

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»**  
( Н И У « Б е л Г У » )

ИНСТИТУТ ИНЖЕНЕРНЫХ И ЦИФРОВЫХ ТЕХНОЛОГИЙ

КАФЕДРА ОБЩЕЙ МАТЕМАТИКИ

**WEB-РЕСУРС ЗАОЧНОЙ МАТЕМАТИЧЕСКОЙ ШКОЛЫ НИУ  
«БЕЛГУ» С ИСПОЛЬЗОВАНИЕМ СРЕДСТВ PHP И MYSQL**

Выпускная квалификационная работа  
обучающегося по направлению подготовки  
01.03.02 Прикладная математика и информатика  
очной формы обучения, группы 12001510  
Громова Владислава Андреевича

Научный руководитель  
д.т.н., профессор  
Аверин Г.В.

БЕЛГОРОД 2019

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	3
1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ.....	6
1.1 Описание поставленной задачи.....	6
1.2 Характеристика информационных ресурсов и его роль в обучении.....	8
1.3 Аналоги разработки.....	10
1.4 Выбор средств для разработки Web-ресурса математической школы...	15
2 ПРОЕКТИРОВАНИЕ WEB-РЕСУРСА ДЛЯ МАТЕМАТИЧЕСКОЙ ШКОЛЫ .....	18
2.1 Проектирование структуры сайта .....	18
2.2 Разработка структуры базы данных .....	21
3 РАЗРАБОТКА WEB-РЕСУРСА ЗАОЧНОЙ МАТЕМАТИЧЕСКОЙ ШКОЛЫ «НИУ» БелГУ.....	24
3.1 Создание HTML разметки .....	24
3.2 Разработка дизайна сайта .....	26
3.3 Создание базы данных MYSQL под управлением PhpMyAdmin.....	28
3.4 Создание разделов сайта с использованием php .....	34
4 ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ И ТЕСТИРОВАНИЕ САЙТА .	40
4.1 Информационная защита сайта и сервера.....	40
4.2 Тестирование web-ресурса.....	42
ЗАКЛЮЧЕНИЕ .....	47
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ.....	49
ПРИЛОЖЕНИЕ А .....	53

## ВВЕДЕНИЕ

Развитие вычислительных средств и компьютерных технологий привело и продолжает приводить к появлению новых технологий в различных сферах научной и практической деятельности. Одним из важных достижений человечества – это создание сети Интернет. Это позволило связать между собой множество компьютеров. Сеть Интернет позволяет людям общаться между собой, обмениваться опытом и информацией. В Интернете, на сегодняшний день, существуют почти 2 миллиарда сайтов, содержащие информацию во всех областях (данные за 2019 год).

Интернет – всемирная сеть, объединившая множество компьютерных сетей различного уровня и отдельных вычислительных средств, позволяющая им обмениваться информацией по каналам общественной коммуникации на базе протоколов связи TCP/IP.

Для хранения информации требуются сервера. Они в свою очередь, объединены высокоскоростными магистралями и, тем самым, составляют базовую часть сети. Пользователи получают доступ к информации с помощью провайдеров и корпоративных сетей. С каждым днём количество пользователей, использующих информацию из интернета растёт. На сегодняшний день насчитывается свыше 4 миллиардов пользователей. Помимо общедоступной информации Интернет предоставляет множество способов обучения пользователей. Образование – процесс передачи систематизированных знаний, навыков и умений от одного человека к другому.

Современные технологии обучения требуют наличия удалённого обучения с использованием Интернета и IT-технологий. Один из вариантов, как можно реализовать данное обучение – связан с созданием заочных школ по принципу заочных школ обучения.

Многие образовательные организации на сегодняшний момент имеют свои собственные обучающие сайты.

Данные ресурсы обеспечивают взаимодействие между преподавателями и учениками. Первый должен иметь возможность выслать задания пользователям, вторые же должны иметь возможность выслать решение данного задания. Так же по выставленным оценкам должна вестись статистика по каждому пользователю.

Целью выпускной квалификационной работы является проектирование и разработка Web-сайта заочной математической школы, с использованием средств php и базы данных MySQL под управлением системы PhpMyAdmin.

Для достижения поставленной цели было необходимо решить следующие задачи:

1. Рассмотреть предметную область и провести обзор аналогов web-ресурса;
2. Сформировать требования к структуре сайта в соответствующих методических документах;
3. Выполнить проектирование web-ресурса с использованием средств php и MySQL;
4. Провести тестирование и апробацию разработанного web-ресурса.

**Структура и объем работы:** выпускная квалификационная работа выполнена на 64 страницах машинописного текста, состоит из введения, четырёх глав, заключения и приложения.

В первой главе проводится анализ предметной области и состоит из четырёх подразделов. В главе описываются поставленная задача, характеристика информационных ресурсов и их роль в обучении, аналоги разработки и выбор средств для разработки web-ресурса.

Во второй главе описывается проектирование web-ресурса для математической школы, и состоит из двух подразделов. В главе описывается проектирование структуры базы данных и структуру базы данных.

В третьей главе проводится разработка web-ресурса заочной математической школы и состоит из четырёх подразделов. В главе описывается создание HTML разметки, разработка дизайна сайта, создание базы данных MYSQL под управлением PhPMyAdmin, создание разделов сайта с использованием php.

Четвёртая глава посвящена информационной безопасности и тестированию сайта. В главе говорится об информационной безопасности и тестировании web-ресурса.

# 1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

## 1.1 Описание поставленной задачи

Зачастую у многих из обучающихся в школе нет возможности посещать учебное заведение по разным причинам: по состоянию здоровья, нахождению в другом городе. Преподавателю же сложно работать в живую с большим количеством учеников в ограниченном промежутке времени. Для того чтобы решить две эти проблемы нужно создать ресурс, который позволит дистанционно обучать учеников удалённо, а также сделает этот процесс более непрерывным. А для тех, кто посещает школу данный ресурс может быть использован для получения домашнего задания, дополнительного образования или продления учебного процесса.

Такой ресурс должен позволять наладить коммуникацию между преподавателем и учениками. Преподаватель должен иметь возможность высылать учащимся школы, материалы для их обучения, а также задания, которые им нужно будет выполнить к определённому сроку.

Так же такой образовательный ресурс должен отображать общую успеваемость учеников, как для преподавателей, так и для всех, кто посещает сайт. Зная эту информацию, преподаватель может видеть, как справляются ученики школы и корректировать выдаваемые им задания и сроки их выполнения, а также выставлять итоговую оценку в конце каждого этапа обучения. Те же, кто посещают сайт будут видеть насколько эффективно ведётся обучение на данном ресурсе.

Для того, чтобы зашедшего на сайт пользователя заинтересовала предоставляемая информация по обучению в данной школе, сайт должен давать краткое описание своей деятельности, а также вести новостной блог, в котором будет отображаться последние новости школы.

Таким образом, всё причисленное сможет решить поставленные задачи, а именно оптимизировать обучение, вести статистику об успеваемости каждого ученика и заинтересовать новых пользователей записаться на обучение именно в эту заочную математическую школу с удалённым обучением.

## 1.2 Характеристика информационных ресурсов и его роль в обучении

Перед тем как приступить к разработке сайта, необходимо познакомиться с основными понятиями и терминами, и средствами разработки. Сайт – это набор страниц в формате HTML, которые лежат на Web-сервере.

Web – система гиперссылок, связывающая узел в единое целое, и оформление узла, и современное обновление находящейся информации на узле, а так же скорость загрузки отдельных страниц.

Главное отличие Web документов от обычных электронных документов это наличие гиперссылок, которые помогают просматривать Web документ, или обратиться к одной из служб Интернета. Такие гиперссылки обычно выделяются синим цветом, а также могут иметь подчёркивание. Если кликнуть на такую ссылку, то произойдёт переход на страницу, указанную в гиперссылке. При возврате на предыдущую страницу она будет окрашена вишнёвым цветом, это будет означать, что по этой ссылке уже переходили. При переходе между страницами сайта также используются гиперссылки, но при этом выделение цвета скрывается, так как оставить выделение цветом, будет плохим решением с точки зрения дизайна сайта.

Домашняя страница – это web-страница, которая передаётся сервером при запросе пользователя, указавшего в запросе имя Web страницы без указания какого-либо конкретного его раздела. Домашняя страница является лицом сайта, именно поэтому ей уделяется особое значение, как в проектирование дизайна, так и в проектирование дальнейшего перехода на разделы сайта.

Сайты образовательных учреждений относятся к группе образовательных сайтов. Под образовательным ресурсом, понимается web-сайт, состоящий из совокупности web-страниц с повторяющимся дизайном, несущий в себе целенаправленный процесс обучения, объединённых по



смыслу, навигационно и физически находящихся на одном сервере, использование которых может сопровождаться аттестацией обучающихся. На сайте можно просматривать новости, опубликованные одним из преподавателей.

Всё это будет доступно всем пользователям Интернета, которые зайдут на сайт. Регулярные проводимые социальные эксперименты показывают, что первой впечатление о школе, пользователи получают по его при посещении сайта. А значит целесообразно расположить важные новости на главной странице.

Те, кто поступят в школу получают возможность обучаться у себя дома или в дали от самого учебного учреждения.

Дизайн образовательного интернет-сайта – процесс выбора и организации графических компонентов с целью достичь определённого результата, будь то эстетика, либо достижение определённого функционала, но зачастую преследуются и первое, и второе.

Техническая реализация образовательного интернет-сайта – выбор компонентов, интегрирование технологии, программных продуктов и технических средств, для предоставления образовательной информации для пользователя.

Эксплуатационность образовательных интернет-сайтов – характеристики, информирующие об использовании образовательных сайтов пользователями.

### 1.3 Аналоги разработки

Для рассмотрение существующих сайтов заочных математических школ выбраны следующие ресурсы: математическая школа имени Софьи Ковалевской расположенной по ссылке школа.белгу.рф, заочная физико-техническая школа МФТИ, московский центр непрерывного образования.

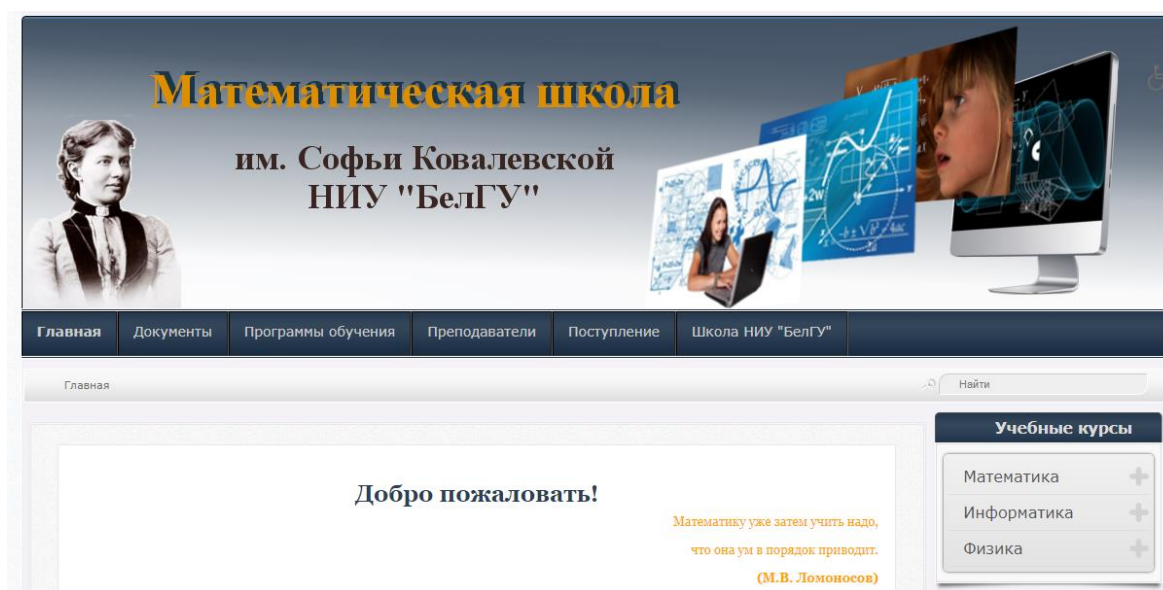


Рисунок 1.3.1 Математическая школа Софьи Ковалевской НИУ «БелГУ»

Первым рассмотрим сайт математической школы имени Софьи Ковалевской (рисунок 1.3.1). Данный сайт встречает пользователя приветствием и кратким описание данной школы. Также переходя на разделы сайта, можно подробнее узнать об образовательной организации. Сайт имеет следующие разделы: Главная, Документы, Программы обучения, преподаватели, условия поступления, Школа НИУ «БелГУ».

В разделе «Документы», предоставлена концепция развития математической школы и положение о школе НИУ «БелГУ». Данные документы представлены в виде двух ссылок, при переходе по которым, открывается pdf файл.

Раздел «Программы обучения», предоставляет три раздела: математика, физика и информатика. В каждом из этих разделов находятся курсы, на которые может записаться пользователь. Каждый курс имеет своё описание, а также переход на обучение по данному курсу, представленный в виде кнопки. Нажав на кнопку «Перейти на курс», пользователю предоставляется список доступных курсов, перейдя в каждый из которых, предлагается вход или создание новой учётной записи, так же некоторые курсы имеют возможность входа пользователя Интернета, как «гостя».

Во вкладке «Преподаватели», перечислен весь список преподавателей, которые учувствуют в образовательной деятельности. Здесь же указывается образовательное учреждение, в котором работает данный преподаватель, его специальность, квалификация, должность и опыт работы.

Вкладка «Поступление», предоставляет подробную информацию о приёме заявок в данный университет, а также этапы поступления. Снизу предлагается пройти регистрацию, в формате кнопки с надписью «Регистрация», при нажатии на которую, пользователь перенаправляется на тот же сайт, что и при выборе курса, во вкладке «Программы обучения».

Вкладка «Школа НИУ БелГУ», перенаправляет на отдельный большой раздел, имеющий свои собственные разделы, в каждом из которых предоставленная информация о данной школе. Данный раздел делится на: раздел, предоставляющий сведения об соответствующих подразделениях, новости сайта, образовательный процесс, галерею фотографий.

Рассмотренный сайт предоставляет всю информацию о школе НИУ «БелГУ», по которой может заинтересовать пользователя, записаться на курсы в данную школу. Так же сайт имеет удобный выбор курса и регистрацию. Имеет удобную и понятную структуру. Но данный сайт лишь предоставляет информацию о физической школе, на курсы которой можно записаться.

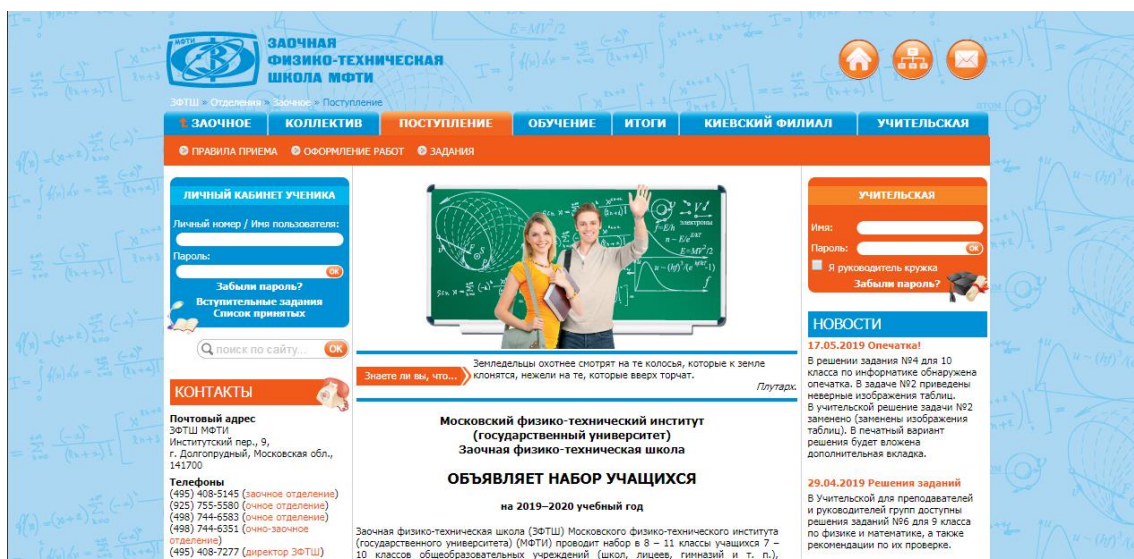


Рисунок 1.3.2 Зачая физико-техническая школа МФТИ

Следующий на рассмотрении web-ресурс – это зачая физико-технической школы МФТИ (рисунок 1.3.2). Данный сайт встречает пользователя списком новостей сайта расположенные по центру сайта. Справа от новостей расположена регистрация преподавателя и небольшое окно новостей связанных с процессом обучения в данном web-ресурсе. Слева располагается регистрация пользователя, как ученика данной зачной физико-технической школы. Ниже расположены средства связи с представителями данной школы, литература, которую сайт предлагает прочесть перед поступлением, и небольшой список форумов, связанный с данной зачной школой.

Web-ресурс поделён на 7 разделов: «Новости», «О школе», «Отделения», «Обучение», «МФТИ школьникам и сервер». Первый раздел предназначен для вывешивания новостей, связанных с процессом образования, публикуемыми заданиями и их проверкой.

Раздел «О школе» предоставляет краткую информацию о школе. Здесь же располагаются ссылки на учебно-методические пособия и устав школы. В окладке «отделения» располагаются ссылки на очное, очно-зачное и зачное отделения, а так же вывешена статистика количество

учеников в каждом классе, в каждом отделении за последний учебный год. При переходе по ссылке отделения, пользователя находит информацию об отделении.

Вкладка «Обучение» предоставляет переходы на разделы, связанные с программами обучения, план работы лектория, информацию для родителей, список рекомендуемой литературы по физике и математике и информацию для учителей.

Раздел «МФТИ школьникам» содержит мероприятия запланированные на данный год. Так же можно перейти на каждое из мероприятий для получения источников информации.

Во вкладке «Сервер» можно узнать информацию о сервере, а так же получить удобную навигацию по сайту.

В связи всем сказанным о web-ресурсе заочной физико-технической школы МФТИ можно сделать следующие выводы. Данный сайт предоставляет возможность пройти обучение на самом сайте, а так же даёт пользователю Интернета подробную информацию как о заочной школе, так и об отделениях очного и очно-заочного обучения. Среди значительных минусов ресурса следует выделить избыточное количество вкладок и переходов. Из-за данной проблемы сайт использует карту сайта, но при этом отсутствует интуитивно понятное перемещение по сайту. Отсутствует открытая регистрация на ресурсе.



Рисунок 1.3.3 Московский центр непрерывного математического образования

Московский центр непрерывного математического образования встречает пользователя разделами сайта (рисунок 1.3.3), ссылками на ресурсы предоставляемые сайтом связанные с обучением, новостями школы, семинарами и п. д. и кратким списком новостей.

Главная страница состоит из ссылок, которые перенаправляют либо на отдельную страницу сайта, либо на отдельный web-ресурс. При переходе на отдельную страницу сайта сохраняется стилистика, но навигация по сайту отсутствует.

Панель навигации состоит из разделов: цели и задачи, попечительский совет, партнёры МЦНМО, наши координаты, сведения об образовательной организации. Эти разделы являются непосредственной частью сайта, остальные же являются ссылками на отдельные web-ресурсы, которые отвечают за ту или иную деятельность сайта.

Каждый из разделов предоставляет определённую информацию об школе и обучение в ней. Также каждый из разделов сохраняет навигацию по сайту и единый стиль.

Отделы, которые переносят на отдельные web-ресурсы, предназначены непосредственно для обучения. Такие ссылки перенаправляют на ресурс, связанные с математикой или геометрией.

Уже на данном этапе можно оценить сайт как образовательный. Из плюсов можно выделить структурированную информацию о школе. Из минусов, не все разделы сайта имеют навигацию, сайт разделён на отдельные web-ресурсы, что лишает сайт единой структуры и собственной обучающей системы. Большинство из ссылок, предоставляющие информацию об учебном заведении, так же являются отдельными ресурсами, что также лишает сайта удобного перемещения и единой структуры.

#### 1.4 Выбор средств для разработки Web-ресурса математической школы

Выбор средств для разработки сайта для математической школы должен быть обоснован его функционалом. Выбранные средства должны полностью удовлетворять требованиям сайта. Не следует выбирать средства с избытком функций, которые в дальнейшем могут быть так и не использованы.

Так как наиболее важным аспектом сайта заочной математической школы является обучение учеников, достаточно выбрать два основных средства для его реализации: средство для проектирования функционала сайта и средства создания базы данных.

Интерфейс сайта, должен обеспечивать удобное взаимодействие с ресурсом. Для достижение этой цели первое, что потребуется, это создать разметку сайта, а значит нам требуется язык разметки сайта HTML. Для того, чтобы получившаяся разметка имела эстетический вид потребуется таблица стилей. Следующее, что потребуется для проектирования функциональности сайта – это язык программирования. На данный момент php, как язык программирования устаревает, и постепенно переходят на javascript или python, но их использования достаточно, для выполнения всех функций, которые требуются от образовательного сайта. Так же из основных достоинств php является: открытый доступ, объектно-ориентированность, что позволяет сэкономить время и ресурсы по разработке сайта, гибкость и хорошая совместимость, масштабируемость и документированность, бесппроблемная поддержка и возможность модифицировать код.

Используемость данных средств требует наличия сервера. Для запуска сайта на php требуется локальный сервер, так как сайт будет перенесён на сервер университета, поддержка его в разных операционных системах не так важна. Среди самых популярных локальных серверов можно выделить такие,

как Denwer, Xampp, AppServ, VertrigoServ, Zend, Open server, Wampserver, EasyPHP.

Denwer – это бесплатный отечественный сервер, предназначенный для работы с web-приложениями и Интернет страницами. Данный продукт имеет необходимые дистрибутивы для упрощённой работы. Одним из примечательных особенностей – это встроенная панель phpMyAdmin и MySQL для работы с базами данных.

Xampp – специальная сборка от Apache, с необходимыми дистрибутивами для запуска полноценного веб-сервера. Главным преимуществом является простой и удобный интерфейс.

AppServ – тайский сервер, концепт которого является лёгкая и быстрая установка, которая длится 1 минуту. Так же, как и выше перечисленные, средства AppServ имеет достаточное количество дистрибутивов для запуска веб-сайта.

Zend Server Community Edition – простой локальный сервер для запуска веб-приложений.

Open server – так же как и denwer, является отечественной разработкой с мульти язычным интерфейсом. Имеет большой функционал, а именно: MySQL, MariaDB, MongoDB, PostgreSQL, PHP от 5 до 7 версии, phpMyAdmin и много других функций.

Wampserver – локальный сервер с русским интерфейсом. Имеет удобную настройку, которая не затрагивает файлы конфигурации, что упрощает работу для начинающим веб-мастерам.

EasyPHP – простой локальный сервер для тестирования небольших проектов.

Из всех рассмотренных вариантов предпочтительнее выбрать denwer, так как он уже в себе имеет встроенную панель для работы с базами данных, что упростит работу, избавив от необходимости создавать соединение с внешней базой данных, так же phpMyAdmin менее нагружает систему, в отличии, например, от Microsoft MySQL. Open server может служить



альтернативой Denver, но для создания сайта математической школы, все его функции не потребуются.

Также требуется редактор для написания html разметки и кода на php.

## 2 ПРОЕКТИРОВАНИЕ WEB-РЕСУРСА ДЛЯ МАТЕМАТИЧЕСКОЙ ШКОЛЫ

### 2.1 Проектирование структуры сайта

Перед решением основных задач, следует разработать структуру Web-ресурса. Для этого следует выделить основные разделы и установить связь между ними.

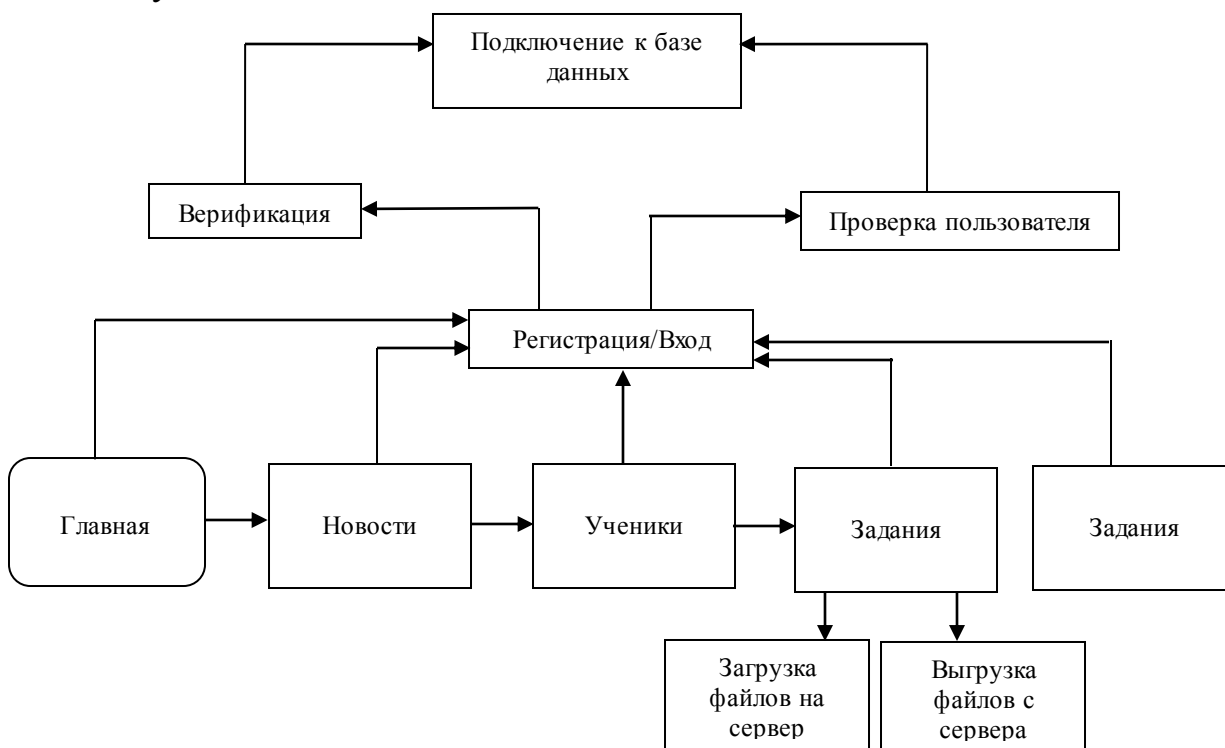


Рисунок 2.1.1 Схема структуры сайта

На рисунке 2.1.1 представлена схема структуры сайта. Вкладка «Главная» является основной и запускается при переходе по адресу сайта, без указания конкретного раздела. При переходе на главную страницу, пользователя встречает приветствие, расположенное после строки разделов. Поле идёт описание сайта, в которой описана основная деятельность школы, устав школы, а также способы связаться с представителями школы.

На вкладке «Новости» расположены новости сайта, которые может опубликовать один из преподавателей. Данная вкладка состоит из списка

новостей. Каждая новость состоит из заголовка и описания, так же могут прилагаться картинки и прикрепленные файлы.

«Ученики» раздел, в котором выводится статистика о успеваемости учеников школы. В данном разделе выводится фамилия, имя, класс, почта и оценки полученные за четверть. Вся информация, помимо электронной почты, доступна каждому пользователю Интернета, кто посетит Web-ресурс данной заочной математической школы. Почта же доступна только преподавателю.

Во вкладке задания располагается поле, предлагающее пользователю (ученику или преподавателю) выложить файл на данный Web-ресурс. Преподавателю предлагается выслать задание определённой группе учеников. После чего отправленные файлы будут отображаться в списке загруженных. Ученику же предлагается загрузить решение на полученное от преподавателя задание. После чего оно также будет отображаться в указанном списке отправленных файлов. Как преподаватель, так и ученик видит список полученных файлов и может загрузить их на своё устройство.

Во вкладке «Преподаватели» располагается подробная информация о преподавателях данной заочной математической школы. Информация представлена в виде заголовка, в котором располагается ФИО преподавателя, его фотография и описание. В описании представлена информация о должности, специальность, квалификация, опыт в преподавании.

Регистрация разделена на два подраздела: регистрация для ученика, регистрация для преподавателя. При переходе на регистрацию как ученик, пользователю предлагается ввести следующие данные: имя, фамилия, класс, электронная почта, пароль. При переходе на регистрацию, как преподаватель, пользователь должен ввести своё ФИО, а так же подробную информацию о себе: образовательное учреждение, специальность, квалификация, должность, опыт преподавания. После пользователь вводит e-mail и пароль. Введя всю перечисленную информацию, пользователь должен подтвердить, что он преподаватель. Для этого он вводит ключ, выданный им ранее, при

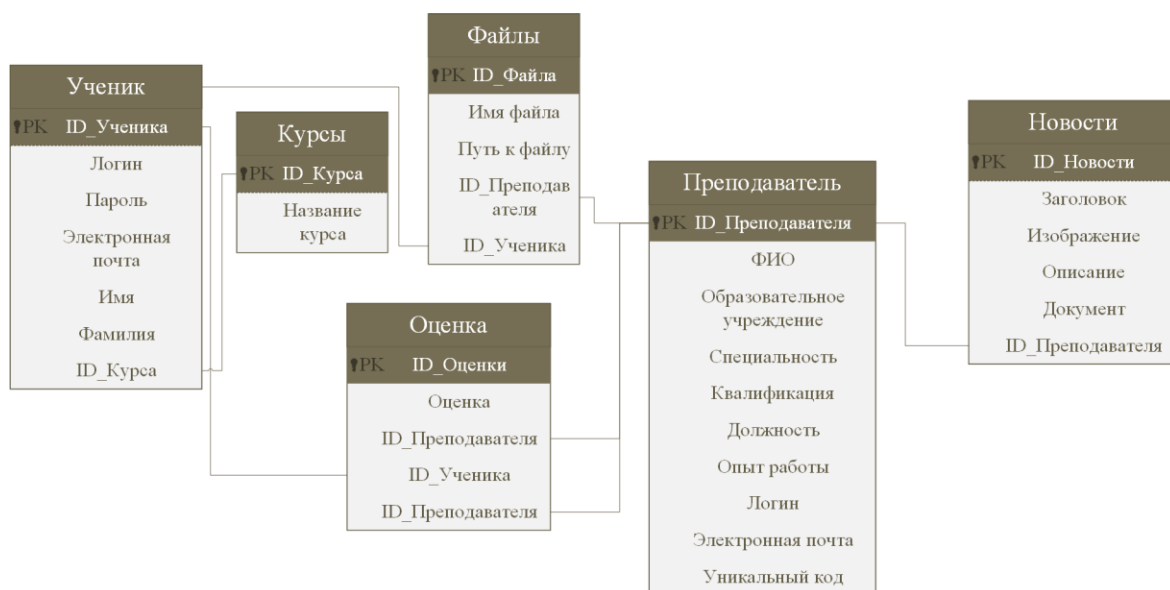
совпадении его с ключом расположенным на сервере, пользователь считается преподавателем.

В структуре сайта имеются и скрытые от пользователя отделы сайта. Проверка на существоющего пользователя, сверяет информацию пользователя, введённые им при входе на сайт с данными введёнными при регистрации. Данный раздел возвращает информацию вошедшего пользователя. Для проверки существоющего пользователя используется подключение к базе данных. При регистрации используется верификация и она также использует связь с базами данных. В данном разделе располагается запись в базу данных информации о пользователях используемых как для входа, так и для вывода информации о пользователе. Данный раздел предназначен и для обработки исключений, возникающих в результате регистрации.

## 2.2 Разработка структуры базы данных

Web-ресурсу заочной математической школы, как и остальным web-ресурсам хранящих большое количество информации, требуется структурированное хранилище данных. Для решение данной проблемы существуют различные базы данных. Но перед тем как перейти к её разработке, требуется спроектировать структуру базы данных.

Для разрабатываемого сайта заочной математической школы были выделены следующие основные объекты: ученики, преподаватели, новости, задания, ответы. Также использовались дополнительные объекты: выставяемые оценки, курсы, идентификационный ключ. Структура базы данных представлена на рисунке 2.2.



2.2 структура базы данных.

Объект «Ученик» отвечает за зарегистрированного пользователя, предоставившего в качестве информации о себе, своё имя и фамилию, класс, в котором он на данный момент учится или окончил, электронную почту, логин, под которым он в дальнейшем будет авторизоваться на web-ресурсе. Данному объекту присваивается уникальный ключ. Объект «ученик»

привязан к объектам: курс, преподаватель, ответы. Заполнение информации о преподавателе и курсе происходит после записи на определённый курс, размещённый на web-ресурсе заочной математической школы.

Объект «Преподаватель» отвечает за зарегистрированного пользователя, предоставившего во время регистрации в качестве информации о себе: своё имя, фамилию, отчество, образовательное учреждение, специальность, квалификацию, должность и опыт работы. Также каждый преподаватель имеет уникальный ключ и уникальный идентификационный номер, который подтверждает, что был зарегистрирован преподаватель, а не просто пользователь Интернета. Идентификационный номер, введённый при регистрации, сверяется с тем, что указан в таблице «идентификационный ключ». Объект «Преподаватель» связан с объектами, курс, задания, ответы, новости.

Объект «Новости» отвечает за хранение публикуемых новостей web-ресурса. В данном объекте хранится заголовок новости, текст новости и путь к изображению, расположенном на сервере. После публикации новости, в данный объект записывается ключ, того, кто опубликовал данную новость.

Объект «Задания» отвечает за отправляемые ученикам материалы по учебной программе. В него записывается имя файла и его месторасположение на сервере. Также объекту «Задание» присваивается ключ преподавателя отправившего задание ученикам.

Объект «Ответы» отвечают за отосланные учениками решения задач. В этом объекте также расположено имя файла, его путь и ключ ученика, который отправил это задание преподавателю.

В объекте «Оценки» записываются баллы поставленные преподавателем в результате оценки работы преподавателем. Каждой оценке присваивается ключ ученика, которому принадлежит данная оценка.

Курсы предлагаемые web-ресурсом расположены в соответствующем объекте «Курсы». В нём располагается название курса и его описание.

«Идентификационный ключ» хранит наборы символов и связаны с преподавателями. При использовании ключа во время регистрации преподавателя, данный ключ больше не будет использован и будет удалён из таблицы.

## 3 РАЗРАБОТКА ВЕБ-РЕСУРСА ЗАОЧНОЙ МАТЕМАТИЧЕКОЙ ШКОЛЫ «НИУ» БелГУ

### 3.1 Создание HTML разметки

На основе разработанной структуры web-ресурса была создана html разметка сайта. Первым этапом был разработан шаблон для каждого из разделов сайта. Шаблон состоит из заголовка `<header>`, в котором располагается название школы, название университета, которому принадлежит данная школа и эмблема. После заголовка построено тело web-ресурса `<body>`. В котором располагается меню разделов сайта: главная, новости, студенты, задания, преподаватели. Для создание меню разделов использовались следующие теги: `<div>`, `<ul>`, `<li>`. Данные теги будут использованы для оформления навигационной панели разделов сайта. Структура шаблона раздела представлена на рисунке 3.1.

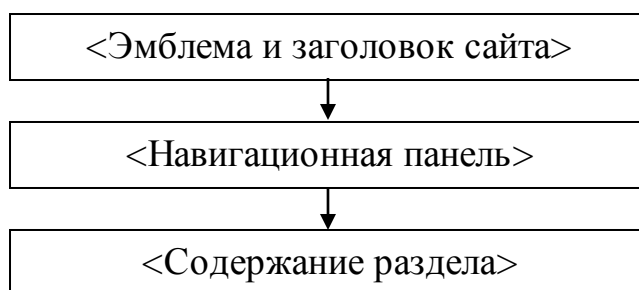


Рисунок 3.1.1 Структура шаблона раздела

После создание каркаса сайта были созданы копии получившейся страницы, они и будут являться разделами сайта. После созданий копий каркаса, были созданы ссылки для перехода между разделами web-ресурсами, добавив после тега `<li>` тег `<a>` и добавив свойство `href` в качестве параметра указав название файла соответствующему разделу сайта.



В каждом разделе сайта выше навигационной панели были добавлены кнопки для входа и регистрации пользователя, как преподавателя и пользователя как ученика школы.

Для входа были созданы две страницы, на которых расположены поля для ввода логина и для ввода пароля. Для полей регистрации были размещены соответствующее количество полей, в зависимости от того регистрируется ученик или преподаватель. Для создание разделов регистрации и входа используется тег `<form>`, отвечающий за форму элементов расположенных на странице. В качестве полей используются теги `<input>` с типом `text` и тег `<input>` с типом `button`. Тег с типом `text` будет отвечает за ввод информации, а тег с типом `button` используется для отправки данных. После создания разделов входа и регистрации, были созданы ссылки для перехода на них, аналогично переходу на раздел сайта.

На главной страницы расположена информация о школе со вставкой тега `<a>`, в котором располагается ссылки на устав школы. Данная информация не требует подключения к базе данных, так как в дальнейшем не будет изменяться.

### 3.2 Разработка дизайна сайта

Для создания дизайна сайта была использована таблица стилей CSS. Для её использование в файле с html разметкой был расположен тег `<link>` со следующими свойствами: `rel` равным `'stylesheet'`, `href` с указанием файла отвечающего за таблицу стилей и `type` равным `'text/css'`.

Таблица стилей строится из название тегов, которые будут учувствовать в создании дизайна сайта. Так как в создании html разметки используются повторяющиеся элементы, при указании некоторых тегов используются классы, которые будут использоваться в качестве свойства объекта с указанием имени данного класса.

В первую очередь были указаны параметры по умолчанию, которые будут применяться ко всей странице, в тех случаях если не используется тег, в дизайне которого будут переопределены одно или несколько из этих свойств. Для этого заданы следующие параметры для тегов `<html>` и `<body>`: отступы (сверху, справа и слева), семейство используемых шрифтов и цвет заднего фона. Были созданы отступы в верхней части сайта в 10px, для этого использовался параметр `margin-top`. С помощью `padding-left` и `padding-top` были созданы отступы слева и справа соответственно и присвоены значение в 40 пикселей. С помощью `font-family` были добавлены следующие шрифты: Geneva, Arial, Helvetica, sans-serif. В качестве заднего фона был выбран белый цвет, для этого был прописан параметр `background-color` со значение `white`.

Далее были оформлены заголовки и эмблема. Для этого в качестве параметра изображения были выставлены следующие параметры: `align` равная `left`, отвечающий за расположение слева, `width` равный 120 и `height` равный 150 отвечающие за длину и высоту соответственно, `vspace` и `hspace` равные 1, что позволяет расположить заголовки и эмблему на одном уровне. Для создания дизайна заголовка была использована таблица стилей и тег

<div>. В таблице стилей был создан классы title1 и title2. В них прописан отступ слева, шрифт, размер текста, тип текста, вертикальная ориентация и цвет текста.

Для создание навигационной панели так же был использован тег <div>, но с параметром class равным main-menu. В данном классе описывается ряд параметров, которые в совокупности отвечают за дизайн навигационной панели. Первые из них это отступы сверху, слева и справа, белый цвет текста на оранжевом поле, расстояние между названиями разделов. Навигационная панель располагается горизонтально, а сами разделы выполнены в форме блоков. Так же в свойствах данного класса указано скрывание гиперссылки для создания иллюзии кнопки. Для визуального эффекта добавлено подчеркивание при наведении мышью на название раздела web-ресурса. Все разделы размещены слева и разделены чёрной линией.

Описание школы состоит из заголовка и текста описания. Заголовок выровнен по центру и имеет жирный шрифт. Для создания заголовка был использован тег <h> с параметрами размера текста и font-weight равным bold. Для создания дизайна текста описания школы был использован тег <div> с параметром class равный descriptionSchool. В параметрах класса указано выравнивание по ширине, перенос текста с новой строки, отступ при каждом абзаце.

Оформлены были и кнопки регистрации для этого присвоен класс кнопка1 тегу <form>. Класс кнопка1 состоит из трёх частей: оформление при наведении курсора, оформление при наведении курсора, оформление при взаимодействии с кнопкой. В первой части указан цвет фона, форму, тени, выравнивание, отступы, размеры, цвет текста, шрифт, тип текста и цвет обводки текста. Во втором разделе класса определяем находится ли курсор в кнопке и, если это так, то обводка окрашивается во второй цвет. И в третьей части описан дизайн при нажатии на кнопку, а именно уменьшение размера кнопки и увеличение, также изменяется цвет фона кнопки.

### 3.3 Создание базы данных MySQL под управлением PhpMyAdmin

Для web-ресурса образовательного типа требуется хранилище данных, для удобной и структурированной информации требуется использовать базу данных. В данном хранилище можно создать объекты и объединить их связью между собой. В качестве СУБД была выбрана встроенная в локальный сервер Denwer PhpMyAdmin. Разработанная база данных представлена на рисунке 3.3.2.

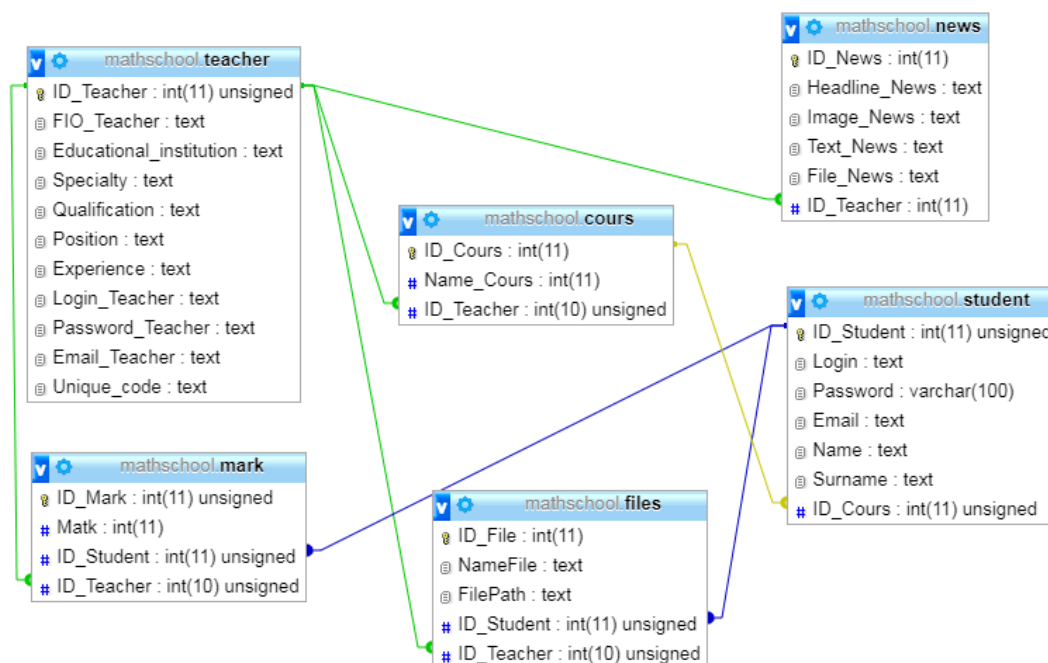


Рисунок 3.3.2 Схема структуры базы данных

Были выделены следующие объекты: user (ученик), teacher (преподаватель), news (новости), course (курс), mark (оценка), files (файлы). Описание объектов расположено в таблице 3.1, которое состоит из названия поля объекта, типа данных и описания.

Таблица 3.3.1 – описание базы данных

№	Таблица Student		
	Поле	Тип	Описание
1	2	3	4
1	ID_Student	Int	Первичный ключ ученика
2	Login	Text	Логин, под которым ученик заходит на веб ресурс.
3	Password	Text	Пароль, для входа на сайт.
4	Email	Text	Электронная почта ученика
5	Name	Text	Имя ученика.
6	Surname	Text	Фамилия ученика
7	ID_Course	Int	Вторичный ключ таблицы курсов.
8	ID_Teacher	Int	Вторичный ключ таблицы преподавателей.
9	Таблица Teacher		
10	ID_Teacher	Int	Первичный ключ преподавателя
11	FIO_Teacher	Text	Фамилия, имя и отчество преподавателя.
1	2	3	4

Продолжение таблицы 3.1

1	2	3	4
12	Educational_institution	Text	Образовательное учреждение, в котором работает преподаватель.
13	Specialty	Text	Специальность преподавателя.
14	Qualification	Text	Квалификация преподавателя.
15	Position	Text	Должность преподавателя.
16	Experience	Text	Опыт работы преподавателя.
17	Login_Teacher	Text	Логин, по которому будет авторизоваться преподаватель.
18	Password_Teacher	Text	Пароль, требуемый для входа преподавателя.
19	Email_Teacher	Text	Электронная почта преподавателя.
20	Unique_code	Text	Уникальный код, подтверждающий, что был зарегистрирован преподаватель.
1	2	3	4

Продолжение таблицы 3.1

1	2		
21	Таблица News		
22	ID_News	Int	Первичный ключ новости.
23	Headline_News	Text	Заголовок новости.
24	Image_News	Text	Поле хранящее место положение картинки на сервере, если он был прикреплен к новости.
25	Text_News	Text	Описание новости.
26	ID_Teacher	Int	Вторичный ключ новостей для связи с таблицей Teacher
27	Таблица Course		
28	ID_Course	Int	Первичный ключ курса
29	Name_Course	Text	Название курса
1	2	3	4

Окончание таблицы 3.1

1	2		
30	Таблица Files		
31	ID_File	Int	Первичный ключ файла
32	Name_File	Text	Имя файла с указанием его типа
33	File_Path	Text	Путь к файлу на сервере
34	ID_Student	Int	Вторичный ключ для связи с таблицей учеников
35	ID_Teacher	Int	Вторичный ключ для связи с таблицей преподавателя.
36	Таблица Mark		
37	ID_Mark	Int	Первичный ключ оценки
38	Mark	Int	Оценка
39	ID_Sudent	Int	Связь через вторичный ключ с таблицей ученика
40	ID_Teacher	int	Связь с помощью вторичного ключа с таблицей университета.



Связь между объектами была построена путём добавления вторичного ключа и приравнивание первичного ключа одного объекта к ключу второго объекта. Объект «Студент» связан со следующими таблицами: курс, файлы, оценки преподаватели. С файлами реализована связь один ко многим, так как каждый из учеников может загружать несколько файлов в неограниченном количестве. В качестве вторичного ключа выступает поле ID\_Student в таблице Files. Также загружать файлы имеет возможность преподаватель, поэтому была реализована связь между таблицей Teacher и Files. В качестве вторичного ключа у таблицы Files выступает поле ID\_Teacher.

Каждый из учеников записан на какой-либо курс, поэтому реализована связь между Course и Student один (курс) ко многим (ученикам). В качестве вторичного ключа у Student выступает ID\_Course.

К каждому курсу привязан преподаватель. В качестве вторичного ключа выступает поле ID\_Teacher в таблице Course. Преподаватель также связан с таблицей новостей со вторичным ключом ID\_Teacher.

Преподаватель выставляет оценки, значит требуется знать кто из преподавателей поставил оценку тому или иному ученику. Для этого были связаны таблицы Teacher и Mark. В качестве вторичного ключа выступает поле в таблице Mark, ID\_Teacher.

Таким образом, в течение проектирования была разработана база данных, которая готова к работе с сайтом. Создав связь между объектами в дальнейшем будет легко находить информацию о нужном объекте. Данная база данных предоставит комфортное и структурированное хранение данных.

### 3.4 Создание разделов сайта с использованием php

После разработки структуры сайта, созданного дизайна и базы данных были реализованы разделы сайта, а именно их функционал. В первую очередь была реализована регистрация и вход на web-ресурс. За вход на сайт отвечают следующие файлы RegSt.php (регистрация студента), RegTeach.php (регистрация преподавателя), VerificationSt.php, VerificationTeach.php, db.php.

Файл RegSt.php и RegTeach.php содержит поля для отвечающие за ввод информации о пользователе. После ввода данных происходит взаимодействие с файлом VerificationSt.php, VerificationTeach.php, в котором, с помощью языка программирования php, происходит проверка введённых данных, а так же их занесение в базу данных. Для этого происходит подключение к базе данных через файл db. Проверка данных происходит путём добавления скрипта и вывода сообщения при возникновении ошибки (рисунок 3.4.3). Перед тем как в базу данных будут занесены данные зарегистрированного пользователя, его пароль будет зашифрован. Благодаря шифрованию пароля, он будет храниться в базе данных в виде набора символов. Если все данные введены правильно производится запрос к в базу данных, в котором происходит обращение к объекту пользователя (ученика или преподавателя), в котором в качестве данных на заполнение передаются название переменных, советующие имени поля. Для преподавателя предусмотрено поле с вводом уникального ключа, который будет выдан ему заранее. При вводе его в поле проверяется его наличие в файле с ключами регистрации. При совпадении регистрация будет успешной. После чего ключ регистрации будет удалён из файла и в дальнейшем использоваться не будет.

```

4  ⊖ if (isset($_POST['submit'])){
5      |   if(empty($_POST['login'])) {
6          |   echo '<br><font color="red"> Введите логин!</font>';
7      |   }
8      |   elseif (!preg_match("/^\w{3,}$/", $_POST['login'])) {
9          |   echo '<br><font color="red">
10         |   В поле "Логин" введены недопустимые символы! Только буквы, цифры и подчеркивание!</font>';
11         |   }
12         |   elseif(empty($_POST['password'])) {
13             |   echo '<br><font color="red">Введите пароль!</font>';
14         |   }
15         |   elseif (!preg_match("/^\A(\w){6,20}\Z/", $_POST['password'])) {
16             |   echo '<br><font color="red">Пароль слишком короткий!
17             |   | Пароль должен быть не менее 6 символов! </font>';
18         |   }
19         |   elseif(empty($_POST['password2'])) {
20             |   echo '<br><font color="red">Введите подтверждение пароля!</font>';
21         |   }
22         |   elseif($_POST['password'] != $_POST['password2']) {
23             |   echo '<br><font color="red">Введенные пароли не совпадают!</font>';
24         |   }
25         |   elseif(empty($_POST['email'])) {
26             |   echo '<br><font color="red">Введите E-mail! </font>';
27         |   }
28         |   elseif (!preg_match("/^[a-zA-Z0-9_\.~]+@[a-zA-Z0-9~\-\.\.]+\.[a-zA-Z]{2,6}$/", $_POST['email'])) {
29             |   echo '<br><font color="red">E-mail имеет недопустимый формат!
30             |   | Например, name@gmail.com! </font>';
31         |   }

```

### Рисунок 3.4.3 Проверка введённых данных

Для входа используются файлы InSt и InTeach для ученика и преподавателя соответственно. Для входа используются два поля, логин и пароль. При вводе логина происходит обращение к файлу loginST или loginTeach в зависимости от того, кто производит вход. В данном файле происходит обращение к таблице пользователя (ученика или преподавателя) и сравнивается с переменными соответствующими названию полей. При этом происходит дешифрование пароля хранящимся в базе данных. После входа данные пользователя записываются в сессию и будут использоваться в дальнейшем.

Так как раздел главной страницы уже реализован, то следующим разделом был реализован раздел новостей. Здесь располагается список новостей, который будет автоматически заполняться при появлении новой новости. Каждый элемент списка является ссылкой на страницу новости. Сама новость состоит из заголовка, изображения (если оно прикреплено к новости), файла (если он был прикреплен) и описания новости.

С помощью таблицы стилей были оформлен заголовок и текст. Для этого был прописан тег <div> и класс headerNews, в котором определён цвет фона заголовка, крупный размер шрифта, жирный шрифт, выравнивание по центру. Для описания использовался класс descriptionNews, в котором был

указан размер текста. После чего идёт обращение к таблице News, в котором идёт обращение к полям заголовка, изображения, описания и файлу. Каждый из параметров присваивается переменной с соответствующими именам. Переменная отвечающая за заголовок передаётся тегу `<h1>...</h1>`. Переменная хранящая путь к файлу изображения передаётся в качестве параметра тега `image`, `href`. Каждый из элементов одной новости имеет параметр `href`, который перенаправляет на страницу, где расположены те же элементы, но уже в одном экземпляре. На первоначальной странице новость будет добавляться динамически, при появлении новых записей в таблице News.

При посещении сайта гостем или пользователем, который авторизовался как ученик, на вкладке новости расположены только новости произошедшие за последнее время. Если же авторизуется преподаватель, то будет показана кнопка «Добавить новость». При нажатии на которую происходит переадресация на страницу, где расположены теги `<div>` с использованием класса `headlineNews`, кнопки «Вставить изображение», на форме расположен `TextArial`, в котором записывается описание новости, кнопка «Прикрепить документ» и кнопка «Отправить новость». Прикрепление изображения и документа не обязательны. При нажатии на «Отправить новость» происходит обращение к таблице в News, в котором вносится информация в соответствующие поля. Если изображение или документ не были прикреплены, поля остаются пустыми.

Раздел «Студенты» предназначен для вывода информации о всех зарегистрированных учениках. Информация о пользователях представлена в виде таблицы со скрытыми границами. При выводе информации о пользователе происходит запрос к таблице Student, Course и Mark. Из первой таблицы выводится Фамилия, Имя, почта (видна только преподавателям). Из второй таблицы выводится название курса, где `ID_Course` в таблице Student равен `ID_Course` в таблице Course. Аналогично с помощью проверки равенства первичного и вторичного ключей в таблице Mark и Student

происходит вывод всех оценок заработанные учеников. В отличии от остальной информации, оценки будут выводиться при повелении новой записи в таблице Mark, если ID\_Mark в таблице Mark будет равен ID\_Mark в таблице Student.

Раздел «Задания» предназначен для выдачи ученикам заданий и для загрузки учениками решений этих заданий. Данный раздел должны видеть только авторизированные пользователи. Для этого проверяется существуют ли в данной сессии логин и пароль пользователя. В основном блоке раздела проверяется кому принадлежат логин и пароль. Если логин и пароль принадлежат преподавателю, то ему предоставляется возможность загрузить задания каждой из групп. После чего происходит отображение, какое задание какой группе принадлежит. Если решение было загружено учеником преподаватель видит, кем оно отправлено, кнопку для сохранения решения, поле для оценивание задания и кнопку подтверждения. При введении значения оценки и подтверждения действия, происходит запрос к таблице Mark, в котором отправляется значение этой оценки и ID пользователя в поле ID\_Student. После занесения оценки в базу данных, поле для ввода оценки и кнопка подтверждения скрываются.

Если авторизовался ученик, то ему отображаются задания отправленные преподавателем, а так же видно от кого именно пришло задание. При нажатии на названия файла задания предлагается сохранить файл, если в настройках указана папка для сохранения файлов по умолчанию, то файл сохранится автоматически. Ниже загруженного файла расположено поле для загрузки решения задания. При нажатии на кнопку «Выбрать файл», открывается диалоговое окно, в котором нужно выбрать тот файл, которых хочет отправить пользователь. После пользователю будет отображён файл, который он отправил преподавателю и поле «Оценка», которое буде заполнено, после того как преподаватель введёт её при получении файла. Так же оценка будет отображаться в разделе «Студенты».

При отправке файла идёт переадресация к файлу STask.php. В котором происходит подключение базе данных через файл db.php, для занесение информации о загруженном файле. Далее происходит обращение к папке на сервере, в которую будут сохраняться файлы. После чего происходит попытка подключения к базе данных. При удачном подключении выполняются два запроса. Первый запрос записывает в переменную ID пользователя, ученика или преподавателя, в зависимости от того, кто загружает файл. После получения ID проверяется кем является пользователе и в зависимости от того ученик или преподаватель загружает файл, строится второй запрос. Второй запрос предназначен для таблицы Files. В нём идёт обращение к полям: имя файла, путь файла ID\_Student или ID\_Teacher, в зависимости от того, кто загружает файл. В качестве параметров передаются значение переменной, хранящей в себе место положения файла и имя файла, а также ID пользователя, полученное в первом запросе (рисунок 3.4.4). После записи данных о файле в базу данных, происходит переадресация на главную страницу.

```

33 $uploaddir = './files/';
34 $login = $_SESSION['login'];
35 $password = $_SESSION['password'];
36 $id_user = (int) mysqli_query($link, "SELECT ID_User FROM User WHERE Login = '$login' AND Password = '$password'");
37 $uploadfile = $uploaddir . basename(iconv("utf-8", "cp1251", $_FILES['userfile']['name']));
38
39 echo '<pre>';
40 if (move_uploaded_file($_FILES['userfile']['tmp_name'], $uploadfile)) {
41     echo $f1="Файл корректен и был успешно загружен.\n";
42 } else {
43     echo $f2="Возможная атака с помощью файловой загрузки!\n";
44 }
45 if(isset($_POST['save'])){
46     $path="upload/";
47     $name = $_FILES['userfile']['name'];//Name of the File
48     $temp = $_FILES['userfile']['tmp_name'];
49     if(move_uploaded_file($temp, $path . $name)){
50         echo "success";
51     }else{
52         echo "failed";
53     }
54 }
55 $NameFile = $_FILES['userfile']['name'];
56 if (isset($NameFile)) {
57     //Вставляем данные, подставляя их в запрос
58     $sql = mysqli_query($link, "INSERT INTO Files (NameFile, FilePath, ID_User) VALUES ('$NameFile','files/$NameFile','$id_user')");
59     //Если вставка прошла успешно
60     if ($sql) {
61         echo '<p>Данные успешно добавлены в таблицу.</p>';
62     } else {
63         echo '<p>Произошла ошибка: ' . mysqli_error($link) . '</p>';
64     }
65 }

```

Рисунок 3.4.4 запись информации о файле в базу данных

При выборе учеником или преподавателем файл из списка для скачивания, будь то решённое задание или само задание, обращение к папке с загруженными файлами. В нём с помощью цикла выводятся файлы из директории расположенной на сервере, в которой хранятся все загруженные файлы. Каждому из выведенного списка присваивается свойство href, в котором прописывается полученный путь файла, благодаря чему каждый элемент списка автоматически становится ссылкой на конкретный файл.

Последним разделом является страница преподавателей. На данной странице располагается информация о преподавателях. В неё входит: своё имя, фамилию, отчество, образовательное учреждение, специальность, квалификацию, должность и опыт работы. Шаблон информации о преподавателе состоит из тегов <div>, <ul> и <li>. В первых тегах <li> располагается статический текст выделенный жирным шрифтом: ФИО, образовательное учреждение, специальность, квалификация и опыт работы. Справа располагаются теги, которые будут заполнены через базу данных.

В данном файле происходит подключение к базе данных, через файл db.php. После чего происходит 5 запросов, в результате которых были записаны в переменные соответственно статическому тексту. После чего переменные записываются в правые теги <li>.

## 4 ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ И ТЕСТИРОВАНИЕ САЙТА

### 4.1 Информационная защита сайта и сервера

В наше время не редкость атаки на Web-ресурс. Чтобы обеспечить целостность информации web-ресурса требуется выполнить ряд мер по защите информации и сохранения работоспособности сайта при возможной атаке.

В первую очередь была обеспечена защита информации и обеспечена защита паролем пользователей от взлома. Пароль не должен храниться в базе данных таким, каким вводил его пользователь. Поэтому при регистрации пользователя идёт шифрование пароля. В результате чего пароль в базе данных хранится в виде несвязанных между собой случайных символов. Для того, чтобы вытащить пароль пользователя потребуется дешифратор, который используется при входе пользователя, чтобы сравнить введённый им пароль, с тем что записан в базу данных.

Для того, чтобы сохранить работоспособность сайта при атаке, требуется, в первую очередь создать, копию сайта, которая будет запущена при атаке на сайт. После чего будет производиться ограничение потока на web-ресурс, в результате чего будет восстановлена работа сайта.

Ещё одно уязвимое место – это загрузка файлов на сервер. В первую очередь, следует разрешить загрузку файлов только формата текста и изображения, а для этого выделим основные типы файлов: doc, docx, pdf, png, jpg. Это самые часто используемые форматы текста и изображений. Второе, что нужно обеспечить при загрузке файлов на сервер web-ресурса, это защита от файловой атаки на сервер. Файловая атака грозит переполнением файлами сервера, а так же нагрузкой на сервер, что может вызвать перебои в работе сайта.



Для того, чтобы обеспечить защиту от файловой атаки производится ограничение разовой загрузки файлов. Также проверяется корректность файла, который отправляется на сервер. В случае, если файл корректный выдаётся сообщение о том, что он был загружен на сервер, а так же загружена информация о файле в базу данных. Если же файл не корректен, по причине того, что пользователь загружает не рабочий или не поддерживаемый формат файла или хочет специально загрузить неблагоприятный файл, web-ресурс выдаст сообщение о том, что происходит атака сайта с помощью файловой атаки. Данная проверка была написанная на php и представлена на рисунке 4.1

```
33 $uploaddir = './files/';
34 $login = $_SESSION['login'];
35 $password= $_SESSION['password'];
36 $id_user =(int) mysqli_query($link, "SELECT ID_User FROM User WHERE Login = '$login' AND Password = '$password'");
37 $uploadfile = $uploaddir . basename(iconv("utf-8", "cp1251", $_FILES['userfile']['name']));
38
39 echo '<pre>';
40 if (move_uploaded_file($_FILES['userfile']['tmp_name'], $uploadfile)) {
41     echo $f1="Файл корректен и был успешно загружен.\n";
42 } else {
43     echo $f2="Возможная атака с помощью файловой загрузки!\n";
44 }
45 if(isset($_POST['save'])){
46     $path="upload/";
47     $name = $_FILES['userfile']['name'];//Name of the File
48     $temp = $_FILES['userfile']['tmp_name'];
49     if(move_uploaded_file($temp, $path . $name)){
50         echo "success";
51     }else{
52         echo "failed";
53     }
54 }
```

Рисунок 4.1.1 Проверка файлов на корректность

## 4.2 Тестирование web-ресурса

После завершения разработки web-ресурса было проведено его тестирование. В результате данного тестирования была изучена необходимость разделов сайта заочной математической школы НИУ «БелГУ».

На рисунке 4.2 представлена главная страница web-ресурса. На ней можно увидеть весь основной дизайн сайта, который состоит из эмблемы заочной математической школы, расположенной в левой верхней части страницы, герба НИУ «БелГУ», расположенного справа сверху страницы. Между эмблемой школы и гербом НИУ «БелГУ» расположено название школы. Ниже эмблемы школы, её названия и герба НИУ «БелГУ» располагается навигационная панель, с помощью которой происходит перемещение между разделами сайта. Выше навигационной панели, слева расположены кнопки для входа и регистрации.

Ниже навигационной панели располагается основное содержание раздела. На главной странице располагается приветствие и описание Web-ресурса.



Рисунок 4.2.2 Главная страница web-ресурса

При наведении на раздел сайта он подчёркивается белой полосой. При отводе мыши от раздела сайта, белая полоса сворачивается. При нажатии на раздел, происходит переход на страницу, на котором так же располагается основные элементы, меняется содержимое раздела, расположенного ниже навигационной панели.

Следующий раздел – новости. В разделе новости располагается новостной блок выглядящий в виде списка (рисунок 4.2.3). Элемент списка новостного блока состоит из заголовка новости, изображения прикрепленного к новости и кратное содержание новости.

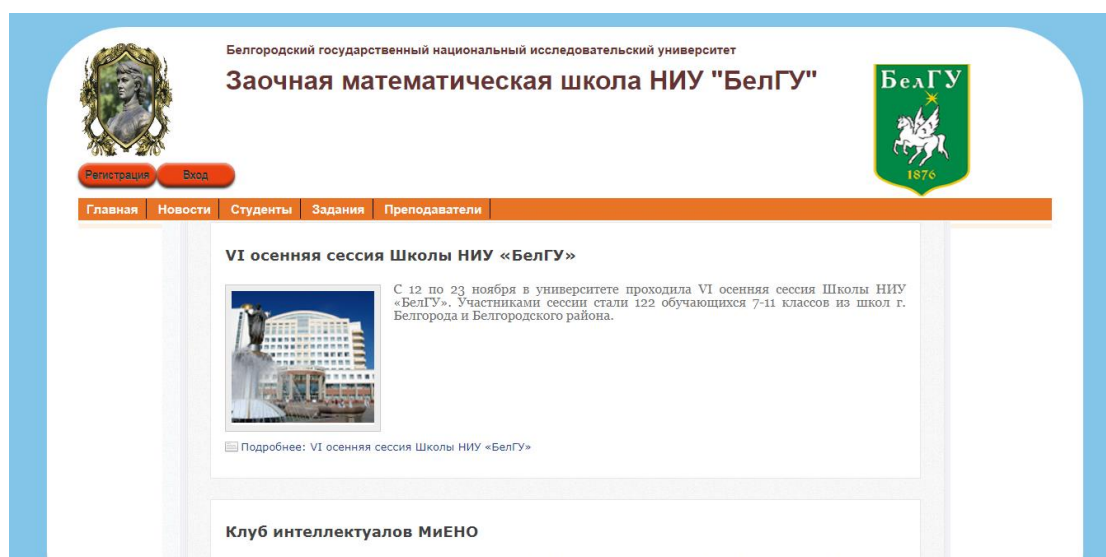


Рисунок 4.2.3 Раздел новостного блока

При нажатии на элемент списка новостного блока открывается страница, на которой расположен элемент списка, который содержит заголовок, изображение и текст новости (рисунок 4.4). Часть содержания новости располагается сверху изображения, вторая часть содержания сайта располагается ниже изображения.



Рисунок 4.2.4 Открытая новость из списка

Следующий раздел web-ресурса страница «Студенты». В данном разделе располагается список студентов с их данными в которые входят ФИО ученика, электронная почта, класс и оценки полученные за семестр (рисунок 4.2.5). Данный раздел предназначен для введения статистики по пользователям данного web-ресурса.

СТУДЕНТ	E-MAIL	КЛАСС	ОЦЕНКИ
АНТОНОВ АНТОН ИГОРЕВИЧ	ANTON.IGOREVICH658@MAIL.RU	8	4, 4, 5, 4, 4
БОЛГОВ АРТЁМ АНДРЕЕВИЧ	ARTEM.ANDREVICH56@GMAIL.COM	8	4, 3, 4, 4, 5
ВАСИЛЬЕВ ДЕНИС ФИЛИПОВИЧ	DENIS.FILIPOVICH@GMAIL.COM	9	4, 4, 4, 4, 4
ВАСИЛЬЕВА	NATALLA.ALEKSEEVNA57@MAIL.RU	9	3 3 4 4 5

Рисунок 4.2.5 Статистика по ученикам

Главным разделом веб-ресурса, является раздел «Задания». Именно данный раздел отвечает за обучающий процесс. На рисунке 4.6 представлен

вид раздела при входе преподавателя. В верхней части располагается раздел для загрузки задания. Если вход был выполнен в роли ученика данный раздел предназначен для загрузки решения. Ниже располагается список загруженных файлов. У преподавателя – это список присланных решённых заданий, которые он оценивает после проверки. У ученика – это список загруженных заданий, ожидающих оценки. После выставления оценки преподавателем напротив файла появляется полученная оценка. При попытке совершить файловую атаку, именно этот раздел будет оповещать о том что производится атака с помощью файлов, и что файлы не были загружены на сервер, а также не внесены в базу данных.

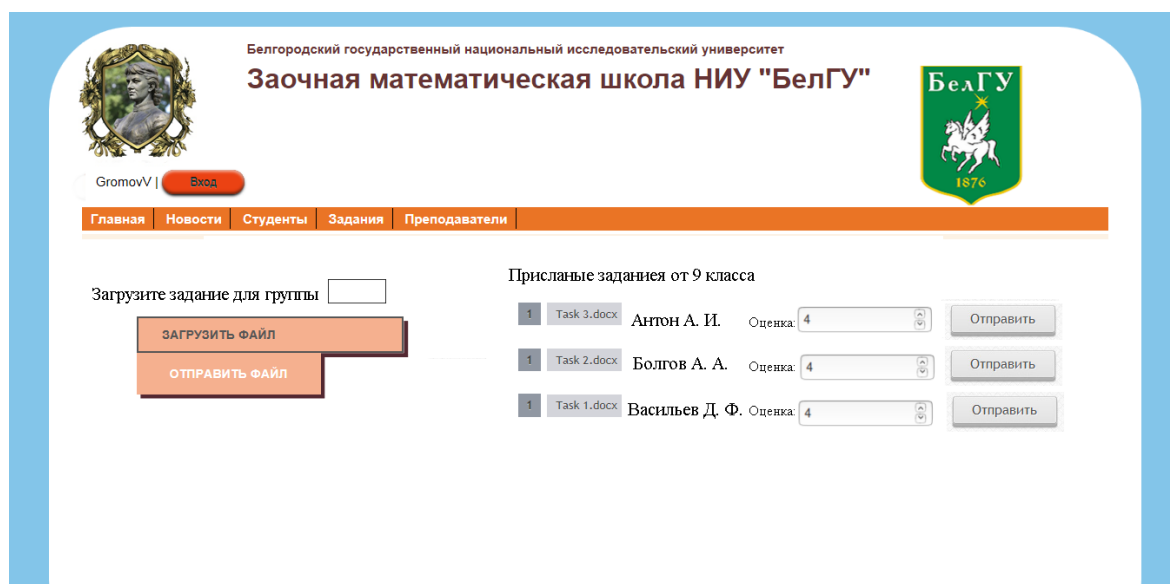


Рисунок 4.2.6 Раздел заданий

Последний раздел web-ресурса заочной математической школы – «Преподаватели». Данный раздел представляет собой список преподавателей которые ведут образовательный процесс в данной школе (рисунок 4.7). Каждый из них составляет свою обучающую систему по тем предметам которые, они ведут. Список состоит из следующих элементов:

- заголовки, в которых указаны фамилии, имена и отчества преподавателей.

- подзаголовки, в которых говорится, какой предмет ведёт данный преподаватель
- фотографии преподавателя, расположенная слева от заголовка.
- информация о преподавателе, расположенная справа от фотографии и снизу от подзаголовка.

Информация о преподавателе состоит из: образовательного учреждения, в котором он преподаёт, специальности, квалификации, должности и опыте работы.

Белгородский государственный национальный исследовательский университет  
**Заочная математическая школа НИУ "БелГУ"**

Регистрация Вход

Главная Новости Студенты Задания Преподаватели

**Аверин Геннадий Викторович**  
 Преподаватель математики

**Образовательное учреждение:** Николаевский ордена Трудового Красного Знамени кораблестроительный институт им. адм. С.О. Макарова, 1980  
**Специальность:** холодильные и компрессорные машины и установки  
**Квалификация:** инженер-механик  
**Должность:** доктор технических наук, профессор  
**Должность:** заведующий кафедрой общей математики, профессор  
**Опыт преподавания:** 19 лет

**Некрасова Ирина Викторовна**  
 Преподаватель математики

Рисунок 4.2.7 Список преподавателей

## ЗАКЛЮЧЕНИЕ

В эпоху развития информационных технологий образовательному учреждению требуется образовательный web-ресурс. Web-сайт помогает решить целый ряд разнообразных задач, служит визитной карточкой образовательного учреждения, выполняет образовательную функцию, привлекает внимание целевой аудитории.

Многие образовательные учреждения имеют свои web-ресурсы. Свои заочные школы имеют университеты, многие колледжи и другие учреждения образования. Однако большая часть таких web-ресурсов имеют лишь информативную часть. Для реализации образовательного процесса заочные школы имеют свой сайт с описанием деятельности, инструментами возможности записи на обучение в данную школу, и встроенную в сайт уникальную образовательную программу. Web-ресурс должен решать проблему навигации по сайту. Главная задача при создании web-ресурса связана с структурой и информативной составляющей.

В результате выполнения выпускной квалификационной работы был создан web-ресурс заочной математической школы НИУ «БелГУ». Данный ресурс, в первую очередь, имеет простую и удобную структуру сайта состоящую из одной навигационной панели. Ресурс предоставляет информацию о школе для привлечения новых пользователей. Сайт заочной математической школы имеет новостной блок, который может вести преподаватель, а также возможность вести статистики по ученикам зарегистрированным на web-ресурсе. Сайт также имеет систему удалённого обучения. В последнем разделе web-ресурса можно ознакомиться со списком преподавателей, которые преподают в данной заочной математической школе. При создании сайта была создана htmk разметка, спроектирован дизайн сайта школы, создана база данных MySQL.

В результате выше сказанного была поставлена цель выпускной квалификационной работы – создание web-ресурса заочной математической школы НИУ «БелГУ» с использованием php и MYSQL под управлением phpmyadmin.

В процессе выполнения выпускной квалификационной работы были получены следующие результаты:

1. Проанализирована структура и функционал аналогичных web-ресурсов школьного заочного образования;
2. Сформированы требования к структуре сайта в соответствующих методических документах;
3. Выполнены проектирования web-ресурса с использованием средств php и MySQL;
4. Проведены тестирование и апробация разработанного web-ресурса.



## СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Мишель Е. Д. Изучаем PHP и MySQL / Е. Д. Мишель, А. Ф. Джон. – СПб.: Символ-Плюс, 2008. 442с.
2. Прохоренок Н. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера / Н. Прохоренок. – СПб.: БХВ-Петербург, 2010. 900с.
3. Колисниченко Д.Н. PHP 5/6 и MySQL 6. Разработка web приложений / Д.Н. Колисниченко. – СПб.:БХВ-Петербург, 2010. 560с.
4. Советов Б. Я. Архитектура информационных систем / Б. Я. Советов, А. И. Водяхо, В. А. Дубенецкий, В. В. Цехановский. – М.: Академия, 2012. 228с.
5. Кузнецов М. Объектно-ориентированное программирование на PHP /М. Кузнецов, И. Симдянов. – СПб.: БХВ-Петербург, 2007. 608с.
6. Ловэйн П. PHP 5. Объектно-ориентированное программирование. / П.Ловэйн. – М.:ИТ Пресс, 2007. 224с.
7. Зандстра М. PHP. Объекты, шаблоны и методики программирования. / М. Зандстра– М: Вильямс, 2011. 560с.
8. ВеллингЛ. Разработка веб-приложений с помощью PHP и MySQL / Л. Веллинг, Л. Томсон. – М : Вильямс, 2010. 848с.
9. <http://dev.mysql.com> – официальный сайт свободной система управления базами данныхMySQL.
10. <http://httpd.apache.org> – официальный сайт свободного сервера Apache.
11. <http://php.net/> – официальный сайт скриптового языка.
12. <http://netbeans.org/> – свободная интегрированная среда разработки приложений.
13. <http://www.web-learn.ru> – сайт организации дистанционного обучения.

14. <http://www.tinymce.com/> – платформонезависимый Javascript HTML WYSIWYG редактор на основе Web.
15. Китинг, Джоди Flash MX. Искусство создания web-сайтов; ТИД ДС - Москва, 2012. - 848 с.
16. Фридман В. А., Александров А. В., Сергеев Г. Г., Костин С. П. Строительство Web-сайтов (+ CD-ROM); Триумф - Москва, 2011. - 288 с.
17. Хуторской А. В., Орешко А. П. Технология создания сайтов. 10-11 классы; Дрофа - Москва, 2011. - 256 с.  
Ларри Ульман. Ульман Л. Основы программирования на PHP:/Ларри Ульман. Пер. с англ. -М.: ДМК Пресс, 2001. -288 с.: ил. (Самоучитель).
18. Александр Мазуркевич. МВ PHP: настольная книга программиста /Александр Мазуркевич, Дмитрий Еловой. — Мн.: Новое знание, 2003. — 480 с.: ил.
19. Материалы официального сайта языка программирования PHP <http://www.php.net/> Петербург, 2013. - 416 с.
20. Квинт, Игорь Создаем сайты с помощью HTML, XHTML и CSS / Игорь Квинт. - М.: Питер, 2014. - 448 с.
21. Квинт, Игорь Создаем сайты с помощью HTML, XHTML и CSS на 100% / Игорь Квинт. - М.: Питер, 2012. - 448 с.
22. Лазаро, Исси Коэн Полный справочник по HTML, CSS и JavaScript / Лазаро Исси Коэн, Джозеф Исси Коэн. - М.: ЭКОМ Паблишерз, 2014. - 938 с.
23. 14. Мержевич, Влад HTML и CSS на примерах / Влад Мержевич. - М.: "БХВ-Петербург", 2012. - 448 с.
24. 15. Никсон, Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5 / Р. Никсон. - Москва: Машиностроение, 2016. - 688 с.
25. Никсон, Робин Создаем динамические веб-сайты с помощью PHP, MySQL и JavaScript / Робин Никсон. - М.: Питер, 2013. - 496 с.

MySQL, JavaScript и CSS / Робин Никсон. - М.: "Издательство "Питер", 2013. - 560 с.

26. Пауэрс, Дэвид Adobe Dreamweaver, CSS, Ajax и PHP / Дэвид Пауэрс. - М.: БХВ-Петербург, 2012. - 829 с.

27. Прохоренок, Н. А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера / Н.А. Прохоренок, В.А. Дронов. - Москва: СПб. [и др.] : Питер, 2015. - 768 с.

Выпускная квалификационная работа выполнена мной самостоятельно.  
Все использованные в работе материалы и концепции из опубликованной научной литературы и других источников имеют ссылки на них.

« \_\_\_ » \_\_\_\_\_ Г.

\_\_\_\_\_

\_\_\_\_\_  
(подпись)

\_\_\_\_\_

(Ф.И.О.)

```

<html>
<header>
<head>
<link rel='stylesheet' href='HomeStyle.css' type='text/css'>
  <title>Название сайта</title>
</head>

Белгородский государственный
национальный исследовательский университет<div
class='title1'></div>
<div class='title2'>Заочная математическая школа НИУ
"БелГУ"</p></div>
<style>
body, html {
  margin-top: 10px;
  padding-left: 40px;
  padding-right: 40px;
  font-family: Geneva, Arial, Helvetica, sans-serif;
}

li {display: inline-block;}
.menu-main {
  list-style: none;
  margin: 1px 0 5px;
  padding: 1px 1px 2px;
  text-align: left;
  background: white;
  //padding-top: 4px;
  background-color: transparent;
}
.menu-main {
  background-color: #ea7425; /* Цвет фона */
  color: white; /* Цвет текста */
}
.menu-main li {
  display: inline-block;
  border-right: 1px solid black;
  height: 25px;
}
.menu-main li:after {
  color: #606060;
  display: inline-block;
  vertical-align: bottom;
}
.menu-main li:last-child:after {content: none;}
.menu-main a {
  text-decoration: none;
  font-family: Arial;
  position: relative;
  padding-bottom: 0;
  margin: -20px 10px -20px 10px;
  font-size: 16px;
  display: inline-block;
  font-weight: bold;
  margin-top: 4px;
}
.menu-main a, .menu-main a:visited {color: white;}
.menu-main a:before,
.menu-main a:after {
  content: "";
  position: absolute;
  height: 1px;
  top: auto;
  right: 50%;
  bottom: 1px;
  left: 50%;
  background: white;
  //transition: .8s;
}
.menu-main a:hover:before, .menu-main .current:before {left: 0;}
.menu-main a:hover:after, .menu-main .current:after {right: 0;}
@media (max-width: 100px) {
  .menu-main {padding-top: 0;}
  .menu-main li {display: block;}
  .menu-main li:after {content: none;}
  .menu-main a {
    padding: 10px 0 20px;
    margin: 0 10px;
  }
}
div {content: none;}
.title1 {

```

```

padding-left: 200px;
font-family: Arial;
font-size: 16px;
font-weight: bold;
vertical-align: bottom;
//padding-bottom: 10px;
color: #663333; /* Цвет текста */
}
.title2 {
padding-left: 200px;
font-family: Arial;
font-size: 32px;
font-weight: bold;
vertical-align: bottom;
padding-bottom: 50px;
color: #663333; /* Цвет текста */
}

</style>
<style>
.knopka01 {
display: inline-block;
width: 100px;
height: 30px;
line-height: 30px;
font-size: 14px;
text-align: center;
text-decoration: none;
text-shadow: 0 1px rgba(255,255,255,.2), 0 -1px rgba(0,0,0,.8);
outline: none;
border: none;
border-radius: 100px;
background: rgb(235, 66, 19); radial-gradient(150% 100% at 50%
5px, rgba(255,255,255,.2), rgba(0,0,0,0));
box-shadow: inset rgba(0,0,0,.6) 0 -2px 5px, inset
rgba(252,255,255,.7) 0 2px 5px, rgba(0,0,0,.8) 0 2px 5px -1px;
color: rgb(0,79,86);
user-select: none;
}
.knopka01:hover {
color: rgb(35, 212, 0);
cursor: pointer;
}
.knopka01:active {
padding-bottom: 1px;
box-shadow:
inset rgba(0,0,0,1) 0 1px 3px,
inset rgba(0,0,0,.6) 0 -2px 5px,
inset rgba(252,255,255,.7) 0 2px 5px,
0 1px rgba(255,255,255,.08);

```

```

color: rgb(128,207,214);
}
</style>
<br>
<?php
session_start();
echo "<link rel='stylesheet' href='HomeStyle.css' type='text/css'>";
$login = $_SESSION['login'];
$password = $_SESSION['password'];
if(empty($login) and empty($password))
{
echo "<form enctype='multipart/form-data' action='Reg.php'
method='POST' class='knopka01'>";
echo "<input type='submit' class='knopka01' value='Регистрация'
action='Reg.php'></form>";
echo "<form enctype='multipart/form-data' action='WReg.php'
method='POST' class='knopka01'>";
echo "<input type='submit' class='knopka01' value='Вход'
></form><br>";
}
else{
echo $login." | <form enctype='multipart/form-data'
action='exit.php' method='POST' class='knopka01'>";
echo "<input type='submit' class='knopka01' value='Вход'
></form>";
}
?>
</header>
<div>
<ul class='menu-main' >
<li><a href='Main.php'>Главная</a></li>
<li><a href='News.php'>Новости</a></li>
<li><a href='Stud.php'>Студенты</a></li>
<li><a href='Task.php'>Задания</a></li>
</ul>
</div>
</body>
</html>
<?php
require_once 'connection.php'; // подключаем скрипт

$link = mysqli_connect($host, $user, $password, $database)
or die("Ошибка " . mysqli_error($link));

```

```

$query = "SELECT Text_News FROM News";

mysqli_close($link);
$link = mysqli_connect($host, $user, $password, $database)
    or die("Ошибка " . mysqli_error($link));
?>
<html>
<header>
<head>
<link rel='stylesheet' href='HomeStyle.css' type='text/css'>
<title>Название сайта</title>

</head>

<div class='title1'>Белгородский
государственный национальный исследовательский
университет</div>
<div class='title2'>Институт инженерных и цифровых
технологий</p></div>
<style>
.leftimg {
float:left; /* Выравнивание по левому краю */
margin: 7px 7px 7px 0; /* Отступы вокруг картинки */
}
.rightimg {
float:right; /* Выравнивание по правому краю */
margin: 7px 0 7px 7px; /* Отступы вокруг картинки */
}
.three {
background: #FCF2E5;
padding: 20px 20px;
text-align: center;
}
.three h1 {
font-family: 'Merriweather', serif;
position: relative;
color: #FCF2E5;
background: #90806A;
font-size: 2.5em;
font-weight: normal;
padding: 10px 40px;
display: inline-block;
margin: 0;
line-height: 1;
}

}

@media (max-width: 500px) {
.three h1 {font-size: 1.8em;}
}
@media (max-width: 400px) {
.three h1 {
font-size: 1.5em;
padding: 10px 30px;
}
}
</style>
<style>
header {

height:500;
display: block; /* Для старых браузеров */
background: url(BackGroundImage/logo-belgu.png) no-repeat
center bottom;

}
html{
display: block; /* Для старых браузеров */
background:#83C5E9;
}
body {
display: block; /* Для старых браузеров */
background:white;
border-radius: 100px;
}
</style>
<style>
.knopka01 {
display: inline-block;
width: 100px;
height: 30px;
line-height: 30px;
font-size: 14px;
text-align: center;
text-decoration: none;
text-shadow: 0 1px rgba(255,255,255,.2), 0 -1px rgba(0,0,0,.8);
outline: none;
border: none;
border-radius: 100px;
background: rgb(235, 66, 19); radial-gradient(150% 100% at 50%
5px, rgba(255,255,255,.2), rgba(0,0,0,0));
box-shadow: inset rgba(0,0,0,.6) 0 -2px 5px, inset
rgba(252,255,255,.7) 0 2px 5px, rgba(0,0,0,.8) 0 2px 5px -1px;
color: rgb(0,79,86);
user-select: none;
}

```

```

}
.knopka01:hover {
color: rgb(35, 212, 0);
cursor: pointer;
}
.knopka01:active {
padding-bottom: 1px;
box-shadow:
inset rgba(0,0,0,1) 0 1px 3px,
inset rgba(0,0,0,.6) 0 -2px 5px,
inset rgba(252,255,255,.7) 0 2px 5px,
0 1px rgba(255,255,255,.08);
color: rgb(128,207,214);
}
</style>

<br>
<?php
session_start();
echo "<link rel='stylesheet' href='HomeStyle.css' type='text/css'>";
$login = $_SESSION['login'];
$password = $_SESSION['password'];
if(empty($login) and empty($password)){
echo "<form enctype='multipart/form-data' action='Reg.php'
method='POST' class='knopka01'>";
echo "<input type='submit' class='knopka01' value='Регистрация'
action='Reg.php'></form>";
echo "<form enctype='multipart/form-data' action='WReg.php'
method='POST' class='knopka01'>";
echo "<input type='submit' class='knopka01' value='Вход'
></form><br>";
}
else{
echo $login. " | <form enctype='multipart/form-data'
action='exit.php' method='POST' class='knopka01'>";
echo "<input type='submit' class='knopka01' value='Вход'
></form>";
}
?>
</header>
<div>
<ul class='menu-main' >
<li><a href='Main.php'>Главная</a></li>
<li><a href='News.php'>Новости</a></li>
<li><a href='Stud.php'>Студенты</a></li>
<li><a href='Task.php'>Задания</a></li>
<li><a href='Task.php'>Преподаватели</a></li>

```

```

</ul>
</div>
<div class="three"><h1> Математическую школу Софьи
Ковалевской создадут в Белгороде</h1></div>

<?php
$result = mysqli_query($link,$query) or die("Ошибка " .
mysqli_error($link));
if($result)
{
$rows = mysqli_num_rows($result); // количество полученных
строк

for ($i = 0 ; $i < $rows ; ++$i)
{
$row = mysqli_fetch_row($result);
for ($j = 0 ; $j < 3 ; ++$j) echo "<p class
='layer2' align='justify' width='100px'>$row[$j]<br></p>";
}
echo "</table>";
// очищаем результат
mysqli_free_result($result);
}
?>
</body>
</html>
<?php
$host = 'localhost'; // адрес сервера
$dbase = 'MathSchool'; // имя базы данных
$user = 'root'; // имя пользователя
$password = ""; // пароль
$link = mysqli_connect($host,$user,$password,$dbase);

if (!$link) {
echo 'Не могу соединиться с БД. Код ошибки: ' .
mysqli_connect_errno() . ', ошибка: ' . mysqli_connect_error();
exit;
}
mysql_select_db("MathSchool");

$link = mysqli_connect($host,$user,$password,$dbase)
or die("Ошибка " . mysqli_error($link));

mysqli_close($link);
$link = mysqli_connect($host,$user,$password,$dbase)
or die("Ошибка " . mysqli_error($link));

```



```

?>
<html>
<header>
<head>
<link rel='stylesheet' href='HomeStyle.css' type='text/css'>
  <title>Название сайта</title>
</head>

<div class='title1'>Белгородский
государственный национальный исследовательский
университет</div>
<div class='title2'> Институт инженерных и цифровых
технологий</p></div>
<style>
.example-2 .btn-tertiary{color:#555;padding:0;line-
height:40px;width:300px;display:block;border:2px solid #555 }
.example-2 .btn-tertiary:hover,.example-2 .btn-
tertiary:focus{color:#888;border-color:#888}
.example-2 .input-
file{ width:.1px;height:.1px;opacity:0;overflow:hidden;position:abs-
olute;z-index:-1}
.example-2 .input-file + .js-labelFile { overflow:hidden;text-
overflow:ellipsis;white-space:nowrap;padding:0
10px;cursor:pointer}
.example-2 .input-file + .js-labelFile .icon:before{content:"\f093"}
.example-2 .input-file + .js-labelFile.has-file
.icon:before{content:"\f00c";color:#5AAC7B}
.btn {
  position: relative;
  overflow: hidden;
  z-index: 1;
  padding: 10px 40px;
  margin: 0 20px;
  font-family: 'Montserrat Alternates', sans-serif;
  font-weight: 600;
  line-height: 30px;
  color: white;
  font-size: 15px;
  text-transform: uppercase;
  background: #f5b08f;
  border-width: 0;
  box-shadow: 5px 5px 0 #532831;
  outline: none;
  cursor: pointer;
  transition: 1.5s;
}

```

```

.btn:before,
.btn:after {
  content: "";
  position: absolute;
  height: 200px;
  left: -50%;
  margin-top: -100px;
  top: 50%;
  width: 200px;
  border-radius: 50%;
  opacity: 0.3;
  z-index: -1;
  transform: scale(0);
}
.btn:before {
  background: #ffeede;
  transition: .8s ease-out;
}
.btn:after {
  transition: .4s ease-in .3s;
}
.btn:hover {
  color: #532831;
}
.btn:hover:before,
.btn:hover:after{
  opacity: 1;
  transform: scale(4);
}
</style>
<style>
.rectangle {
  counter-reset: li;
  list-style: none;
  font: 14px "Trebuchet MS", "Lucida Sans";
  padding: 0;
  text-shadow: 0 1px 0 rgba(255,255,255,.5);
}
.rectangle a {
  position: relative;
  display: block;
  padding: .4em .4em .4em .8em;
  margin: .5em 0 .5em 2.5em;
  background: #D3D4DA;
  color: #444;
  text-decoration: none;
  transition: all .3s ease-out;
}
.rectangle a:hover {background: #DCDDE1;}
.rectangle a:before {

```

```

content: counter(li);
counter-increment: li;
position: absolute;
left: -2.5em;
top: 50%;
margin-top: -1em;
background: #9097A2;
height: 2em;
width: 2em;
line-height: 2em;
text-align: center;
font-weight: bold;
}
.rectangle a:after {
position: absolute;
content: "";
border: .5em solid transparent;
left: -1em;
top: 50%;
margin-top: -.5em;
transition: all .3s ease-out;
}
.rectangle a:hover:after {
left: -.5em;
border-left-color: #9097A2;
}
</style>
<style>
header {

    hight:500;
    display: block; /* Для старых браузеров */
    background: url(BackGroundImage/logo-belgu.png) no-repeat
center bottom;

}
html{
    display: block; /* Для старых браузеров */
    background:#83C5E9;
}
body{
    display: block; /* Для старых браузеров */
    background:white;
    border-radius: 100px;
}
</style>
<style>
.knopka01 {
    display: inline-block;
    width: 100px;

    height: 30px;
    line-height: 30px;
    font-size: 14px;
    text-align: center;
    text-decoration: none;
    text-shadow: 0 1px rgba(255,255,255,.2), 0 -1px rgba(0,0,0,.8);
    outline: none;
    border: none;
    border-radius: 100px;
    background: rgb(235, 66, 19);radial-gradient(150% 100% at 50%
5px, rgba(255,255,255,.2), rgba(0,0,0,0));
    box-shadow: inset rgba(0,0,0,.6) 0 -2px 5px, inset
rgba(252,255,255,.7) 0 2px 5px, rgba(0,0,0,.8) 0 2px 5px -1px;
    color: rgb(0,79,86);
    user-select: none;
}
.knopka01:hover {
    color: rgb(35, 212, 0);
    cursor: pointer;
}
.knopka01:active {
    padding-bottom: 1px;
    box-shadow:
inset rgba(0,0,0,1) 0 1px 3px,
inset rgba(0,0,0,.6) 0 -2px 5px,
inset rgba(252,255,255,.7) 0 2px 5px,
0 1px rgba(255,255,255,.08);
    color: rgb(128,207,214);
}
}
</style>
<br>
<?php
session_start();
echo "<link rel='stylesheet' href='HomeStyle.css' type='text/css'>";
$login = $_SESSION['login'];
$password = $_SESSION['password'];
if(empty($login) and empty($password)){
echo "<form enctype='multipart/form-data' action='Reg.php'
method='POST' class='knopka01'>";
echo "<input type='submit' class='knopka01' value='Регистрация'
action='Reg.php'></fom'>";
echo "<form enctype='multipart/form-data' action='WReg.php'
method='POST' class='knopka01'>";
echo "<input type='submit' class='knopka01' value='Вход'
></form><br>";
}
}

```

```

else{
echo $login." | <form enctype='multipart/form-data'
action='exit.php' method='POST' class='knopka01'>";
echo "<input type='submit' class='knopka01' value='Вход'
></form>";
}
?>
</header>
<ul class='menu-main'>
<li><a href='Main.php'>Главная</a></li>
<li><a href='News.php'>Новости</a></li>
<li><a href='Stud.php'>Студенты</a></li>
<li><a href='Task.php'>Задания</a></li>
<li><a href='Task.php'>Преподаватели</a></li>
</ul>
</div>

<?php
$login = $_SESSION['login'];
$password = $_SESSION['password'];

if(empty($login) and empty($password)){ else{

echo "<form enctype='multipart/form-data' action='STask.php'
method='POST' '>";
echo "";
echo "<div class='example-2'><input name='userfile' type='file'
value='Открыть' class='input-file' /></div><br>";
echo "<div class='example-2'>";
echo "<div class='form-group'>";
echo " <input type='file' name='userfile' id='file' class='input-
file'>";
echo " <label for='file' class='btn btn-tertiary js-labelFile'>";
echo " <i class='icon fa fa-check'></i>";
echo " <span class='js-fileName'>Загрузить файл</span>";
echo " </label>";
echo "</div>";
echo "";
echo "<input class='btn' type='submit' value='Отправить файл'
/><br><br>";

header("Content-Type: text/html; charset=utf-8");

$dir = 'files/';
$skip = array('.', '..');
$files = scandir($dir, 1 );

foreach ($files as $file){
if(!in_array($file, $skip))

echo "<ol class='rectangle'>";
echo "<li><a href='$dir$file' charset='utf-8'>$file</a></li>
<br>";
echo "</ol>";
}
}
?>
</form>

</body>
</html>
<?php
error_reporting(E_ALL);
session_start();
if (isset($_POST['login'])){
$login = $_POST['login'];
}
if (isset($_POST['password'])){
$password = $_POST['password'];
}
error_reporting(-1);
header('Content-Type: text/html; charset=utf-8');
$host = 'localhost'; // адрес сервера
$dbname = 'MathSchool'; // имя базы данных
$user = 'root'; // имя пользователя
$password = ''; // пароль
$link = mysqli_connect($host, $user, $password, $dbname);

if (!$link) {
echo 'Не могу соединиться с БД. Код ошибки: '.
mysqli_connect_errno() . ', ошибка: ' . mysqli_connect_error();
exit;
}
mysql_select_db("MathSchool");

$link = mysqli_connect($host, $user, $password, $dbname)
or die("Ошибка " . mysqli_error($link));

mysqli_close($link);
$link = mysqli_connect($host, $user, $password, $dbname)
or die("Ошибка " . mysqli_error($link));

$uploaddir = './files/';
$login = $_SESSION['login'];
$password = $_SESSION['password'];
$id_user = (int) mysqli_query($link, "SELECT ID_User FROM
User WHERE Login = '$login' AND Password = '$password'");

```

```

$uploadfile = $uploaddir . basename(iconv("utf-8", "cp1251",
$_FILES['userfile']['name']));

echo '<pre>';
if (move_uploaded_file($_FILES['userfile']['tmp_name'],
$uploadfile)) {
    echo $f1="Файл корректен и был успешно загружен.\n";
} else {
    echo $f2="Возможная атака с помощью файловой
загрузки!\n";
}
if(isset($_POST['save'])){
    $path="upload/";
    $name = $_FILES['userfile']['name'];//Name of the File
    $temp=$_FILES['userfile']['tmp_name'];
    if(move_uploaded_file($temp,$path.$name)){
        echo "success";
    }else{
        echo "failed";
    }
}
$NameFile = $_FILES['userfile']['name'];
if (isset($NameFile)) {
    //Вставляем данные, подставляя их в запрос
    $sql = mysqli_query($link, "INSERT INTO Files (NameFile,
FilePath, ID_User) VALUES
('$NameFile',/files/$NameFile','$id_user')");
    //Если вставка прошла успешно
    if ($sql) {
        echo '<p>Данные успешно добавлены в таблицу.</p>';
    } else {
        echo '<p>Произошла ошибка: ' . mysqli_error($link) . '</p>';
    }
}
require_once("Task.php");
//header('Location: Task.php');
// session_destroy();
?>
body, html {
    margin-top: 10px;
    padding-left: 40px;
    padding-right: 40px;
    font-family: Geneva, Arial, Helvetica;
}

li {display: inline-block;}
.menu-main {
    list-style: none;
    margin: 1px 0 5px;
    padding: 1px 1px 2px;

```

```

text-align: left;
background: white;
padding-top: 4px;
background-color:transparent;
}
.menu-main {
    background-color:#ea7425;/* Цвет фона*/
    color: white;/* Цвет текста */
}
.menu-main li {
    display: inline-block;
    border-right: 1px solid black;
    height: 25px;}
.menu-main li:after {
    color: #606060;
    display: inline-block;
    vertical-align:bottom;
}
.menu-main li:last-child:after {content: none;}
.menu-main a {
    text-decoration: none;
    font-family: Arial;
    position: relative;
    padding-bottom: 0;
    margin: -20px 10px -20px 10px;
    font-size: 16px;
    display: inline-block;
    font-weight: bold;
    margin-top: 4px;
}
.menu-main a, .menu-main a:visited {color: white;}
.menu-main a:before,
.menu-main a:after {
    content: "";
    position: absolute;
    height: 1px;
    top: auto;
    right: 50%;
    bottom: 1px;
    left: 50%;
    background: white;
    transition: .8s;
}
.menu-main a:hover:before, .menu-main .current:before {left: 0;}
.menu-main a:hover:after, .menu-main .current:after {right: 0;}
@media (max-width: 100px) {
.menu-main {padding-top: 0;}
.menu-main li {display: block;}

```

```

.menu-main li:after {content: none;}
.menu-main a {
padding: 10px 0 20px;
margin: 0 10px;
}
}

div {content: none;}
.title1 {
padding-left: 200px;
font-family: Arial;
font-size: 16px;
font-weight: bold;
vertical-align: bottom;
padding-bottom: 10px;
columns: #666666; /* Цвет текста */
}
.title2 {
padding-left: 200px;
font-family: Arial;
font-size: 32px;
font-weight: bold;
vertical-align: bottom;
padding-bottom: 50px;
color: #663333; /* Цвет текста */
}

<?php
include_once("db.php");

if (isset($_POST['submit'])) {
    if (empty($_POST['login'])) {
        echo '<br><font color="red"> Введите логин!</font>';
    }
    elseif (!preg_match("/^\w{3,}$/", $_POST['login'])) {
        echo '<br><font color="red">В
поле "Логин" введены недопустимые символы! Только буквы,
цифры и подчеркивание!</font>';
    }
    elseif (empty($_POST['password'])) {
        echo '<br><font color="red">Введите пароль!</font>';
    }
    elseif (!preg_match("/^\A(\w){6,20}\Z/", $_POST['password'])) {
        echo '<br><font color="red">Пароль слишком короткий!
Пароль должен быть не менее 6 символов! </font>';
    }
}

```

```

elseif (empty($_POST['password2'])) {
    echo '<br><font color="red">Введите
подтверждение пароля!</font>';
}
elseif ($_POST['password'] != $_POST['password2']) {
    echo '<br><font color="red">Введенные пароли не совпадают!</font>';
}
elseif (empty($_POST['email'])) {
    echo '<br><font color="red">Введите E-mail! </font>';
}
elseif (!preg_match("/^[a-zA-Z0-9_\.]+\@[a-zA-Z0-9\-\+\.]+[a-
zA-Z]{2,6}$/", $_POST['email'])) {
    echo '<br><font color="red">E-mail имеет
недопустимый формат! Например, name@gmail.com! </font>';
}

else {
    $login = $_POST['login'];
    $password = $_POST['password'];
    $mdPassword = md5($password);
    $password2 = $_POST['password2'];
    $email = $_POST['email'];
    $name = $_POST['name'];
    $lastname = $_POST['lastname'];

    $query = ("SELECT ID_Student FROM Student WHERE
Login='$login'");
    $sql = mysql_query($query) or die(mysql_error());

    if (mysql_num_rows($sql) > 0) {
        echo '<font color="red">Пользователь с таким логином
зарегистрирован!</font>';
    }
    else {
        $query2 = ("SELECT ID_Student FROM Student WHERE
Email='$email'");
        $sql = mysql_query($query2) or die(mysql_error());
        if (mysql_num_rows($sql) > 0) {
            echo '<font color="red">Пользователь с таким e-mail уже
зарегистрирован!</font>';
        }
    }
}

```

```

else{
$query = "INSERT INTO Student (Login, Password, Email,
Name, Surname )
VALUES ('$login', '$mdPassword', '$email', '$name', '$lastname)';
$result = mysql_query($query) or die(mysql_error());
echo '<font color="green">Вы успешно
зарегистрировались!</font><br><a href="index.php">На
главную</a>';
}
}
}
}
?>
<?php
require_once 'connection.php'; // подключаем скрипт

$link = mysqli_connect($host, $user, $password, $database)
or die("Ошибка " . mysqli_error($link));

$query = "SELECT Name, Surname, Email FROM Student";

mysqli_close($link);
$link = mysqli_connect($host, $user, $password, $database)
or die("Ошибка " . mysqli_error($link));
?>
<html>
<header>
<head>
<link rel='stylesheet' href='HomeStyle.css' type='text/css'>
<title>Название сайта</title>
</head>

<div class='title1'>Белгородский
государственный национальный исследовательский
университет</div>
<div class='title2'> Институт инженерных и цифровых
технологий</p></div>

<style>
header {
height:500;
display: block; /* Для старых браузеров */

```

```

background: url(BackGroundImage/logo-belgu.png) no-repeat
center bottom;
}
html{
display: block; /* Для старых браузеров */
background:#83C5E9;
}
body {
display: block; /* Для старых браузеров */
background:white;
}
</style>
<style>
.knopka01 {
display: inline-block;
width: 100px;
height: 30px;
line-height: 30px;
font-size: 14px;
text-align: center;
text-decoration: none;
text-shadow: 0 1px rgba(255,255,255,.2), 0 -1px rgba(0,0,0,.8);
outline: none;
border: none;
border-radius: 100px;
background: rgb(235, 66, 19); radial-gradient(150% 100% at 50%
5px, rgba(255,255,255,.2), rgba(0,0,0,0));
box-shadow: inset rgba(0,0,0,.6) 0 -2px 5px, inset
rgba(252,255,255,.7) 0 2px 5px, rgba(0,0,0,.8) 0 2px 5px -1px;
color: rgb(0,79,86);
user-select: none;
}
.knopka01:hover {
color: rgb(35, 212, 0);
cursor: pointer;
}
.knopka01:active {
padding-bottom: 1px;
box-shadow:
inset rgba(0,0,0,1) 0 1px 3px,
inset rgba(0,0,0,.6) 0 -2px 5px,
inset rgba(252,255,255,.7) 0 2px 5px,
0 1px rgba(255,255,255,.08);
color: rgb(128,207,214);
}
}
</style>

```

```

<br>
<?php
session_start();
echo "<link rel='stylesheet' href='HomeStyle.css' type='text/css'>";
$login = $_SESSION['login'];
$password = $_SESSION['password'];
if(empty($login) and empty($password)){
echo "<form enctype='multipart/form-data' action='Reg.php'
method='POST' class='кнопка01'>";
echo "<input type='submit' class='кнопка01' value='Регистрация'
action='Reg.php'></form>";
echo "<form enctype='multipart/form-data' action='WReg.php'
method='POST' class='кнопка01'>";
echo "<input type='submit' class='кнопка01' value='Вход'
></form><br>";
}
else{
echo $login." | <form enctype='multipart/form-data'
action='exit.php' method='POST' class='кнопка01'>";
echo "<input type='submit' class='кнопка01' value='Вход'
></form>";
}
?>
</header>
<div>
<ul class='menu-main'>
<li><a href='Main.php'>Главная</a></li>
</ul>
</div>
</header>
<div>
<table border='1'>
<tr>
<td>Логин</td>
<td><input type='text' name='login'></td>
</tr>
<tr>
<td>Пароль</td>
<td><input type='text' name='password'></td>
</tr>
<tr>
<td colspan='2'><input type='submit' value='Войти' name='
Войти'></td>
</tr>
</table>
<a href='Reg.php'>Регистрация</a>
</div>
</body>
</html>

```

```

<li><a href='News.php'>Новости</a></li>
<li><a href='Stud.php'>Студенты</a></li>
<li><a href='Task.php'>Задания</a></li>
</ul>
</div>
<?php
$result = mysqli_query($link, $query) or die("Ошибка " .
mysqli_error($link));
if($result)
{
$rows = mysqli_num_rows($result); // количество полученных
строк

for ($i = 0 ; $i < $rows ; ++$i)
{
$row = mysqli_fetch_row($result);
echo "<tr>";
for ($j = 0 ; $j < 3 ; ++$j) echo "<td> $row[$j] </td>";
echo "<br></tr>";
}
echo "</table>";
mysqli_free_result($result);
}
?>
</body>
</html>

<table border='1'>
<tr>
<td>Логин<font color='red'>*</font></td>
<td><input type='text' size='20' name='login'></td>
</tr>
<tr>
<td>Пароль<font color='red'>*</font></td>
<td><input type='password' size='20' maxlength='20'
name='password'></td>
</tr>
</table>

```

```

<td>Подтверждения пароля<font color='red'>*</font>:</td>
<td><input type='password' size='20' maxlength='20'
name='password2'></td>
</tr>
<tr>
<td>E-mail<font color='red'>*</font>:</td>
<td><input type='text' size='20' name='email'></td>
</tr>
<tr>
<td>Имя:</td>
<td><input type='text' size='20' name='name'></td>
</tr>
<tr>
<td>Фамилия:</td>
<td><input type='text' size='20' name='lastname'></td>
</tr>
</tr>
<td colspan='2'><input type='submit' value='Зарегистрироваться'
name='submit'>
</td>
</tr>
</table>
</form>

<br>Поля со значком <font color='red'>*</font> обязательны
для заполнения

</body>
</html>

```