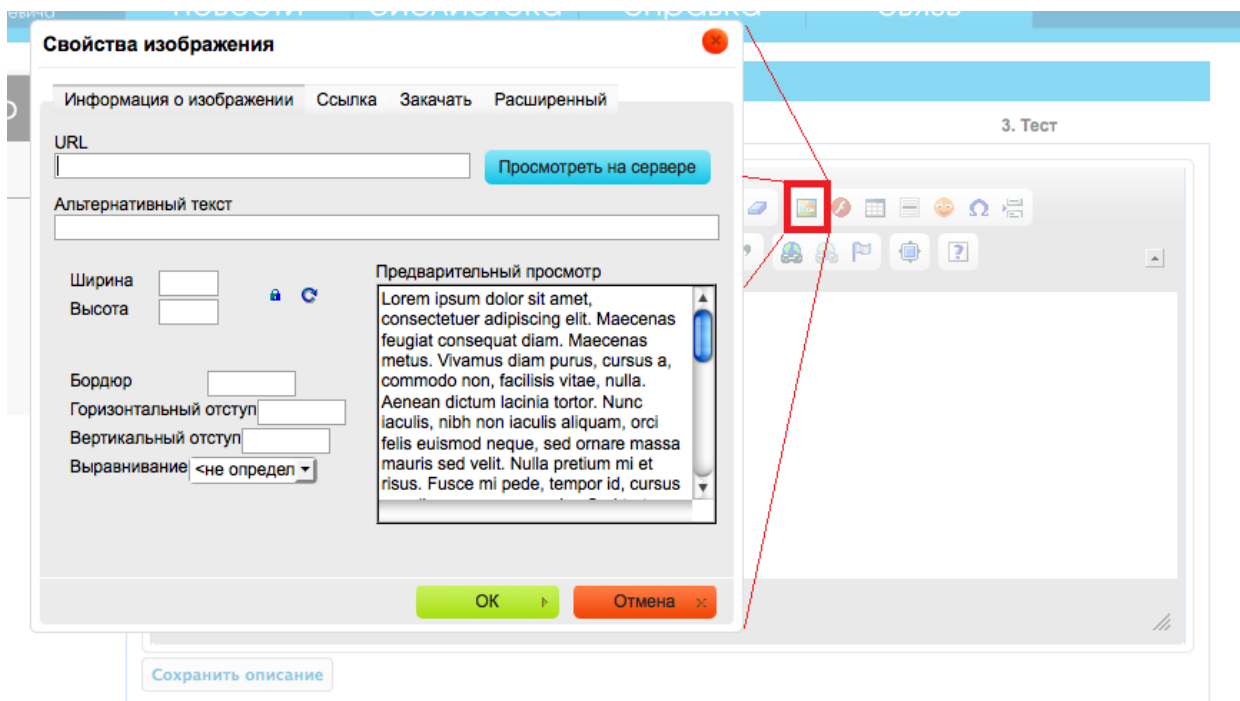


Рис. 3.21. Добавление изображений

Этот диалог предлагает указать адрес требуемого изображения на сервере. Если нужный файл не загружен, преподаватель может загрузить его, нажав на кнопку «Посмотреть на сервере». При создании курса используется метод *create()* класса *curse*. Помимо добавления в базу данных информации о новом курсе, этот метод создает папку на сервере системы с названием «*curseXX*», где *XX* – это идентификатор курса в таблице *curses*. Благодаря этому у каждого курса на сервере существует собственная папка, где создатель может хранить файлы, необходимые для изучения курса.

Встроенный загрузчик файлов автоматически покажет содержимое папки курса. Предлагается использовать три подпапки для хранения файлов разных типов: *image* для изображений, *flash* для хранения flash-клипов и папку *file* для других типов. Использование всех этих папок оставлено на усмотрение

создателей курса и не влияет на работоспособность системы. Пример загрузчика файлов показан на рис. 3.22.

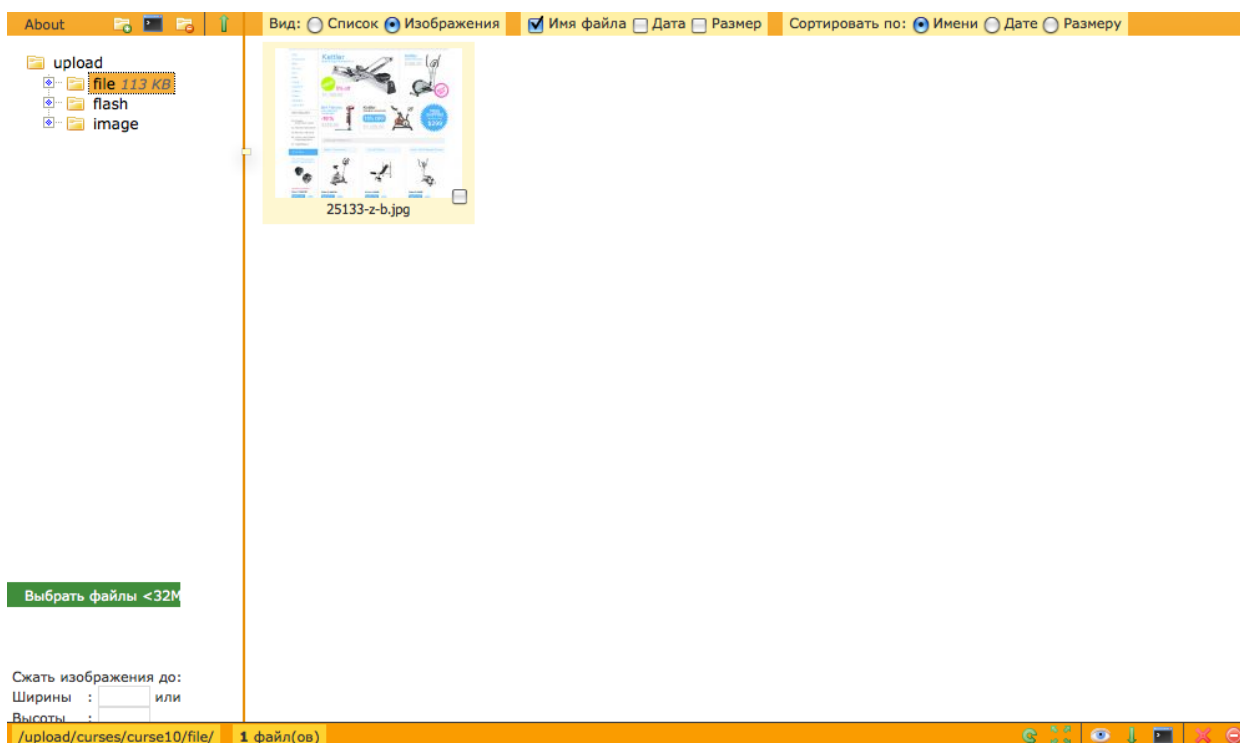


Рис. 3.22. Интерактивный загрузчик файлов

Создав описание курса, преподаватель переходит во вкладку лекции. Модуль создания лекции работает благодаря классу *lectures*. Это класс позволяет создать теоретический материал курса. Лекции в системе дистанционного обучения состоят из страниц. В таблице *lectures* хранится лишь описание лекции. Вся смысловая часть находится в таблице *pages*. Страница разработки лекций представлена на рис. 3.23.

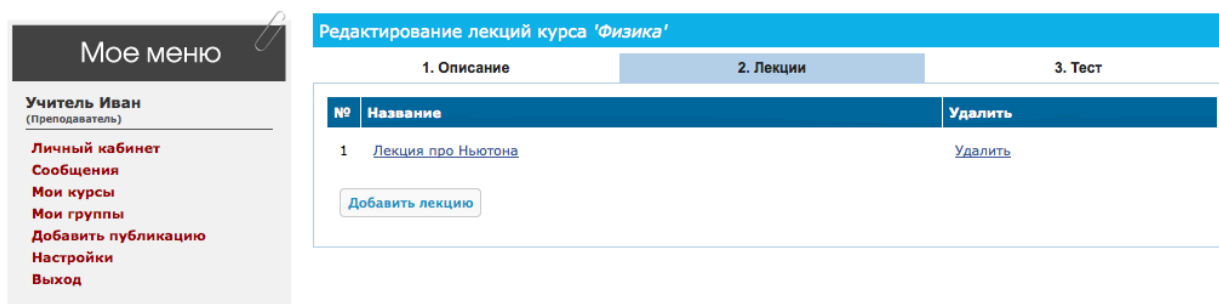


Рис. 3.23. Лекции курса

Лекций в курсе может быть сколько угодно. Чтобы создать лекцию, необходимо воспользоваться соответствующей кнопкой. Чтобы редактировать, нужно выбрать ссылку с названием лекции. Сам модуль редактирования лекций несколько отличается от стандартной страницы сайта. Это является следствием того, что во время редактирования лекций, создается дополнительная сессия. В глобальный массив *_session* добавляется переменная *course*, хранящая идентификатор курса и блокирующая возможность удаления курса при редактировании лекций. Дело в том, что таблица *pages* базы данных независима от других таблиц, благодаря чему, может использоваться при дальнейшей модернизации системы. Если при редактировании страницы удалить курс, тогда и удалятся все лекции этого курса и его страницы. Это приведет к тому, что если после этого сохранить редактируемую страницу, то она не будет связана ни с одной лекцией. А так как таблица *pages* хранит информацию о содержании всех лекций в системе, найти неиспользуемые строки будет довольно проблематично. После редактирования лекций, преподаватель должен нажать на кнопку «Закончить редактирование» в блоке информации страницы редактирования лекций, где так же представлена исчерпывающая информация о лекции (рис. 3.24).

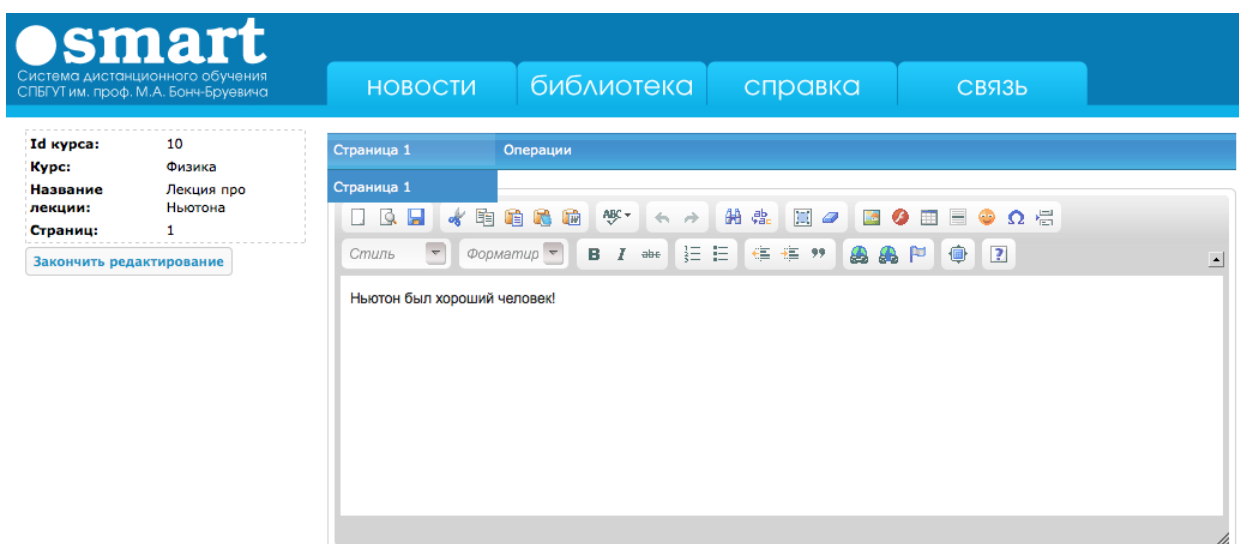


Рис. 3.24. Редактирование лекций

Страница редактирования лекций так же оснащена текстовым редактором. С помощью выпадающего меню преподаватель легко может просматривать все страницы лекций, а также их создавать и удалять. Весь функционал этого модуля организован с помощью экземпляра класса *lectures*.
Последним шаром при разработке курса является создание тестов. Модуль создания тестов находится во вкладке «Тесты» страницы редактирования курса (рис. 3.25).

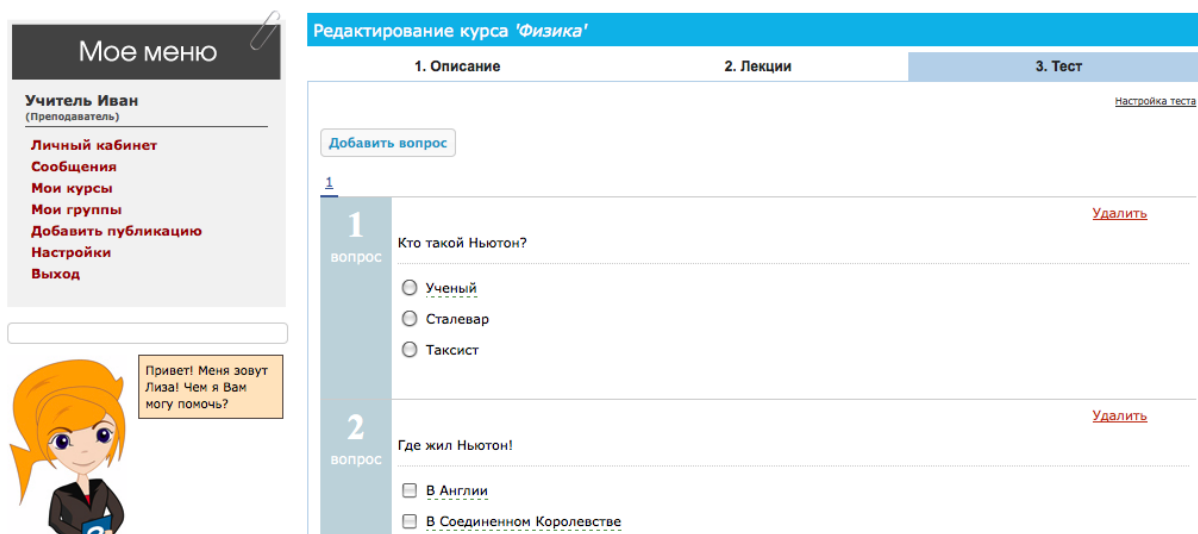


Рис. 3.25. Страница создания курса

Так как создание теста – процесс сложный, в системе организован, так называемый, помощник, который находится в левой части страницы, под пользовательским меню (рис. 3.26).

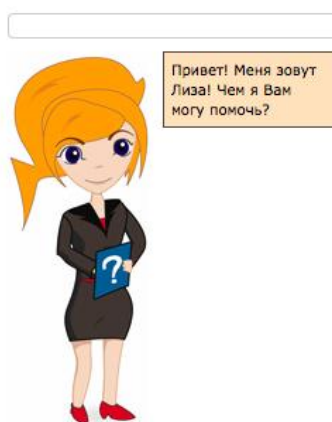


Рис. 3.26. Помощник создания курсов

Прежде всего, он служит для пояснения некоторых принципов создания тестов. Пользоваться им довольно легко. Нажав на изображение помощника, преподаватель получит случайный совет по созданию курсов. Если пользователь хочет получить справку по определенной теме, он вводит в расположенное над помощником поле поиска то, что ему не ясно в разработке теста. При этом набрав несколько первых символов, пользователь получает список наиболее подходящих тем (рис. 3.27).

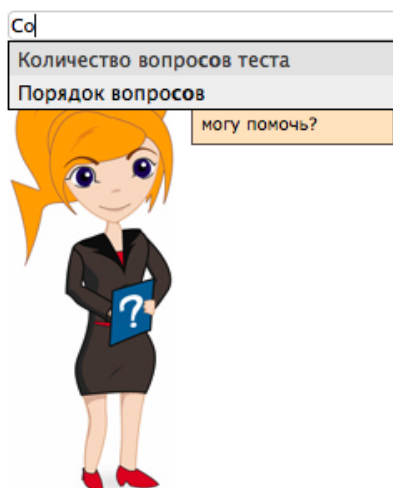


Рис. 3.27. Подсказки помощника

Когда страница получает несколько символов в поле ввода, она посылает их на сервер сценарию *hotTip.php*, который ищет в таблице *tips* базы данных наиболее подходящие строки, после чего возвращает их клиенту пользователя. Аналогично работает технология «Google Suggest» на сайте *google.com*, где после ввода символа в текстовое поле появляется выпадающий список, который содержит десяток строк со словами, начальные символы которых совпадают с теми символами, что посетитель уже успел ввести.

Благодаря помощнику, преподаватель может быстро разобраться в том, как создавать тестовое задание. Тест в системе «Smart» состоит из нескольких вопросов с вариантами ответов. Нажатие кнопки «Добавить вопрос» откроет форму для создания вопроса. Она состоит из поля ввода вопроса и полей вариантов ответа. Правильные ответы вводятся в первую очередь. Чтобы дать

понять системе, какие ответы верны, необходимо выбрать число правильных ответов в соответствующем выпадающем списке (рис. 3.27).

Добавить вопрос

Вопрос:
Кто такой Ньютон?

Поясняющее изображение:
Обзор...

Сперва вводите верные ответы, потом неверные. И не забудьте указать количество правильных ответов ниже.

Вариант 1:
Ученый

Вариант 2:
Политик

Количество правильных ответов(1-6): 1

[Добавить вариант](#)
[Убрать вариант](#)

Сохранить Сбросить форму

1

Рис. 3.27. Добавление вопроса

Вопрос может иметь до шести вариантов ответа, где хотя бы один верный. В базе данных варианты ответов хранятся в поле таблицы *questions* в виде строки и разделяются специальным символом «~». По этой причине использование этого символа при составлении вопроса недопустимо.

Дополнить вопрос можно, загрузив поясняющее изображение. При этом загруженный файл будет сохранен в соответствующей папке курса. Пример вопроса с картинкой представлен на рис. 3.28.

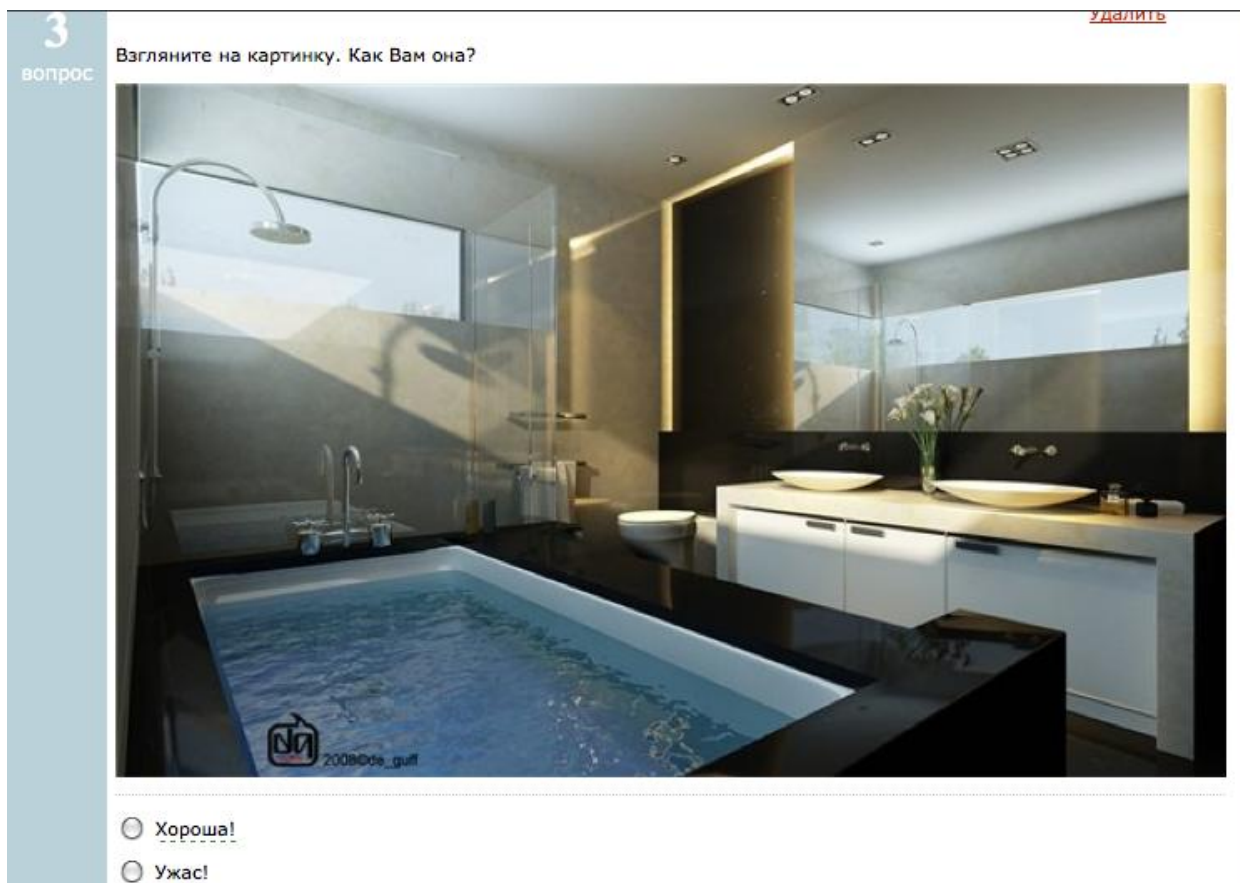


Рис. 3.28. Вопрос с поясняющим изображением

После успешного создания теста, процесс разработки курса заканчивается. При этом создатель курса может в любой момент исправить или дополнить учебные материалы.

3.3.5.2 АДМИНИСТРИРОВАНИЕ КУРСОВ

Помимо создания курса, еще одной немаловажной задачей преподавателя является администрирование вверенных ему курсов. После появления в системе нового курса, администратор назначает его группе студентов для изучения. При этом назначается и администратор курса из числа преподавателей. Эту должность в системе может занимать, как и создатель курса, так и любой другой преподаватель. При назначении указывается и сроки прохождения курса, в течение которых студенты должны изучить учебные материалы и пройти тестовое задание.

Все курсы, которые поручили администрировать преподавателю, находятся в пункте «Мои группы» пользовательского меню (рис. 3.29).

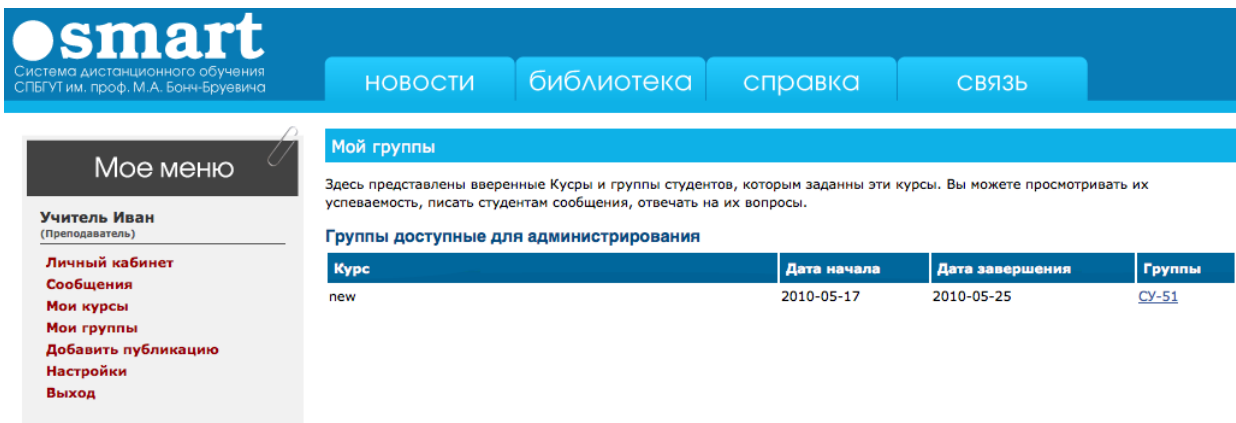


Рис. 3.29. Страница управления группами

На этой странице указаны название курса, сроки прохождения и, собственно, обучающаяся на этом курсе группа. Чтобы посмотреть успехи студентов группы в прохождении курса, необходимо нажать на название группы (рис. 3.30).

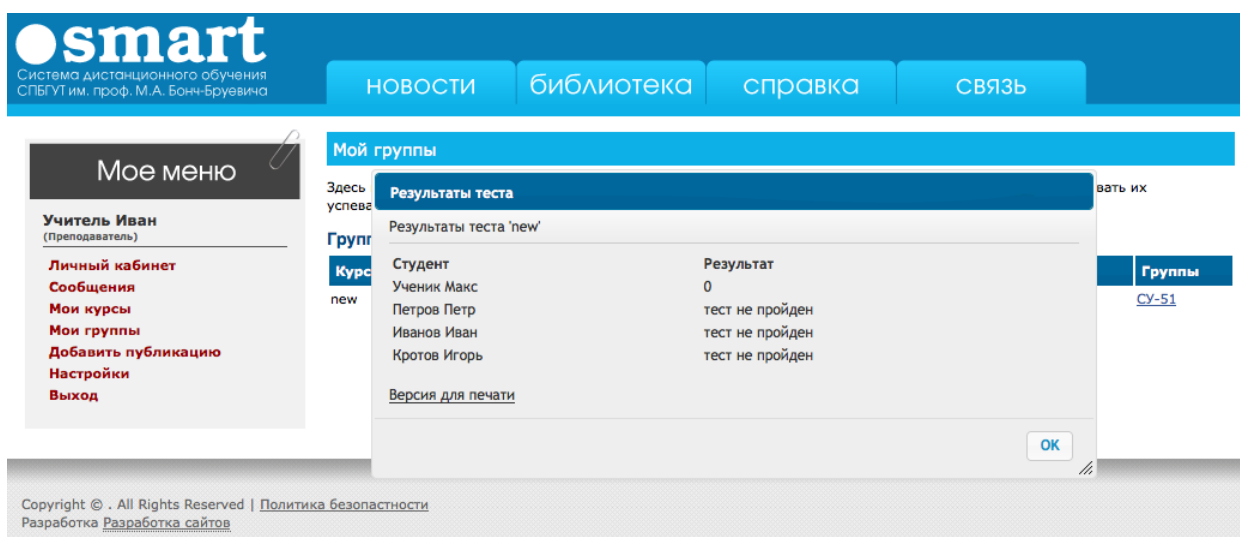


Рис. 3.30. Результаты группы

В появившемся при этом диалоге можно ознакомиться с результатами выполнения теста. Чтобы послать студенту сообщение, необходимо нажать на его имя. Для печати результатов можно использовать версию страницы, без лишнего графического интерфейса.

3.3.6 ФУНКЦИОНАЛ СТУДЕНТА

В системе дистанционного обучения «Smart» главной обязанностью студентов является изучение курсов. После того как курсы созданы

преподавателем и проверенны администратором, они назначаются группе студентов. Каждый студент системы может получить всю информацию о доступных ему курсах, перейдя по ссылке «Курсы» в пользовательском меню (рис. 3.31).

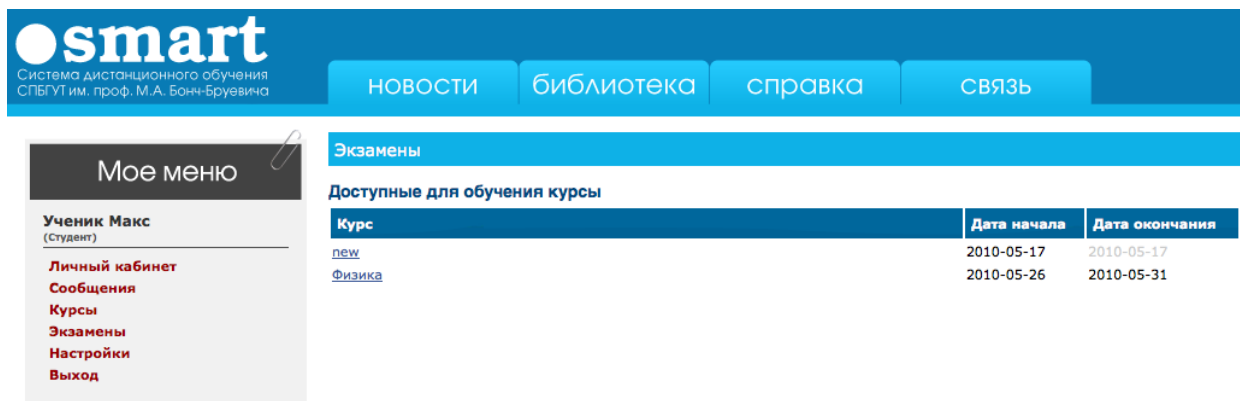


Рис. 3.31. Страница доступных для обучения курсов

Помимо уже доступных курсов, на этой странице студент может увидеть и курсы, которые будут доступны позже. Если курс еще не начался, дата начала в таблице будет выделена серым цветом. Чтобы прочитать лекции, студенту нужно нажать на название курса, который он собирается изучить (рис. 3.32).

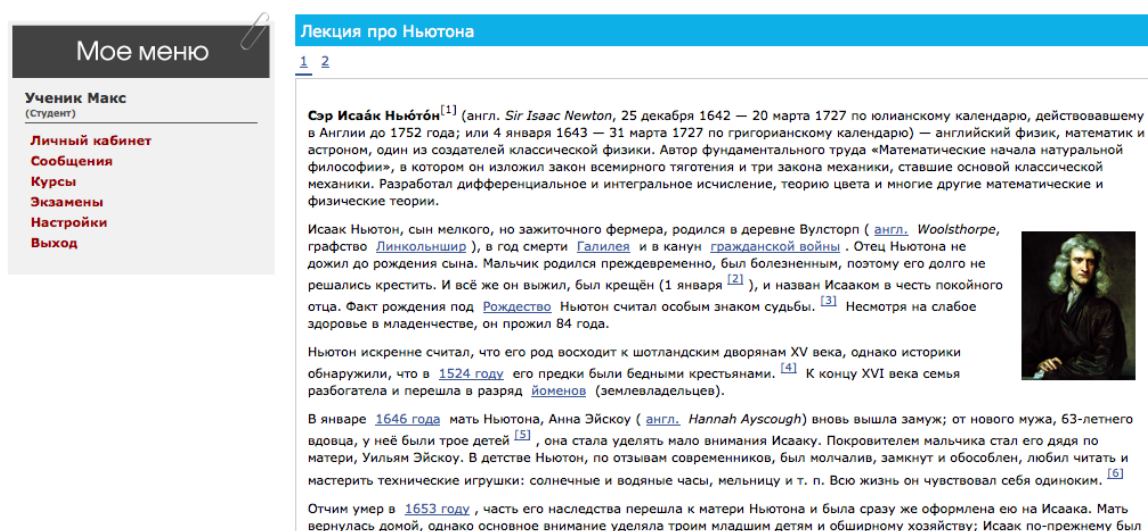


Рис. 3.32. Страница лекции

С помощью ссылок, расположенных над и под текстом, пользователь может переключаться между страницами лекции.

Если при прочтении лекции у студента возникает вопрос, ему необходимо выделить непонятный ему фрагмент текста и нажать клавишу «q». При этом

откроется диалог, с помощью которого студент может задать вопрос администратору курса (рис. 3.33).

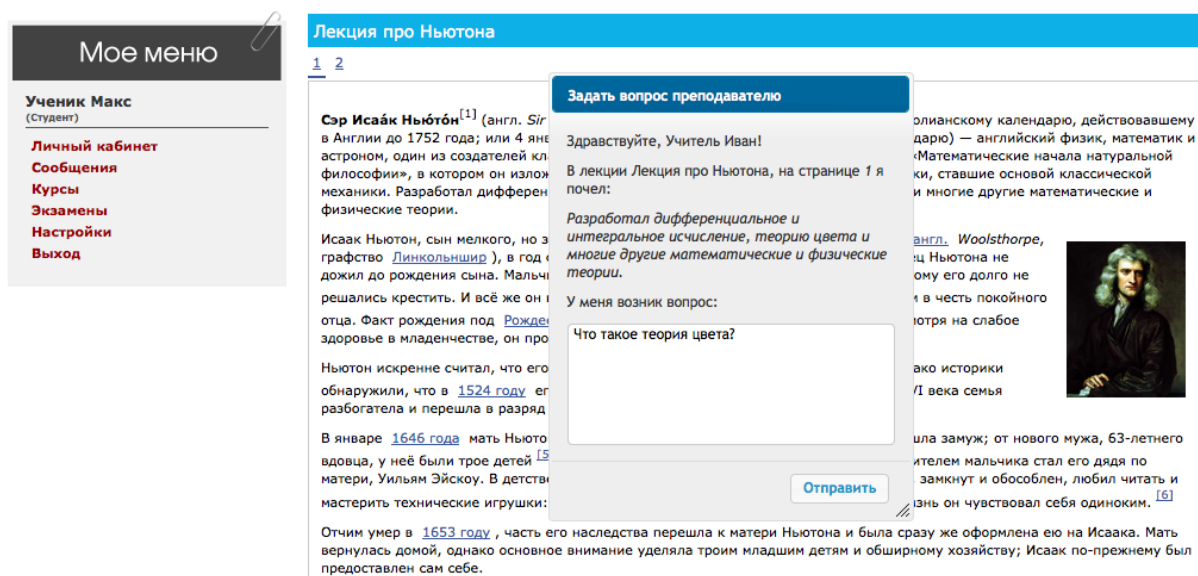


Рис. 3.33. Консультация с преподавателем

Текст сообщения преподавателю уже будет частично сгенерирован. Так же в сообщении будет добавлена выделенная часть текста. Студенту остается лишь задать вопрос преподавателю.

На рис. 3.31 представлена страница пользователя, у которого курс «new» уже закончился. Это видно по выделенной серым цветом дате окончания курса. Для того чтобы посмотреть выполнил ли студент тестовое задание по этому курсу, ему необходимо выбрать пункт «Экзамены» пользовательского меню. На этой странице представлены курсы, которые еще доступны для обучения, и результаты пройденных тестов. У каждого студента есть только одна попытка на сдачу теста. Если студент сдал тест досрочно, то материалы курса ему будут доступны до даты окончания. На рис. 3.34 представлена страница экзаменов пользователя.

The screenshot shows the 'smart' system interface. At the top, there is a navigation bar with 'новости', 'библиотека', 'справка', and 'связь'. On the left, a 'Мое меню' sidebar lists 'Ученик Макс (Студент)' and various options like 'Личный кабинет', 'Сообщения', 'Курсы', 'Экзамены', 'Настройки', and 'Выход'. The main content area is titled 'Экзамены' and contains two tables. The first table, 'Доступные тесты', lists a course named 'new' (Физика) with an examiner 'Учитель Иван' and exam dates '2010-05-25' and '2010-05-31'. The second table, 'Зачетная книжка', shows the result for the 'new' course as '15'.

Название курса	Экзаминатор	Дата сдачи
new Физика	Учитель Иван	2010-05-25
	Учитель Иван	2010-05-31

Название курса	Экзаминатор	Дата сдачи	Результат
new	Учитель Иван	2010-05-17	15

Рис. 3.34. Страница доступных для прохождения курсов и результаты пройденных тестов

Если студент считает, что достаточно изучил учебные материалы, то, чтобы пройти итоговый тест, он должен выбрать ссылку с названием нужного курса (рис. 3.35).

The screenshot shows the 'smart' system interface during a test. The top navigation bar and sidebar are the same as in the previous image. The main content area is titled 'Тест по курсу 'Физика'' and shows a timer at '059:45'. A question is displayed: '1 вопрос: Кто такой Ньютон?'. There are three radio button options: 'Ученый', 'Таксист', and 'Сталеваар'. A 'Далее >>' button is located at the bottom of the question area.

Рис. 3.35. Прохождение теста

На прохождение теста студенту отводится один час. Чтобы избежать неправомерных действий со стороны студента, таймер синхронизируется с сервером каждую секунду. Чтобы ответить на вопрос, пользователь должен нажать кнопку «Далее». При этом страница не перегружается, а выполняется функция языка *javascript*, которая посылает ответ на сторону сервера, который в свою очередь обрабатывает полученную информацию и возвращает новый вопрос. Если студент ответил на все вопросы, сценарий возвращает результат теста. Чтобы избежать потерь данных в связи с возможными неполадками при

каждом ответе студента на вопрос, результаты теста обновляются. На рис. 3.36 изображена страница результата пройденного теста.

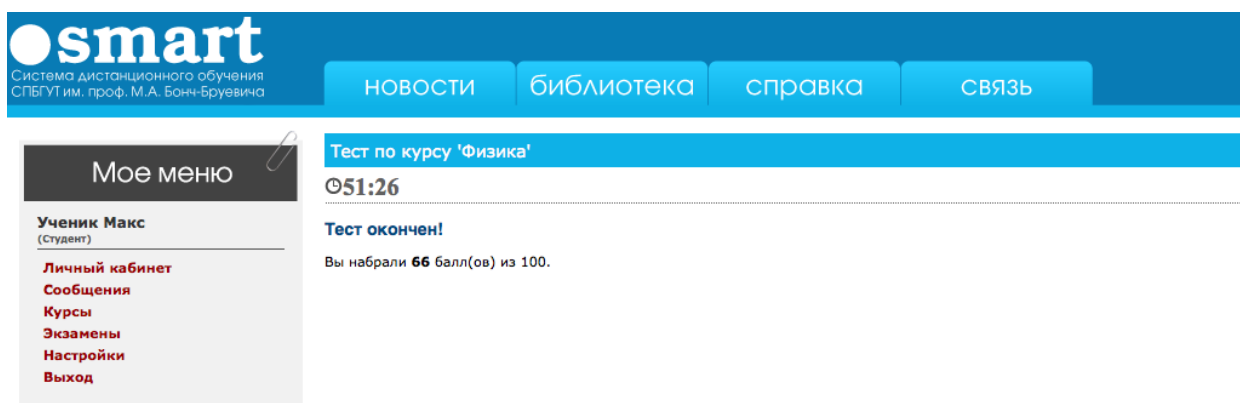


Рис. 3.36. Результат пройденного теста

Модуль выполнения тестов выставляет оценку студентам после ответа на каждый вопрос, руководствуясь следующей формулой:

$R = \bar{R} + a \cdot (1/S) \cdot 100$, где \bar{R} – результат пройденной части теста, a – флаг правильности ответа (1 или 0), S – общее количество вопросов.

Если студент не успеет пройти тест за 60 минут, результат пройденной части теста все равно будет записан в базу данных.

3.4 ОРГАНИЗАЦИЯ БЕЗОПАСНОСТИ В СИСТЕМЕ ДИСТАНЦИОННОГО ОБУЧЕНИЯ

Чтобы систему дистанционного обучения можно было использовать в организации учебной работы государственного образовательного учреждения, требуется поддерживать достаточный уровень безопасности этой системы. Для этого система должна обладать функциями защиты против основных методов сетевых атак.

СДО «Smart» представляет собой набор программных средств, взаимодействующих с пользователем. Большая часть данных хранится в базе MySQL. Самой распространенной сетевой атакой на сайты, использующие SQL-сервера, является, так называемая, SQL-инъекция. Дело в том, что php-сценарии работают с базой данных с помощью языка SQL. Для того, чтобы отправить

серверу SQL команду, программист использует метод *query* класса *mysqli*. Запрос к базе данных выглядит следующим образом:

```
mysqli->query("SELECT * FROM users");
```

В запросе так же могут использоваться данные, введенные пользователем. Основной проблемой является то, что пользователь может ввести все, что угодно, вплоть до SQL-команды, что может привести к недееспособности базы данных. Для того чтобы предотвратить атаки такого типа, вся информация, введенная пользователем, обрабатывается с помощью функции *mysql_escape_string()*, которая предотвращает возможность SQL-инъекции.

Еще одним вопросом безопасности в системе «Smart» является организация доступа к модулям системы. Согласно политике безопасности, такие страницы, как создание и редактирования пользователей, не могут быть доступны студентам. Но если пользователь знает адрес такой страницы, то теоретически он может получить доступ к данным возможностям системы. Чтобы предотвратить несанкционированный доступ, на каждой защищенной странице подключен модуль *check_status_teacher.php* (для страниц, доступных только администраторам и преподавателям) или *check_status_admin.php* (для страниц, доступных только администраторам). Эти web-приложения с помощью сессии получают идентификатор пользователя и проверяют права доступа к данной части сайта. Если возможность доступа не подтверждена, пользователю выводится сообщение об ошибке (рис. 3. 37).

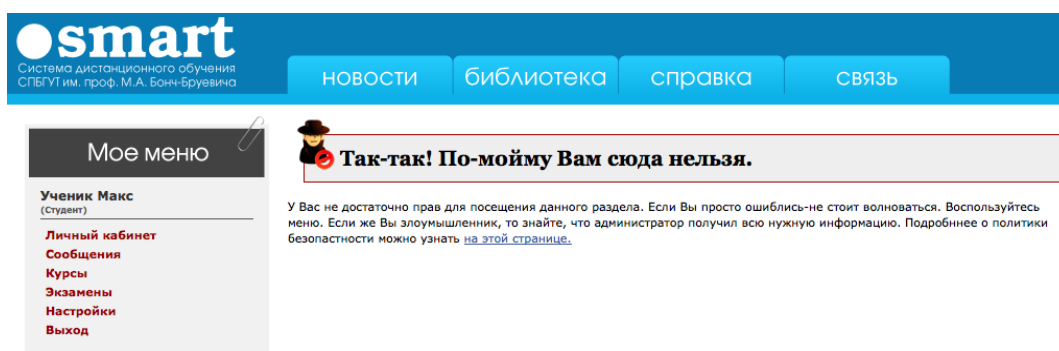


Рис. 3.37. Сообщение о несанкционированном доступе

Чтобы выявить злоумышленника, система «Smart» при каждой попытке доступа к защищенным частям записывает информацию о пользователе в специальный файл `sec_log`. Пример такого файла приведен ниже.

```
06-Apr-2010 00:13:38 : Access denied(UZ): from '/person.php', Userid '4', Ip '127.0.0.1'.
```

```
22-Apr-2010 20:46:42 : Access denied(UZ): from '/teacher/lectures.php', Userid '', Ip '127.0.0.1'.
```

```
27-May-2010 23:36:05 : Access denied(UZ): from '/teacher/courses.php', Userid '3', Ip '127.0.0.1'.
```

В этом файле указаны дата и время потенциальной атаки, ее тип и ip-адрес злоумышленника. Так же, если пользователь был зарегистрирован, указывается его идентификатор в системе. С помощью этой информации можно легко вычислить угрозу и совершить успешные действия для борьбы с ней.

Исходя из указанных выше методов борьбы с сетевыми атаками, можно сделать вывод, что система выполняет поставленные перед ней требования к безопасности.

В данной работе не рассматриваются вопросы безопасности, связанные с сервером, на котором расположена система.

ЗАКЛЮЧЕНИЕ

Целями данной работы являлись анализ возможности использования современных технологий в образовательном процессе высшего учебного заведения и разработка платформы для создания и поддержки дистанционного обучения в Санкт-Петербургском государственном университете телекоммуникаций им. М.А. Бонч-Бруевича.

Рассмотрев возможности применения современных технологий для обучения студентов, можно сделать вывод, что глобальная сеть Интернет является наиболее подходящей коммуникационной средой для организации системы дистанционного обучения. При этом созданный сайт в полной мере иллюстрирует основные возможности применения дистанционного обучения.

С технической точки зрения внедрение системы не требует больших экономических затрат и может быть развернуто на базе сайта СПбГУТ им. М.А. Бонч-Бруевича. Объектно-ориентированный подход, использованный в процессе разработки, позволяет осуществлять дальнейшую модернизацию системы.

Следует отметить, что созданная система дистанционного обучения является начальным этапом по внедрению современных технологий в процессы обучения. При дальнейшей работе необходимо учитывать конкретные требования образования высшего учебного заведения. В частности возможно внедрение технологий видеосвязи, как следующий этап разработки системы дистанционного обучения.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Книги:

1. Веллинг Л., Томсон Л. Разработка Web-приложений с помощью PHP и MySQL, 2009.
2. Бенкен Е.С., Самков Г.А., AJAX: Программирование для Интернет, 2009.

Интернет:

vesti.ru/doc.html?id=358518

adobe.com/products/acrobatconnectpro/?promoid=BPDEA

wikipedia.org

wikipedia.org/Wiki/GPL

code.google.com/p/gp/3rus/wiki/LatestRelease

javascript.crockford.com/popular.html

ru.wikipedia.org/Wiki/Javascript

nbmarketing.ru/innovations

demphest.ru/ajex-filemanager

ckeditor.com