

Министерство образования и науки РФ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Уральский государственный педагогический университет»
Институт математики, информатики и информационных технологий
Кафедра информационно-коммуникационных технологий в образовании

ОРГАНИЗАЦИЯ БАЗЫ ДАННЫХ НА ИНТЕРНЕТ ХОСТИНГЕ ДЛЯ ОБЕСПЕЧЕНИЯ СТУДЕНТОВ МАТЕРИАЛАМИ ДЛЯ УЧЕБНОГО ПРОЦЕССА

*Выпускная квалификационная работа
бакалавра по направлению подготовки
09.03.02 – Информационные системы и технологии*

Исполнитель: студент группы БС-41
Института математики, информатики и ИТ
Шитиков В.Е.

Руководитель: к.п.н., доцент кафедры ИКТО
Кудрявцев А.В.

Работа допущена к защите
« ____ » _____ 2017 г.
Зав. кафедрой _____

Екатеринбург – 2017

Реферат

Шитиков В.Е.ОРГАНИЗАЦИЯ БАЗЫ ДАННЫХ НА ИНТЕРНЕТ ХОСТИНГЕ ДЛЯ ОБЕСПЕЧЕНИЯ СТУДЕНТОВ МАТЕРИАЛАМИ ДЛЯ УЧЕБНОГО ПРОЦЕССА, выпускная квалификационная работа: 55 стр., рис. 21, табл. 4, библиограф. 20 назв., приложений 3.

Ключевые слова: БАЗА ДАННЫХ, ИНТЕРНЕТ ХОСТИНГ, СОЗДАНИЕ БАЗЫ ДАННЫХ, СОЗДАНИЕ САЙТА.

Объект разработки – база данных для хранения URL-адресов учебных материалов, сайт для предоставления информации.

Цель работы – сайт, предоставляющий возможность просмотра лекционных и лабораторных материалов, а также размещение новых преподавателями.

Данный сайт служит для взаимодействия с базой данных, где хранятся ссылки на документы и файлы для учебного процесса. Ученики могут просматривать лекции и задания по изучаемым предметам в любой точке планеты. Преподаватели могут выкладывать данные для изучения учениками.

В ходе работы выявлена проблема предоставления информации учащимся, проанализированы способы обеспечения учеников и студентов материалами для учебного процесса. Разработано техническое задание. Реализована клиентская и серверная часть с помощью СУБД MySQL, языка программирования PHP 4.6.4, языков разметки HTML5 и CSS3, CASE-средством DBDesigner 4. Вспомогательные приложения : текстовый редактор Notepad++.

Оглавление

ВВЕДЕНИЕ	4
ГЛАВА 1. ТЕОРЕТИКО-АНАЛИТИЧЕСКАЯ ЧАСТЬ	5
1.1 АКТУАЛЬНОСТЬ СОЗДАНИЯ СОБСТВЕННОГО ПРОДУКТА.	5
1.2 АНАЛИЗ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ РЕАЛИЗАЦИИ.	11
1.3 ТЕХНИЧЕСКОЕ ЗАДАНИЕ.....	22
ГЛАВА 2. ПРАКТИЧЕСКАЯ (ОРИГИНАЛЬНАЯ) ЧАСТЬ	28
2.1 РАЗРАБОТКА САЙТА И БАЗЫ ДАННЫХ.	28
2.2 ОПИСАНИЕ ПРОДУКТА (РЕЗУЛЬТАТА РАЗРАБОТКИ).....	38
ЗАКЛЮЧЕНИЕ	43
СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ.....	45
ПРИЛОЖЕНИЯ	48
Приложение 1.	48
Приложение 2.....	50

Введение

Каждый преподаватель, как правило, хранит информацию в удобной для себя форме на каком-то своем носителе: локальном или сетевом диске, FTP-сервере или в облачном пространстве. При этом часто возникает проблема передачи информации студентам, особенно если они находятся в других аудиториях, корпусах или вне учебного заведения. Иногда возникает ситуация, когда преподавателю вуза необходимо передать материал лекций или лабораторных работ своим студентам, при этом информация может быть расположена на разных средствах хранения. Иногда приходится тратить много времени, что бы как минимум ее собрать в одном месте, а позже передать ее каждому своему студенту.

Одним из решений этой проблемы является создание базы данных, которая должна стать системой авторизованного доступа к информационным ресурсам единой образовательной среды вуза. Следовательно создание такой базы является актуальным направлением информатизации образования.

Цель: организовать базу данных для обеспечения студентов информацией, а так же создать сайт, предоставляющий доступ к данной базе.

Задачи:

- провести анализ материалов предметной области;
- обосновать выбор инструментов для разработки базы данных и программных средств;
- создать БД для хранения информации и реализовать возможность регистрации новых пользователей и предоставление им доступа к данным;
- разработать web-интерфейс для организации доступа к данным;
- протестировать и внедрить разработанную систему в учебный процесс.

Предмет разработки: база данных для хранения URL-адресов учебных материалов, сайт для предоставления информации.

Глава 1. Теоретико-аналитическая часть

1.1 Актуальность создания собственного продукта.

По данным исследования методов информатизации, имеется недостаток способов передачи и объединения данных информационных систем, имеющихся в большинстве ВУЗов. Работники образовательных учреждений нередко сталкиваются с проблемами оперативности получения - передачи информации, решая которые теряется актуальность этой информации. Зачастую прибегая к различного рода сервисам для публикации информации, возникает так же сложность правильного оформления для передачи или сложность конвертации материала, для корректного распознавания данных используемой системой. Что создаёт дополнительные трудности, толкая пользователя на затраты времени поиска решения этих трудностей, нежели сосредоточения на целевом материале и своевременного «донесения» информации до конечного потребителя - студента. Дополнительным недостатком можно считать ограничения различных систем, которые не могут позволить передать требуемый объём информации в её исходном виде. Что в итоге искажает или вообще, изменяет смысл материала, передаваемого студенту. Последствия таких изменений могут быть самые разнообразные, вплоть до полного неусвоения материала, непонимания студентом цели и требований. Идентификация пользователей на различных ресурсах служит очередным барьером между последними и оперативным доступом. Наложение различных политик безопасности доступа и требования правил авторизации, усложняет доступ и пользование как своими ресурсами так и пользование сервисом в целом. При многочисленных требованиях систем хранения и передачи данных возникает необходимость объединения их в одну централизованную базу данных. Использование единой среды решает в том числе проблему унификации технологий, применяемых в преподавании многих дисциплин [Ошибка! Источник ссылки не найден.].

Основные способы передачи информации от преподавателя студентам:

- **Облачные хранилища;**

- **Социальные сети;**
- **Локальный диск;**

Рассмотрим каждый подробнее.

Облачные диски. Облачное хранилище данных — это "виртуальный носитель" информации, на котором можно хранить, обрабатывать, делиться документами и файлами в интернете.

Облако — набор аппаратного и программного обеспечения, который обеспечивает обработку и исполнение клиентских заявок [1].

С каждым годом того рода "хранилищ" появляется все больше, и на сегодняшний день их количество равняется нескольким сотням. Рассмотрим самые распространенные.

1. Dropbox - Одно из самых «старых» хранилищ, которое успело завоевать популярность до того момента, когда начинали массово внедряться подобные сервисы. Пользовательский интерфейс отлично реализован. Сервис содержит удобные инструменты для генерирования ссылок, позволяет быстро обмениваться скриншотами и видео между устройствами, включает довольно функциональный web-интерфейс.

2. Google Диск - Самый популярный сервис в подборке, главное преимущество которого - многофункциональный web-интерфейс, в котором можно не только гибко управлять правами доступа к объектам и искать информацию, но и просматривать мультимедийный контент, создавать и редактировать документы в режиме совместного доступа, а также расширять функциональность посредством надстроек. Google Диск часто используют для хранения фотографий, лимит на файлы рисунков попросту отсутствует (правда, они сжимаются перед загрузкой в «облако»).

3. Яндекс.Диск - Отечественный сервис, отличающийся высокой скоростью доступа к информации. Его часто используют для того, чтобы поделиться большими файлами, поскольку после загрузки файла в «облако» можно создать уникальную ссылку на объект. С помощью web-консоли можно управ-

лять правами доступа к объектам, а также искать нужные данные среди массива информации. После установки клиента Яндекс.Диск можно задать параметры синхронизации установить настройки прокси и перемещать объекты через проводник системы. Доступен отдельный раздел для работы с фотоснимками.

4. Облако Mail.ru - Сервера этого сервиса находятся в Рунете, поэтому скорость обмена информацией с «облаком» находится на высоком уровне. Консоль web-доступа сервиса довольно функциональна, с ее помощью можно генерировать ссылки на папки, управлять параметрами доступа и просматривать видео без загрузки, создавать и редактировать документы. Все загруженные объекты проверяются посредством интегрированного антивируса [3].

Проведем анализ по следующим критериям:

1. Предоставляемое бесплатное пространство (Гб);
2. Возможности бесплатного расширения (до Гб);
3. Платное расширение (до Тб);
4. Доступ с мобильных устройств;
5. Совместный доступ ;
6. Редактирование документов;
7. Местонахождение серверов.

Сравнительные характеристики	Предоставляемое бесплатное пространство (Гб)	Возможности бесплатного расширения (до Гб)	Платное расширение (до Тб)	Доступ с мобильных устройств	Совместный доступ	Редактирование документов	Серверы на территории РФ
Dropbox	2	48	1	+	-	-	-
Google Диск	15	-	30	+	+	+	-
Яндекс.Диск	10	20	4	+	+	+	+
Облако Mail.ru	25	-	4	+	+	+	+

Таблица 1. Таблица сравнения облачных дисков

Плюсами большинства облачных пространств можно считать:

- Большой бесплатный объем дискового пространства;
- Возможность бесплатного расширения дискового пространства;
- Редактирование документов в браузере.

Как подход к решению проблемы, облачные диски не подходят по нескольким причинам:

- Каждый педагог и студент используют различные облачные диски, а для работы требуются аккаунты одного диска;
- Блокировка облачных дисков в учебных заведениях.

Так же о недостатках данного способа упоминается в работе Кудрявцева А.В.:

- Обязательная регистрация на одном из сервисов, предоставляющих такую услугу;
- Отсутствие взаимодействия между облаками различных сервисов.
- Ограничение по объему хранения данных;
- Ограничение на количество пользователей, которые могут быть подключены к ресурсу;
- Часть сервисов предлагается на платной основе [4].

Социальные сети.

Начиная с 2007 года, в университетах США активно проводятся эксперименты по использованию и внедрению сервисов социальных сетей в учебный процесс [5].

Социальная сеть — платформа, онлайн-сервис и веб-сайт, предназначенные для построения, отражения и организации социальных взаимоотношений в Интернете [6].

Без социальных сетей сейчас трудно представить нашу жизнь. Работа, учеба, покупки, развлечение - все это мы можем реализовать в социальных сетях. За относительно короткий срок, в глобальных сетях появилось очень много

таких онлайн-сервисов. Люди всех поколений имеют аккаунт к той или иной соц. сети. Самой распространенной сетью в России является Вконтакте.

В реализации дистанционного обучения Вконтакте может помочь нам своими функциями как Конференция и Документы. Пользователи могут создавать конференции с разными людьми для распространения информации и документов.

Функция Документы позволяет нам выкладывать документы большинства расширений, не превышающих 200 Мб. Они будут храниться на серверах Вконтакте, что позволяет скачать их в любой момент.

Плюсами данного способа решения проблемы будут:

- Почти каждый студент и педагог имеют аккаунты к данной сети;
- Возможность создания бесед, в которых можно выкладывать файлы заданий и лекционных материалов;
- Какую либо информацию не придется передавать каждому студенту по отдельности.

Проблемы данного способа:

- Не каждый педагог или студент захочет создавать страницу с личной информацией и передавать эти данные остальным;
- Чтобы педагог создал конференцию, в начале он должен найти каждого студента в социальной сети и затем добавить его в "друзья";
- Ссылки на документы из сети Интернет быстро теряются в сообщениях конференции;
- Блокировка большинства социальных сетей в учебных заведениях.

Локальные сети

В большинстве организаций и учреждений, в том числе и образовательных давно используются локальные сети для распространения информации, документов и файлов на все ПК в данной сети.

Локальная сеть объединяет компьютеры, установленные в одном помещении (например, компьютерный класс, состоящий из 8-12 компьютеров) или в

одном здании (несколько десятков компьютеров, установленных в различных кабинетах некоторого учреждения)

Локальная сеть - коммуникационная система, состоящая из нескольких компьютеров, соединенных между собой посредством кабелей (телефонных линий, радиоканалов), позволяющая пользователям совместно использовать ресурсы компьютера: программы, файлы, папки, а также периферийные устройства: принтеры, плоттеры, диски, модемы и т.д. [7].

Рассмотрим как такая сеть может помочь в решении задач передачи информации:

- Выходить со всех компьютеров локальной сети в Интернет;
- Обмениваться файлами между всеми компьютерами локальной сети;
- Открывать документ на нескольких компьютерах.

Недостатки:

- Привязанность к месту (в пределах одной или нескольких аудиторий);
- Частые сбои в сетях.

В рамках решения задачи, оптимальным вариантом будет создание собственной базы данных на интернет хостинге для обеспечения студентов материалами для учебного процесса. Такая база имеет ряд преимуществ по сравнению с другими способами хранения и просмотра данных:

- Возможность разработки с учетом специфики образовательного учреждения;
- Отсутствие зависимости от разработчиков других сервисов;
- Минимальная общая стоимость разрабатываемой системы;
- Простота использования;
- Доступ из любой точки;
- Использование одной комбинации логина и пароля для доступа к сервису.

1.2 Анализ программного обеспечения для реализации.

С базами данных, в большинстве случаев, работают два типа пользователей:

- Администраторы - проектировщики;
- Пользователи;

Администраторы - проектировщики разрабатывают структуру таблиц, создают и настраивают базы данных на сервере, организуют ее безопасность. Как правило, администраторы не занимаются наполнением базы конкретными данными, но в качестве исключения, на этапе отладки, проводится экспериментальное наполнение базы контентом.

Пользователи в свою очередь занимаются только наполнением и просмотром данных в базе данных. В общем случае, им не предоставляется доступа к управлению структурой базы. Доступ к категориям данным предусмотрен режимом доступа.

Следовательно, наш продукт должен работать как в пользовательском, так и в проектировочном режиме работы.

Этапами создания базы данных является:

1. Проектирование базы данных.
2. Создание базы данных.
3. Создание серверной части и обеспечение взаимосвязи с базой данных.
4. Создание сайта
5. Размещение сайта в сети интернет.

Проектирование базы данных.

Проектирование баз данных – сложная, трудоемкая, длительная работа, требующая участия высоко квалифицированных специалистов. На протяжении как создания, так и функционирования информационных систем, постоянно изменяются потребности пользователей, что в разы усложняет разработку, а также сопровождение данных систем.

Эти проблемы существенно облегчаются за счет использования инструментария для проектирования баз данных.

CASE-технология (Computer-Aided Software/System Engineering) представляет собой совокупность методологий анализа, проектирования, разработки и сопровождения сложных систем и поддерживается комплексом взаимосвязанных средств автоматизации. CASE-технология – это инструментарий для системных аналитиков, разработчиков и программистов, заменяющий бумагу и карандаш компьютером, автоматизируя процесс проектирования ПО [8].

Проанализируем несколько CASE-средств и выберем подходящую нам.

1. Rational Rose является самым известным CASE-средством. CASE-средство Rational Rose - продукт Американской фирмы Rational Software Corporation, который автоматизирует этапы проектирования и анализа программного обеспечения. В основе работы данного средства лежит синтез-методология объектно-ориентированного анализа и проектирования.
2. ERwin - CASE-средство, которое в качестве методологии использует IDEF1X. Также имеется возможность использовать методологию IE.
3. IDEF1X была разработана прежде всего для использования в армии США. Широко используется в государственных учреждениях США, в корпорациях финансовых и промышленных направленностей. Методологию IE (Information Engineering) разработали Мартин и Финкельштейн. Она используется большей частью в промышленности. Выпуск ERwin осуществляется в нескольких конфигурациях, они ориентированы на распространенные средства разработки приложений 4GL.
4. BPwin - CASE-средство, реализующее в качестве методологии IDEF0. Данный продукт разработан компанией Ltd. Logic Works. Довольно развитое средство: позволяет анализировать, документировать и улучшать бизнес процессы. В данном CASE-средстве поддерживается функциональное моделирование, что дает возможность проводить систематизированный анализ. В BPwin можно создавать как отдельные, так и смешанные модели.

5. CASE.Аналитик - отечественное CASE-средство. На данный момент наиболее конкурентоспособно на российском рынке. База данных является открытой, реализована в СУБД Paradox. Имеется возможность создания отчетов разной направленности по проекту.[9]
6. DBDesigner - это бесплатное ПО для проектирования базы данных в графической форме, которая объединяет все уровни проектирования, изменение, создание и обслуживание структуры базы данных. Он сочетает в себе дружелюбный интерфейс с мощными инструментами, которые позволяют быстро создавать сценарии SQL или XML. Разработаны инструменты для работы с существующими базами данных, чтобы извлечь структуру и дать графическую интерпретацию. Есть также встроенные модули для создания диаграммы структуры баз данных, а также мини-приложение на PHP для просмотра/изменения на основе представления [10].

Для выбора более подходящего нам CASE-средства, проведем сравнение по нескольким важным критериям:

- Бесплатный продукт;
- Возможность генерации кода SQL;
- Графический интерфейс;
- Простота использования;
- Поддержка русского языка.

Сравнительные характеристики	Бесплатное распространение	Генерация SQL кода	Графический интерфейс	Простота использования	Поддержка русского языка
Rational Rose	-	+	+	+	-
ERwin	-	+	+	-	-
BPwin	-	-	+	-	+
CASE.Аналитик	-	-	+	+	+
DBDesigner	+	+	+	+	-

Таблица 2.Таблица сравнения CASE-средств.

Анализируя данную таблицу сравнения, для проектирования нашей базы данных нам более подходит продукт DBDesigner.

DBDesigner одна из самых простых и эффективных сред графического дизайна для баз данных на рынке. Использовать этот продукт можно без ограничений, так как он выпущен под лицензий GNU.

DBdesigner включает в себя простой графический интерфейс с немалым набором инструментов и опций, которые позволяют с легкостью проектировать, моделировать и создавать нашу базу данных.

Это ПО оптимизировано специально для MySQL, что существенно облегчает дальнейшее создание базы данных.

Создание базы данных.

База данных (БД) – совокупность взаимосвязанных, хранящихся вместе данных при наличии такой минимальной избыточности, которая допускает их использование оптимальным образом для одного или нескольких приложений.

Создание базы данных, ее поддержка и обеспечение доступа пользователей к ней осуществляется централизованно с помощью специального программного инструментария – системы управления базами данных.[11]

Система управления базами данных (СУБД) - это наиболее распространенное и эффективное универсальное программное средство, предназначенное для организации и ведения логически взаимосвязанных данных на машинном носителе, а также обеспечивающее доступ к данным.

СУБД делится по способу доступа к базе данных:

- Файл-серверные;
- Клиент-серверные.

Файл-серверные СУБД. В них файлы с данными расположены централизованно на специальном файл-сервере. СУБД же должны быть расположены на каждом клиенте (рабочей станции). Доступ СУБД к данным производится посредством локальной сети. Поддержка синхронизации чтений и обновлений осуществляется за счет временных блокировок затребованных файлов.

Плюсом этой архитектуры можно назвать низкую нагрузку на файловый сервер.

К минусам же: высокая загрузка трафиком локальной сети; сложность или невозможность централизованного управления; нельзя обеспечить такие важные характеристики как надёжность, доступность и безопасность.

Файл-серверные СУБД используют в локальных приложениях; в системах с малой интенсивностью обработки данных и небольшими пиковыми нагрузками на базу данных.

Сейчас её при создании крупной информационной системы не используют.

Примеры файл-серверных СУБД:

- dBase;
- FoxPro;
- Microsoft Access;
- Paradox;
- Visual FoxPro.

Клиент-серверная СУБД расположена на сервере вместе с базой данных и осуществляет доступ к БД исключительно в монопольном режиме. Все запросы на обработку данных клиентских приложений и станций обрабатываются централизованно.

Недостатком такого типа СУБД можно назвать повышенные требования к серверу. Достоинствами: более низкую загрузку локальной сети; преимущества централизованного управления; поддержку высокой надёжности, доступности и безопасности.

Примеры клиент-серверных СУБД:

- MongoDB;
- Firebird;
- IBM DB2;
- SQLite;
- CouchDB;

- MS SQL Server;
- MySQL;
- Oracle;
- PostgreSQL [12].

Так как вся вычислительная нагрузка должна находиться на сервере базы данных (sql-сервере) для максимальной разгрузки клиентской части от вычислительных работ, нам потребуется **клиент-серверная СУБД**.

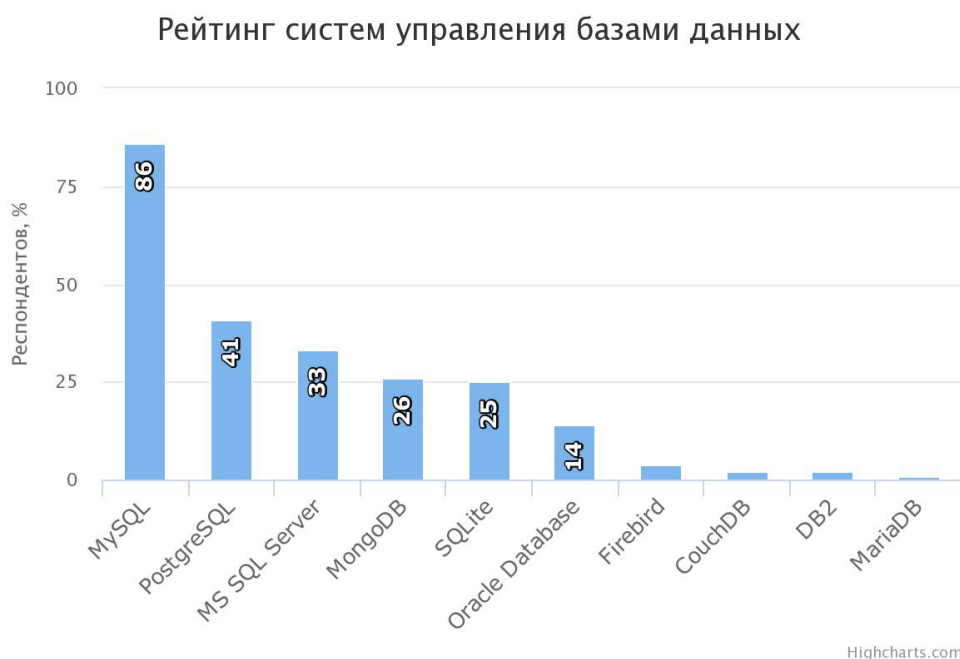


Рис. 1. Рейтинг систем управления базами данных[13].

СУБД MySQL является в наибольшей степени адаптированный для применения в глобальной сети и отличается обильной функциональностью:

- многопоточность;
- записи как фиксированной, так и переменной длины;
- гибкость системы привилегий;
- интерфейс с языком PHP;
- поддержка со стороны интернет-хостингов.

Главные преимущества MySQL:

- производительность (такая корпорация-гигант как Google использует сервера MySQL);

- масштабируемость (в компании Omniture в реальном масштабе времени используется 7000 серверов MySQL);
- надежность (по оценкам специалистов, в коде проприетарных продуктов содержится в десять с лишним раз больше уязвимостей);
- простота использования, простота внедрения (за 15 минут можно скачать и запустить систему);
- открытая и модульная разработка;
- низкие совокупные затраты (платить нужно только при потребности в поддержке).[]

СУБД MySQL является наиболее подходящей для разработки базы данных.

Создание серверной части и обеспечение взаимосвязи с базой данных.

Для программирования на серверной стороне в большинстве систем используют такие языки программирования:

- Java;
- PHP;
- Perl;
- ASP.NET;
- Ruby.

Ниже приведена статистика использования различных языков для программирования серверной части [14]:

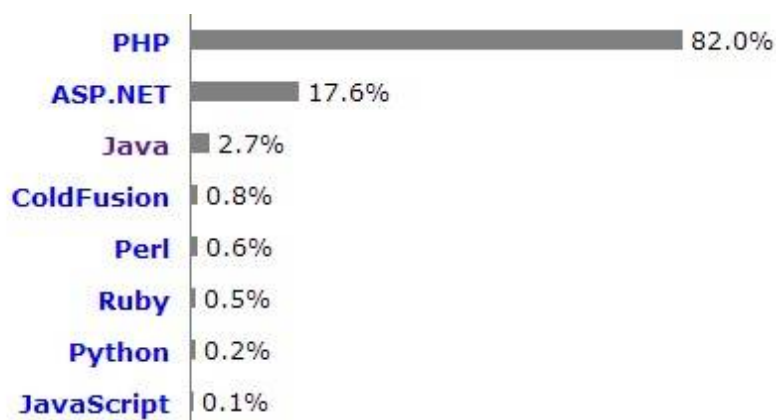


Рис. 2. Статистика использования языков программирования

Для программирования системы, нами выбран язык PHP поскольку он обладает следующими возможностями:

- поддержка большинством интернет-хостингов;
- наличие лицензии на свободное программное обеспечение;
- прост для освоения;
- возможность быстро создать динамически генерируемую web-страницу;
- возможность создания HTML-документа с внедренными командами PHP;
- PHP скрипты выполняются на стороне сервера.

Создание графической части сайта.

Сайт-совокупность логически связанных между собой веб-страниц; также место расположения контента сервера. Обычно сайт в Интернете представляет собой массив связанных данных, имеющий уникальный адрес и воспринимаемый пользователем как единое целое. Веб-сайты называются так, потому что доступ к ним происходит по протоколу HTTP[.]

Сайты в свою очередь делятся на:

- Статические сайты;
- Динамические сайты.

Для того чтобы на нашем сайте, для каждого пользователя отображалась индивидуальная информация, нам нужно создать **динамический сайт**.

Рассмотрим несколько способов создания динамического сайта:

- Самостоятельная разработка сайта. Подразумевается самостоятельное изучение научной литературы для изучения:
 - HTML;
 - CSS;
 - JavaScript;
 - PHP.

На такой способ тратиться большего всего времени, но можно полностью создать качественный сайт без вложения финансовых средств, отсутствие рекламных баннеров.

- Создание сайта с помощью бесплатных онлайн конструкторов. С помощью таких конструкторов значительно упрощается задача создания сайта. Для создания собственного сайта, потребуется только зарегистрироваться, выбрать нужный шаблон, создать доменное имя и наполнить свой сайт информацией. Но такими способами создаются простые сайты, типа тематических или сайтов-визиток. Так же наличие множества рекламы является минусом такого способа.

- Создание сайта с помощью CMS. "Систем Управления Контекстом" достаточно много на рынке, как бесплатных так и платных. Бесплатные CMS почти не отличаются от онлайн конструкторов. Платные в свою очередь дают ряд преимуществ, таких как отсутствие рекламы, полная поддержка, доменное имя второго уровня.

Для создания графической части сайта будет использоваться первый способ по ряду причин:

- отсутствие финансовых затрат;
- возможность различных способов размещения;
- повышение уровня профессионализма;
- полное знание строения сайта для быстрого изменения или модернизации.

Размещение сайта в сети интернет.

Сайты можно разместить на платных серверах, собственных серверах либо на бесплатный интернет-хостингах.

Веб-сервер - сервер, принимающий HTTP-запросы от клиентов, обычно веб-браузеров, и выдающий им HTTP-ответы, как правило, вместе с HTML-страницей, изображением, файлом, медиа-поток или другими данными.

Веб-сервером называют как программное обеспечение, выполняющее функции веб-сервера, так и непосредственно компьютер, на котором это программное обеспечение работает.

Клиент, которым обычно является веб-браузер, передаёт веб-серверу запросы на получение ресурсов, обозначенных URL-адресами.

Ресурсы - это HTML-страницы, изображения, файлы, медиа-поток или другие данные, которые необходимы клиенту. В ответ веб-сервер передаёт клиенту запрошенные данные. Этот обмен происходит по протоколу HTTP [16].

Проведем сравнение бесплатных интернет-хостингов и выберем подходящий нам.

Рассматривать будем хостинги с поддержкой PHP и MySQL.

Хостинг	Тип	Трафик в месяц, (Gb)	Лимит диска, (Mb)	Лимит ОЗУ, (Mb)	Рек- лама	Домены	Load, (kb/s)	Uptime, %
Holm.ru	RU	9999	100	64	+	h1-h19.ru	27,26	99,81
Hut.ru	RU	9999	100	64	+	hut.ru h19.ru	19,09	99,81
Hostinger.ru	RU	100	2000	128	-	zz.mu pusku.com	112,9	99,25
freehost96.ru	RU	200	2000	128	+	freehost96.ru	28,51	99
000webhost.com	EN	100	1500	64	-	000webhosta pp.com	23,38	99,91

Таблица 3. Сравнение интернет-хостингов.

Несмотря на то, что Hostinger.ru выигрывает по большинству критериев, нами выбран хостинг 000webhost.com, так как процент его бесперебойной работы выше, в следствии чего, пользователи смогут использовать данный хостинг для учебных целях.

Исходя из всех вышеуказанных пунктов, оптимальной комбинацией для разработки базы данных на интернет хостинге будет:

- CASE-средство DBDesigner;
- интерпретатор языка PHP;
- СУБД MySQL;
- интернет-хостинг 000webhost.com.

Важно, что все выбранное нами ПО является с открытым исходным кодами и распространяется с бесплатной лицензией.

1.3 Техническое задание.

ТЕХНИЧЕСКОЕ ЗАДАНИЕ на разработку информационной системы Сайт "База учебных материалов"

Составлен на основе ГОСТ 34.602-89 «Техническое задание на создание автоматизированной системы»

1. Общие сведения.

1.1. Название организации-заказчика:

Проектирования и организация базы данных на интернет хостинге для обеспечения студентов материалами и сведениями осуществляется в рамках научно-исследовательской деятельности кафедры информационно-коммуникационных технологий в образовании Уральского Государственного Педагогического Университета.

1.2. Название продукта разработки (проектирования):

Сайт "База учебных материалов" ("Бум").

1.3. Назначение продукта:

Обеспечение студентов лекционными материалами и лабораторными работами (заданиями).

1.4. Плановые сроки начала и окончания работ.

13.01.17-14.02.17

2. Характеристика области применения продукта.

2.1. Процессы и структуры, в которых предполагается использование продукта разработки:

Размещение материалов и сведений для учебного процесса. Использование продукта в образовательных учреждениях.

2.2. Характеристика персонала (количество, квалификация, степень готовности):

Один человек-администратор сайта и базы данных, со знанием языков программирования PHP, HTML, MySQL, CSS.

3. Требования к продукту разработки.

3.1. Требования к продукту в целом:

Для учащихся:

- Регистрация новых пользователей;
- Просмотр имеющихся материалов и заданий по изучаемым предметам;
- Информирование учащихся о новых лекционных материалах, лабораторных работ.

Для преподавателей:

- Просмотр имеющихся материалов и заданий по читаемым предметам;
- Размещение в базе данных и на сайте лекционных материалов, лабораторных работ.

3.2. Аппаратные требования.

Процессор	Pentium 3 1800Гц и выше
RAM	256Мб и больше
HDD	HDD 20Гб и больше
Видеокарта	128Мб и более
Сетевая карта	10-100Mb/s
Разрешение монитора	Не менее 1024x798
Клавиатура	Стандартная PS/2 или USB
Мышь	Стандартная PS/2 или USB

Таблица 4. Аппаратные требования.

Требования к аппаратной конфигурации могут изменяться, в зависимости от размера базы данных, а так же от вариантов реализации продукта.

3.3. Указание системного программного обеспечения:

- Операционная система: Windows 7- Windows 10;
 - Браузер: IE (не ниже 8), Google chrome, Opera, Mazila Firefox, Яндекс Браузер.
- 3.4. Указание программного обеспечения, используемого для реализации.
- PHP 4.6.6;
 - DBDesihner 4;
 - Notepad++;
 - HTML 5;
 - CSS3.
- 3.5. Для сетевых систем – особенности реализации серверной и клиентской частей.
- Серверная часть реализована с помощью интерпретатора языка PHP, СУБД - MySQL. Размещение на бесплатном сервере хостинга.
- 3.6. Форматы входных и выходных данных:
- Страницы сайта сохранены в формате *.php и *.html;
 - Формат базы данных *.sql;
 - Выходные данные - гиперссылка.
- 3.7. Источники данных и порядок их ввода в систему (программу), порядок вывода, хранения.
- Источниками данных является ссылки на документы или файлы:
 - Ввод каждого нового задания или лекции, прикрепление ссылки осуществляется на соответствующей страничке сайта;
 - Выбор данных для вывода в новом браузерном окне осуществляется на соответствующей страничке сайта.
 - Ссылки на документы и файлы хранятся в таблицах базы данных.
- 3.8. Порядок взаимодействия с другими системами, возможности обмена информацией:

- Продукт с помощью ссылок перемещает пользователя на различные сервисы, позволяющие хранить документы и файлы (облачные диски, файлообменники и тд.);
- Возможность импортирования базы данных на другие сервера.

3.9. Меры защиты информации.

- Аутентификация происходит по средствам ввода комбинации логина и пароля;
- Для работы с базой данных создан пользователь с уникальной комбинацией логина и пароля;
- Пароли передаются и хранятся в базе данных в измененном виде с помощью хеширования;
- Запрет на создание аккаунтов с одинаковыми логинами;
- Пользователь (кроме педагога), не сможет пройти регистрацию, не имея оригинального номера (в случае с ВУЗами - номер студенческого билета).

4. Требования к пользовательскому интерфейсу.

4.1. Общая характеристика пользовательского интерфейса.

Сайт состоит из:

- Верхней навигационной панели,
- Центрального окна вывода информации;
- левой панели навигации и состояния.

4.2. Размещение информации на экране, дизайн экрана.

- На левой панели меню ввода логина и пароля, регистрации. После входа на сайт, указывается информация о пользователе;
- На верхней панели находятся кнопки Главная, Список предметов, Информация;
- В центральном окне отображается информация, либо меню, в зависимости от выбранной категории.

4.3. Особенности ввода информации пользователем, представление выходных данных.

- Ввод информации педагогом происходит по средствам ручного ввода URL-ссылки и названия для этой ссылки;
- Представление данных происходит в новом браузерном окне, после нажатия на интересующее название.

5. Требования к документированию.

5.1. Перечень сопроводительной документации.

- ВКР с приложениями;
- Реферат;

5.2. Требования к содержанию отдельных документов.

6. Порядок сдачи-приемки продукта.

Начало работ:13.11.17;

Окончание работ:30.11.17.

В процессе исследования и написания данной главы, были выполнены следующие задачи:

1. Обоснование актуальности создания собственного сайта и базы данных для образовательных целей;
2. Выбрано программное обеспечение для организации базы данных;
3. Разработано техническое задание.

Глава 2. Практическая (оригинальная) часть

2.1 Разработка сайта и базы данных.

Проектирование базы данных.

Проектирование базы данных было выполнено в CASE-средстве DBDesigner 4.

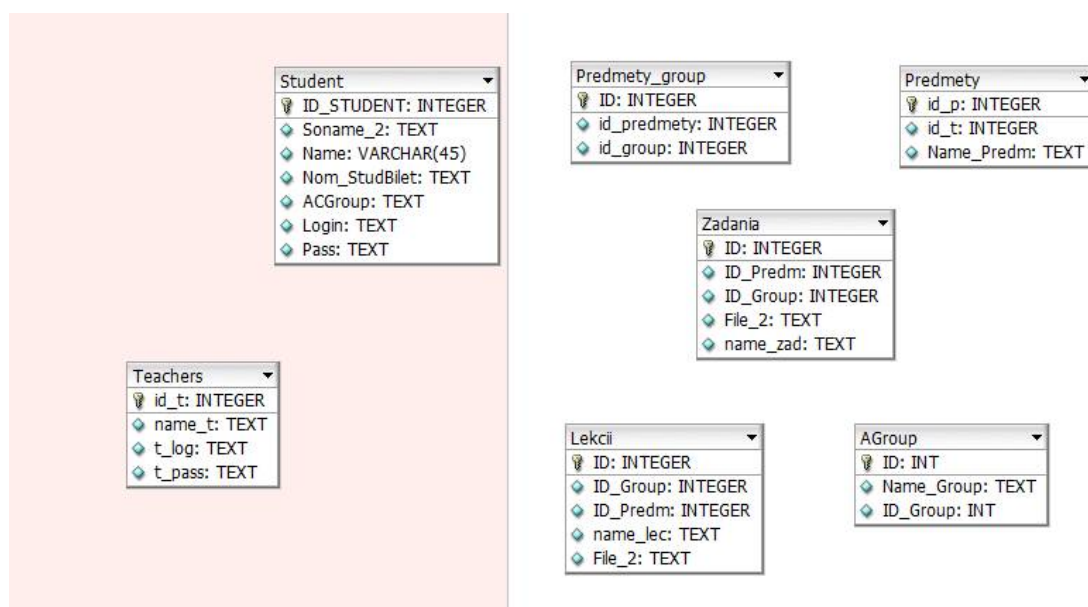


Рисунок 3. Проектирование таблиц для базы данных.

В DBDesigner 4 создаются таблицы с полями, задаются названия и параметры полей. Далее с помощью функции **SQL Create Script ...** генерируется код SQL, который вставляется в СУБД MySQL для создания базы данных с таблицами и полями. Сгенерированный SQL код предоставлен в Приложении 1.

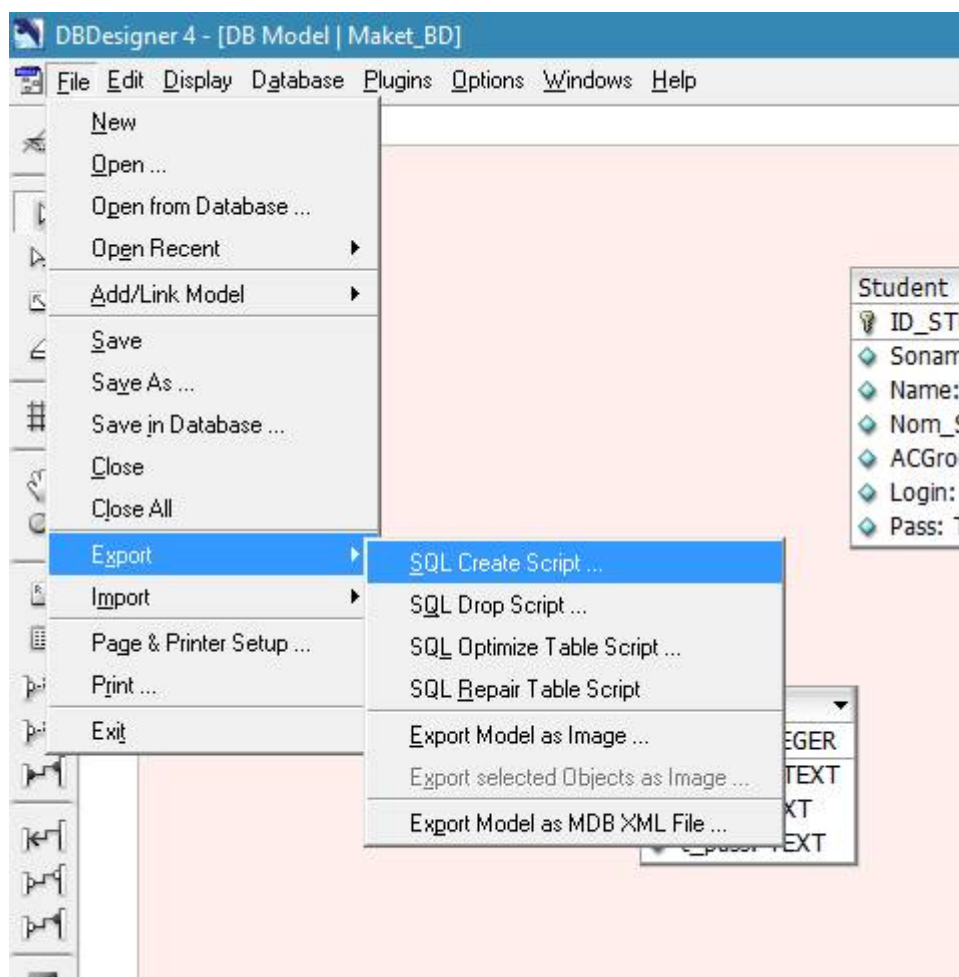


Рисунок 4. Генерация SQL кода.

Создание базы данных.

До начала этого этапа, была проведена регистрация на интернет хостинге 000webhost.com. После регистрации и создания сайта с созданным доменным именем третьего порядка, создается база данных при нажатии на кнопку **New Database**.

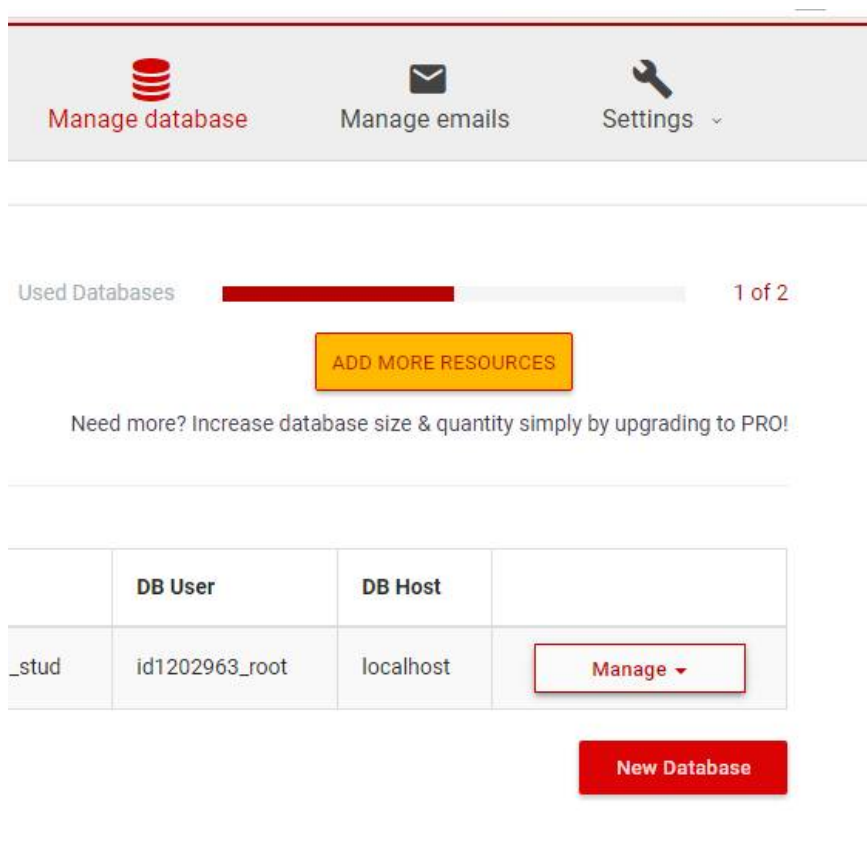


Рисунок 5. Создание БД на хостинге.

После входа в базу данных с root правами, открывается вкладка SQL, вставляется код SQL и нажимается кнопка GO, которая выполняет SQL код.

Table	Action	Rows
☐ AGroup	★ Browse Structure Search Insert Empty Drop	1
☐ Lekcii	★ Browse Structure Search Insert Empty Drop	2
☐ Predmety	★ Browse Structure Search Insert Empty Drop	1
☐ Predmety_group	★ Browse Structure Search Insert Empty Drop	1
☐ Student	★ Browse Structure Search Insert Empty Drop	1
☐ Teachers	★ Browse Structure Search Insert Empty Drop	1
☐ Zadania	★ Browse Structure Search Insert Empty Drop	1
7 tables	Sum	6

Рисунок 6. Созданные таблицы.

Заполнение таблиц проводится администратором базы данных в ручном режиме.

Клиентская часть.

Внешний вид сайта был создан с помощью HTML-разметки и языка описания внешнего вида CSS.

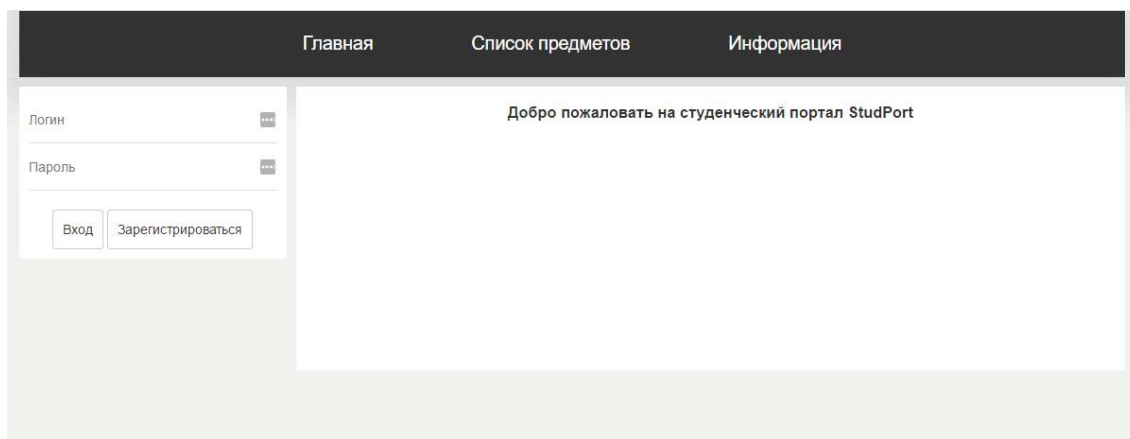


Рисунок 7. Внешний вид сайта.

Подключение к базе данных.

Для взаимодействия базы данных с сайтом, создан скрипт создания подключения базы данных к сайту. Код файла указан в Приложение 2.

Группы пользователей.

Пользователи web-сайта разделены на 2 группы:

- Гости;
- Действующие пользователи.

К группе «Гости» относятся пользователи, не имеющие доступа к внутренним ресурсам сайта. Для этой группы предусмотрена страничка регистрации и страничка информации. Зарегистрироваться смогут только те пользователи, которые обучаются в учебном заведении, использующее данный сайт в учебных целях.

Группа «Действующие пользователи» в свою очередь делятся еще на 2 группы:

- Учащиеся;
- Сотрудник:

- Преподаватель;
- Методист;
- Административная должность.

Этой группе предусмотрено несколько функций:

- Авторизация на сайте;
- Просмотр изучаемых/преподаваемых предметов;
- Просмотр лекционных и лабораторных материалов (для учащихся);
- Просмотр уже имеющихся, и добавление новых лекционных и лабораторных материалов (для сотрудников).

Регистрация новых пользователей.

Регистрация на сайте – это действия, направленные на создание личной учетной записи на web-ресурсе, с целью получения доступа к его полному функционалу (будь то расширенные возможности или доступ к страницам Интернет-ресурсов, просматривать которые возможно лишь после регистрации). [17]

Для учащихся имеется возможность регистрации. Регистрация новых пользователей-учащихся проходит в несколько этапов.

1. На главной странице расположена кнопка «Зарегистрироваться», при нажатии на которую вызывается форма регистрации.

Номер студенческого билета:

Логин:

Пароль:

Рисунок 8. Форма регистрации.

Необходимо заполнить следующие поля:

- Номер студенческого билета. По этому номеру определяется, учиться ли будущий пользователь в учебном заведении. В базе данных в таблице Student указаны ФИО, название групп и номер студенческого билета.

+ Options			ID_STUDENT	Soname_2	Name	ACGroup	Nom_StudBilet	Login	Pass
☐	Edit Copy Delete		1	Шитиков	Виталий	ИС-41	54878	shit	c4ca4238a0b923820dcc509a6f75849b
☐	Edit Copy Delete		2	Стрелкова	Александра	ИС-41	1	sasha	e99ae0eadc4b1eafec9aae55fe67f842
☐	Edit Copy Delete		3	Баженова	Лиза	БИ-41	2	baz	c81e728d9d4c2f636f067f89cc14862c
☐	Edit Copy Delete		4	Камалитдинова	Элида	ИС-41	3	kamal	202cb962ac59075b964b07152d234b70
☐	Edit Copy Delete		5	Догодин	Кирилл	ИС-41	213456	kinill	9003d1df22eb4d3820015070385194c8
☐	Edit Copy Delete		6	Кобелев	Алексей	Б-41	4548	kob	7a54486317b6dde9ce8fadaf3e7601f4
☐	Edit Copy Delete		7	Тюкин	Сергей	БИ-41	4777	tuk	c20ad4d76fe97759aa27a0c99bfff6710
☐	Edit Copy Delete		8	Ряпосова	Валерия	БИ-41	7777	123	202cb962ac59075b964b07152d234b70
☐	Edit Copy Delete		9	Глушкова	Люда	БИ-41	4477	Lu	3e7e122bf08f48432c961ba491089dc9
☐	Edit Copy Delete		10	Пяткова	Анна	БИ-41	4447	Anna	ef67e3e4b1af8a2ccf2b71f0c31cbe22
☐	Edit Copy Delete		11	Рямов	Илья	Б-41	1777	NULL	NULL
☐	Edit Copy Delete		12	Давыдова	Екатерина	Б-41	2777	dov	c4ca4238a0b923820dcc509a6f75849b

Рисунок 9. Таблица Student.

- **Логин.** При вводе логина проверяется, существует ли пользователь с таким же логином пароля, и при нахождении соответствия, предлагается использовать другой логин.
- **Пароль.**

Авторизация.

Авторизация на сайте проходит с помощью сессий.

Сессии в РНР представляют из себя механизм сохранения на стороне сервера информации о компьютере клиента. Слово session с английского переводится как сеанс.

Сессии в РНР очень похожи на механизм cookie, те же самые пары ключ => значение, только они хранятся на стороне сервера.[18]

Авторизация на сайте происходит следующим образом:



Рисунок 10. Процесс авторизации.

Код документа Login.php указан в Приложении 3.

Авторизация учеников происходит только в том случае, если они зарегистрированы. Сотрудники образовательного учреждения не проходят этап регистрации, их логин и пароль задается в систему через базу данных.

Безопасность.

Так как в базе данных не хранятся данные о пользователях, которые могли бы пригодиться злоумышленникам, было реализовано несколько простых методов защиты:

- Авторизация на сайте с помощью логина и пароля;

- Пароли по сети передаются зашифрованными, используется алгоритм md5;
- Отказ в регистрации пользователей, применяющих логин, уже используемый на сайте;
- Аутентификация пользователей;

MD5 (англ. Message Digest 5) — 128-битный алгоритм хеширования, разработанный профессором Рональдом Л. Ривестом из Массачусетского технологического института (Massachusetts Institute of Technology, MIT) в 1991 году. Предназначен для создания «отпечатков» или дайджестов сообщения произвольной длины и последующей проверки их подлинности. Широко применялся для проверки целостности информации и хранения паролей в закрытом виде.[19]

Список предметов.

Для пользователей-учеников во вкладке **Список предметов**, предоставлен список изучаемых предметов. Все предметы и коды групп, которым преподается этот предмет, хранятся в таблице **Predmety**. Так же, в этой таблице хранится код преподавателя, читающего этот предмет. Вывод списка предметов реализован в файле sp_predm.php следующим путем:

```
$query_AG = "SELECT ACGroup FROM Student WHERE Login = '$login' ";
$result_AG = mysql_query($query_AG) or die ( "Error : ".mysql_error() );
while (($query_AG1 = mysql_fetch_assoc($result_AG)))
{ $AG= $query_AG1['ACGroup'];//считали название академической группы }

$query_IG = "SELECT ID_Group FROM AGroup WHERE Name_Group = '$AG'";

$result_IG = mysql_query($query_IG) or die ( "Error : ".mysql_error() );
while (($query_IG1 = mysql_fetch_assoc($result_IG)))
{ $IG= $query_IG1['ID_Group'];//считали код группы }

$query_NP = "SELECT * from Predmety INNER JOIN Predmety_group
```

```

ON Predmety.id_p = Predmety_group.id_predmety
WHERE Predmety_group.id_group=" . $IG;
$result_NP = mysql_query($query_NP) or die ( "Error : ".mysql_error() );
while (($query_NP1 = mysql_fetch_assoc($result_NP)))
{ $NP= $query_NP1['Name_Predm'];
  $ID_Predm= $query_NP1['id_p'];
  echo ' <a href="Vybor.php?id_p=' . $ID_Predm. '"> . $NP . ' </a><br>'
; } //Вывод списка предметов с возможностью нажатия на название для даль-
нейшего перехода

```

При нажатии на один из элементов массива, выполняется переход на вкладку, где предлагаются дальнейшие действия:

- Просмотр лекционных заданий;
- Просмотр лабораторных работ.

При нажатии на тот, либо другой элемент, происходит отображение списка занесенных в базу данных названий материалов и хранящих в себе гиперссылку на ресурс, где хранится файл или документ.

Для пользователей-преподавателей на этих страничках предусмотрены кнопки добавления лекционных и лабораторных материалов. Пример реализации указан ниже:

```

<?php
if (isset($_SESSION['teacher'])) //Если авторизован учитель, то
{require_once("dbconnect.php");
echo "<a href='dob_l.php?id_p=" . $_GET['id_p']."'> Добавить лекцию
</a><br>" ; } //вывести кнопку для перехода на вкладку добавления лекции
?>

```

Общая модель сайта.

Структура сайта указана на картинке ниже:

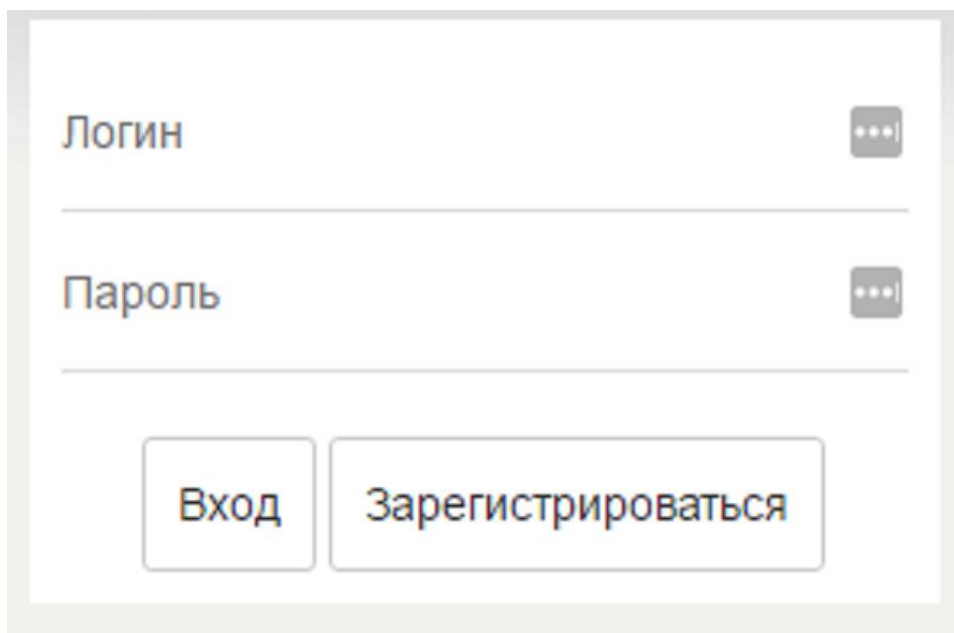


Рисунок 11. Структура сайта.

По этой структуре видно, что сайт имеет три основные вкладки, но к просмотру информации на вкладке Список предметом имеют доступ только авторизованные пользователи.

2.2 Описание продукта (результата разработки)

Работа с сайтом начинается с авторизации.

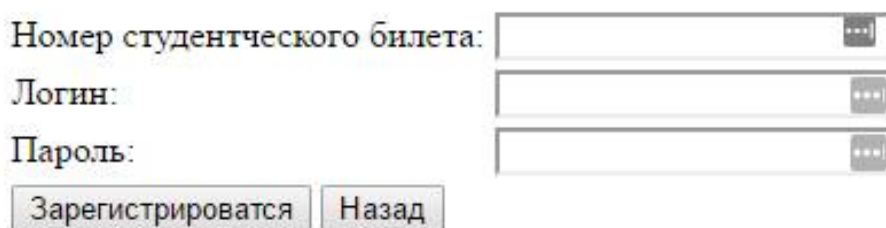


The image shows a login form with two input fields. The first field is labeled 'Логин' (Login) and the second is labeled 'Пароль' (Password). Both fields have a small icon with three dots to their right, likely for toggling password visibility. Below the fields are two buttons: 'Вход' (Login) and 'Зарегистрироваться' (Register).

Рисунок 12. Авторизация на сайте.

Если пользователь уже зарегистрирован в системе, то он должен ввести свой логин и пароль и нажать кнопку вход.

Для регистрации пользователь нажимает кнопку Зарегистрироваться и попадает на вкладку регистрации:



The image shows a registration form. It has three input fields: 'Номер студенческого билета:' (Student ID), 'Логин:' (Login), and 'Пароль:' (Password). Each field has a small icon with three dots to its right. Below the fields are two buttons: 'Зарегистрироваться' (Register) and 'Назад' (Back).

Рисунок 13. Регистрация.

Студент вводит свой номер студенческого билета (пункт может меняться в зависимости от учебного учреждения), логин и пароль.

При успешной регистрации появляется следующее окно:

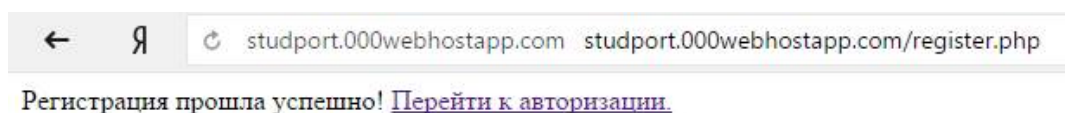


Рисунок 14. Окно при успешной регистрации.

Пользователь переходит к регистрации, вводит свои логин и пароль и попадает на главную страничку сайта.

Если же регистрация не прошла, а это могло произойти по таким причинам, как:

- Логин уже используется;
- Номера студенческого билета не существует в базе данных,

то пользователю отображается ошибка регистрации:

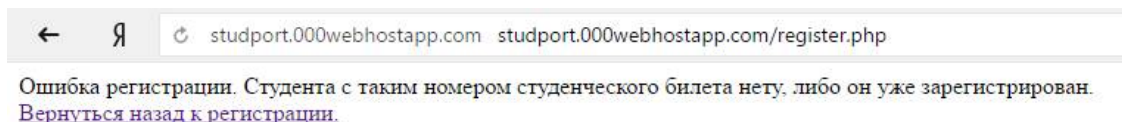


Рисунок 15. Ошибка регистрации.

После успешной авторизации, пользователь попадает на главную страничку сайта.

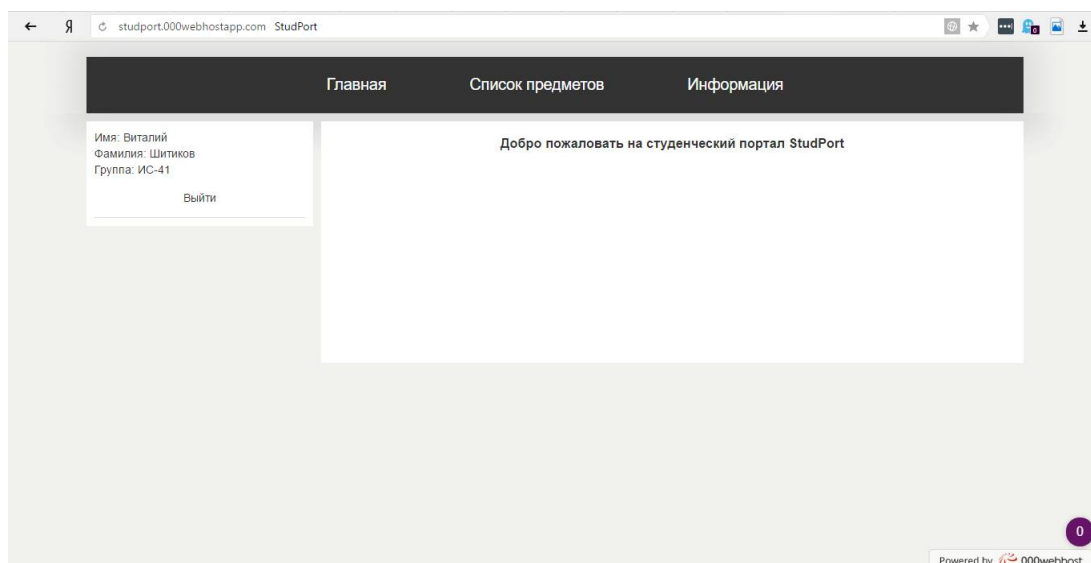


Рисунок 16. Главная страничка.

В левой части сайта указывается информация о пользователе, так же кнопка завершения сессии.

До данного этапа у пользователей разных групп (ученики и сотрудники) работа с сайтом одинаковая, за исключением процесса регистрации (у пользователей группы сотрудники регистрация не предусмотрена, все данные уже занесены в базу и выдаются каждому пользователю этой группы логин и пароль).

Рассмотрим дальнейшую работу с сайтом для пользователей группы "ученик".

Для просмотра списка предметов, нужно перейти во вкладку Список предметов:

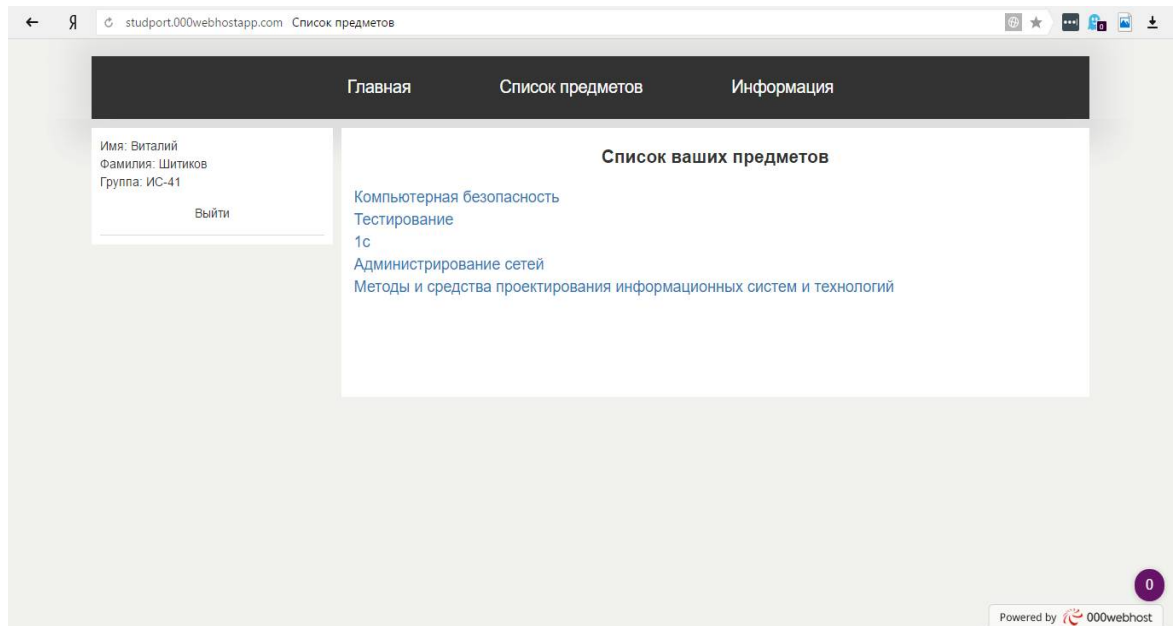


Рисунок 17. Список изучаемых предметов.

При нажатии на один из предметов, пользователь попадает во вкладку с выбором просмотра материалов лекционных либо лабораторных, а так же предоставляется информация о преподавателе, читающем данный курс.

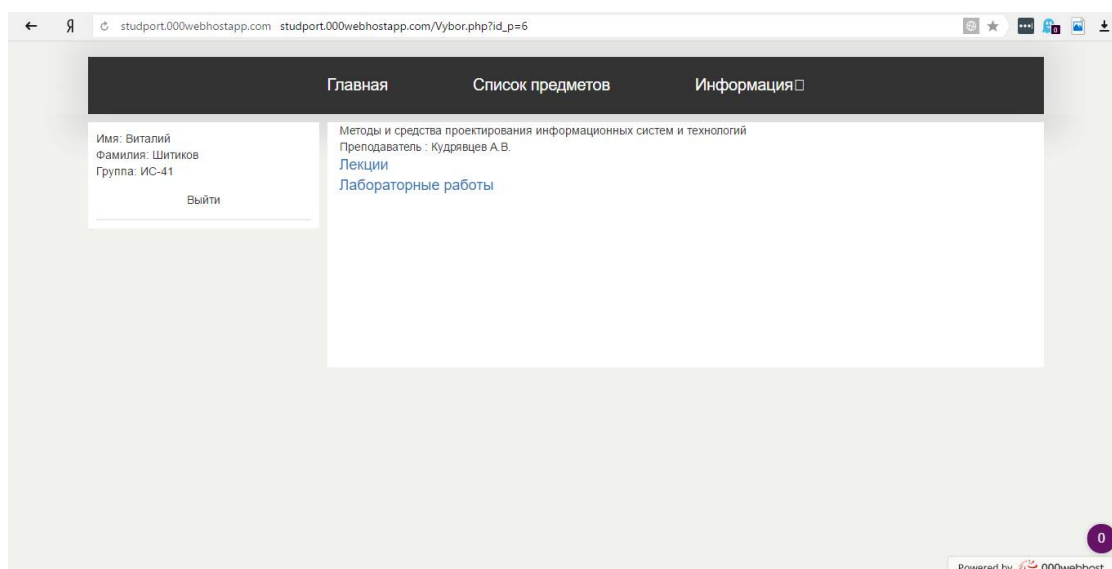


Рисунок 18. Выбор материалов предмета.

Для просмотра лекций данного курса, следует выбрать пункт Лекции:

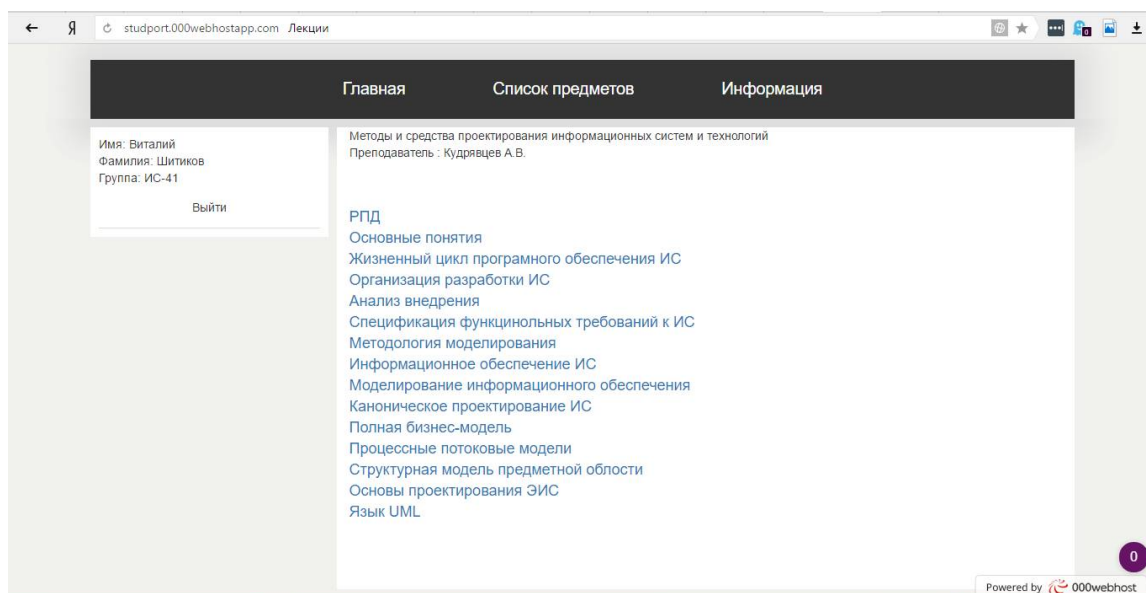


Рисунок 19. Лекции.

При нажатии на одну из лекций, происходит перенаправление на сервис, где хранится данный документ.

Теперь рассмотрим работу с сайтом для группы "сотрудник".

При переходе на вкладку Список предметов, преподаватель видит список предметов, которые он преподает. Для добавления новой лекции, преподаватель должен выбрать вкладку Добавить лекцию:

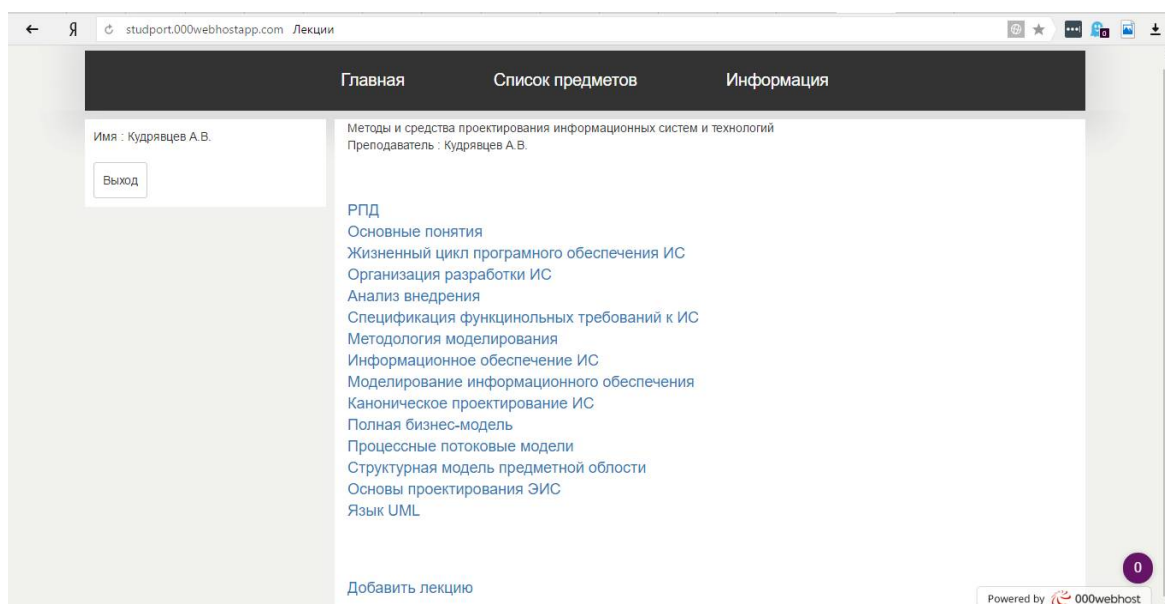
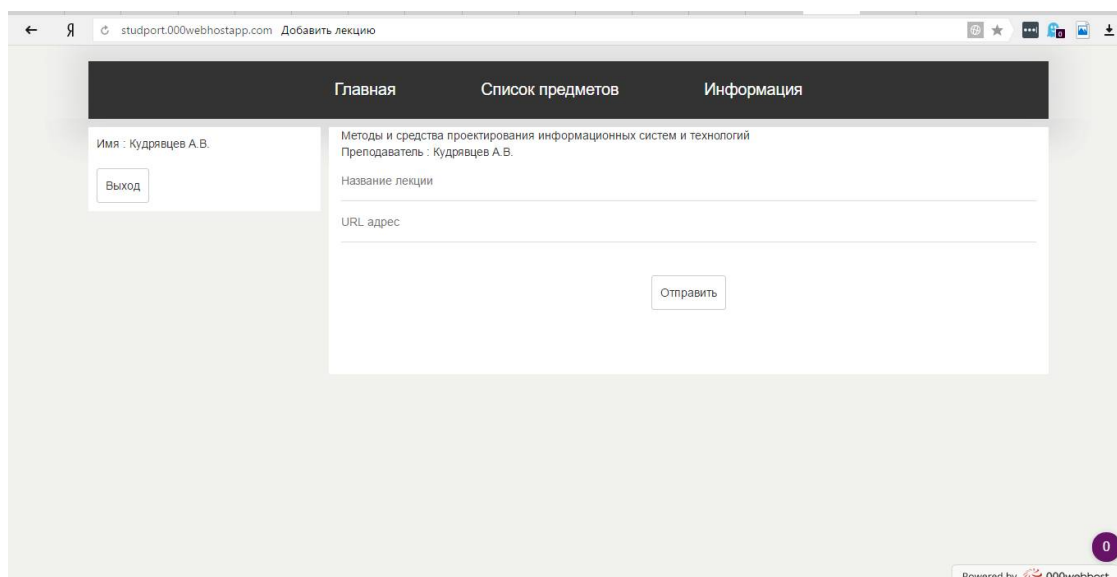


Рисунок 20. Список добавленных лекций.

Для добавления лекции по выбранному предмету, требуется ввести название лекции и добавить URL-адрес, где хранится документ или файл.



Добавление лабораторных работ проходит в аналогичном порядке.

Заключение

1. Данная работа посвящена разработке базы данных и сайта "Базы учебных материалов". У данного сайта имеется ряд достоинств, которые оценят пользователи различных групп: простой интерфейс, приятный дизайн, а главное-доступность. С данным сайтом, преподавателям не требуется решать, каким способом организовать обеспечение учащихся информацией, а учащимся не требуется запоминать различные сервисы, где хранятся материалы различных преподавателей. Создание такого сайта является актуальной темой.
2. Для реализации продукта было выбрано следующее ПО и языки программирования и разметки:
DBDesigner, PHP, СУБД MySQL, Notepad++, HTML, CSS. Данные ПО были проанализированы с подобными, и выбраны как более подходящие для реализации сайта "Бум". Все выбранные ПО распространяются под бесплатной лицензией.
3. Структура и дизайн сайта, база данных были разработаны в соответствии с требованиями технического задания. Реализованы механизмы авторизации, аутентификации, внедрены некоторые методы организации безопасности. Сайт наполнен некоторым контентом.
4. Работа сайта протестирована на нескольких браузерах. В каждом, сайт выполняет свои функции без замечаний.

Список информационных источников

1. Стариченко Б.Е. О построении информационного обеспечения учебного процесса в вузе // Педагогическое образование в России. — 2012. — №5. — С. 39 - 44.
2. Емельянова О. А. Применение облачных технологий в образовании // Молодой ученый. — 2014. — №3. — С. 907-909.
3. Программы для хранения данных (фото, видео, документы) в облаке // SoftHome URL: <https://www.softhome.ru/article/programmy-dlya-hranieniya-dannyh> (дата обращения: 27.04.17).
4. Кудрявцев А.В. Использование интернет-хостингов для хранения учебной информации в целях реализации принципов открытого образования. // Педагогическое образование в России. - 2016. - №7.
5. Воронкин А. С. Социальные сети: эволюция, структура, анализ. // Образовательные технологии и общество. - 2014. - №1.
6. Социальная сеть // Википедия URL: https://ru.wikipedia.org/wiki/Социальная_сеть (дата обращения: 25.04.17).
7. Локальная сеть. Что это? Виды локальных сетей // Pro-Spo URL: <http://pro-spo.ru/network-tech/3011-2012-03-06-11-42-08> (дата обращения: 27.04.17).
8. Рыбанов А.А. Инструментальные средства автоматизированного проектирования баз данных: Учебное пособие и варианты заданий к лабораторным работам по дисциплине «Базы данных» / ВолгГТУ, Волгоград, 2007. – 96 с.
9. Билялова В.М. Сравнительный обзор case-средств для проектирования программных систем. // Материалы VII Международной студенческой электронной научной конференции «Студенческий научный форум».

- 10.DBDesigner // Википедия URL:
<https://fr.wikipedia.org/wiki/DBDesigner> (дата обращения: 27.04.17).
11. Информатика: Учебно-методический комплекс.
/Г.А. Столярова, Н.В. Клиначёв, и др. – Offline версия 1.0 – Челябинск, 2000-2013
12. СУБД - система управления базами данных // Информатика
URL: <http://inphormatika.ru/lectures/subd.html> (дата обращения: 02.04.17).
- 13.Рейтинг систем управления базами данных (СУБД) 2016 // TAGLINE URL: <http://inphormatika.ru/lectures/subd.html> (дата обращения: 05.03.17).
- 14.Usage of server-side programming languages for websites // W3Techs
URL:https://w3techs.com/technologies/overview/programming_language/all (дата обращения: 05.05.17).
- 15.Воройский Ф.С. Информатика. Энциклопедический систематизированный словарь-справочник. — М.: Физматлит, 2006. — С. 432. — 945 с.
- 16.Веб-сервер // Википедия URL: <https://ru.wikipedia.org/wiki/Веб-сервер> (дата обращения: 27.04.17).
- 17.Зачем нужна регистрация на web-ресурсах? // Компьютерная грамотность с Надеждой URL: <http://www.compgramotnost.ru/internet-gramotnost/zachem-nuzhna-registratsiya-na-web-resursah> (дата обращения: 05.05.17).
- 18.Сессии в PHP // Уроки сайтостроения URL:
<http://www.compgramotnost.ru/internet-gramotnost/zachem-nuzhna-registratsiya-na-web-resursah> (дата обращения: 05.04.17).
- 19.MD5 // Википедия URL: <https://ru.wikipedia.org/wiki/MD5> (дата обращения: 27.04.17).

Приложения

Приложение 1.

```
CREATE TABLE Student (  
    ID_STUDENT INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,  
    Soname_2 TEXT NULL,  
    Name VARCHAR(45) NULL,  
    Nom_StudBilet TEXT NULL,  
    ACGroup TEXT NULL,  
    Login TEXT NULL,  
    Pass TEXT NULL,  
    PRIMARY KEY(ID_STUDENT)  
);  
  
CREATE TABLE Teachers (  
    id_t INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,  
    name_t TEXT NULL,  
    t_log TEXT NULL,  
    t_pass TEXT NULL,  
    PRIMARY KEY(id_t)  
);  
  
CREATE TABLE Zadania (  
    ID INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,  
    ID_Predm INTEGER UNSIGNED NULL,  
    ID_Group INTEGER UNSIGNED NULL,  
    File_2 TEXT NULL,  
    name_zad TEXT NULL,  
    PRIMARY KEY(ID)  
);  
  
CREATE TABLE Predmety_group (  
    ID INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
```

```

id_predmety INTEGER UNSIGNED NULL,
id_group INTEGER UNSIGNED NULL,
PRIMARY KEY(ID)
);

CREATE TABLE AGroup (
  ID INT NOT NULL AUTO_INCREMENT,
  Name_Group TEXT NULL,
  ID_Group INT NULL,
  PRIMARY KEY(ID)
);

CREATE TABLE Lekcii (
  ID INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  ID_Group INTEGER UNSIGNED NULL,
  ID_Predm INTEGER UNSIGNED NULL,
  name_lec TEXT NULL,
  File_2 TEXT NULL,
  PRIMARY KEY(ID)
);

CREATE TABLE Predmety (
  id_p INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  id_t INTEGER UNSIGNED NULL,
  Name_Predm TEXT NULL,
  PRIMARY KEY(id_p));

```

Приложение 2.

```
<?php
```

```
    $server = "localhost:3306"; /* Имя хоста */
```

```
    $database = "id1202963_stud"; /* Имя БД */
```

```
    $username = "id1202963_root"; /* Логин пользователя БД */
```

```
    $password = "qwertyqwerty1116e8"; /* Пароль пользователя */
```

```
    // Подключение к серверу MySQL
```

```
    $connect = mysql_connect($server,$username,$password) or die (
mysql_error() );
```

```
    // Выбираем БД к которой хотим подключиться
```

```
    mysql_select_db($database, $connect);
```

```
?>
```

Приложение 3.

```
<?php
```

```
session_start();
```

```
require_once("dbconnect.php");
```

```
if(isset($_POST["l_username"])){ $l_username = $_POST["l_username"]; }
```

```
if(isset($_POST["l_password"])){ $l_password = $_POST["l_password"]; }
```

```
if(isset($_POST["l_send"])){ $l_send = $_POST["l_send"]; }
```

```
if(isset($l_send)){
```

```
    // Проверяем если была нажата кнопка Войти. Если да, то сравниваем  
    данные полученные из формы с тем логином и паролем который есть в  
    БД и если они совпадают то пользователь успешно авторизирован, иначе,  
    выводим сообщение что неправильный логин или пароль. Если кнопка не  
    была нажата, значит что пользователь зашел на страницу напрямую и по-  
    этому выводим ему сообщение об этом..
```

```
    $query = " SELECT * FROM Student WHERE Login = '$l_username'  
AND Pass = md5('$l_password')";
```

```
    $result = mysql_query($query) or die ( "Error : ".mysql_error() );
```

```
    /* Проверяем, если в базе нет пользователей с такими данными, то  
    выводим сообщение об ошибке */
```

```
if(mysql_num_rows($result) < 1){
```

```
    $query_t = " SELECT * FROM Teachers WHERE t_log =  
'$l_username' AND t_pass = ('$l_password')";
```

```
    $result_t = mysql_query($query_t) or die ( "Error : ".mysql_error() );
```

```

        if(mysql_num_rows($result_t) < 1){
echo "Неправильный логин или пароль";
        }else{
            $_SESSION['teacher']=$l_username;
            include ("menu.php");}

    }else{

        // Если введенные данные совпадают с данными из базы, то сохра-
        няем логин и пароль в массив сессий.
        $_SESSION['username'] = $l_username;

        if (isset($_SESSION['username']))
        {
            include("menu.php");
        }

    }

}

}else{

    echo "Вы зашли на эту страницу напрямую, поэтому нет данных для
    обработки. Вы можете возвращаться на <a href='index.php'> главную
    страницу </a>";

}

?>

```

