

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет

«Вычислительная техника»

Кафедра

«Системы автоматизированного проектирования»

Направление подготовки

09.03.01 «Информатика и вычислительная техника»

Профиль

«Системы автоматизированного проектирования»

БАКАЛАВРСКАЯ РАБОТА

на тему

*Разработка системы поддержки проведения
конференции*

Студент

19.05.17 Кудренко Я.Т.
(подпись, дата) (фамилия, инициалы)

Руководитель

1.06.17 Бермисов М.И.
(подпись, дата) (фамилия, инициалы)

Нормоконтролёр

доп. кат. САПР ПСУ 22.05.17 Тоугас Д.С.
(должность, место работы) (подпись, дата) (фамилия, инициалы)

Работа допущена к защите (протокол заседания кафедры от _____ № _____)

Заведующий кафедрой

Бермисов М.И.
(подпись) (фамилия, инициалы)

Работа защищена с отметкой _____ (протокол заседания ГЭК от _____ № _____)

Секретарь ГЭК

(подпись) (фамилия, инициалы)

Пенза 2017

РЕФЕРАТ

Пояснительная записка к выпускной квалификационной работе содержит: 93 страницы, 29 рисунков, 1 таблицу, 13 источников, 4 приложения.

СИСТЕМА ПРОВЕДЕНИЯ КОНФЕРЕНЦИИ, PHP, JAVASCRIPT, ОБРАБОТКА ЭЛЕКТРОННЫХ ЗАЯВОК, АВТОМАТИЗАЦИЯ, ОБРАБОТКА ДАННЫХ, РАССЫЛКА УВЕДОМЛЕНИЙ.

Цель работы: разработка системы, предназначенной для повышения эффективности проведения конференции и снижение трудовых затрат организаторов за счет автоматизации сбора, учета и обработки данных.

Объект исследований: программные средства автоматизированной системы.

Результатом данной работы является система поддержки проведения конференции, разработанная на языках программирования PHP, JavaScript в среде разработки Chrome Developer Tools.

Разработанная система может использоваться в учебных заведениях, учреждениях и организациях, где проводятся научные конференции.

					ВКР.090301.09.17.81 01				
Изм.	Лист	№ докум.	Подпись	Дата					
Разраб.		Курденкова Я.Г.			Разработка системы поддержки проведения конференции. Пояснительная записка.	Лит.	Лист	Листов	
Провер.		Бурукина И.П.							
Реценз.							3		
Н. Контр.		Бождай А.С				гр. 13вв2			
Утверд.		Бершадский А.М							

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	5
1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ	6
1.1 Описание предметной области	6
1.2 Обзор аналогов разрабатываемой системы.....	8
2. ОБОСНОВАНИЕ ВЫБОРА СРЕДСТВ РАЗРАБОТКИ	11
3. ПРОЕКТИРОВАНИЕ СИСТЕМЫ	14
3.1 Проектирование базы данных.....	15
4. РЕАЛИЗАЦИЯ СИСТЕМЫ	18
4.1 Реализация страницы с формой заявки	19
4.2 Реализация административной части	23
5. ТЕСТИРОВАНИЕ СИСТЕМЫ.....	47
ЗАКЛЮЧЕНИЕ	52
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	53
ПРИЛОЖЕНИЕ А «Диаграмма вариантов использования».....	55
ПРИЛОЖЕНИЕ Б «Схема работы системы».....	56
ПРИЛОЖЕНИЕ В «Схема компонентов системы»	57
ПРИЛОЖЕНИЕ Г «Исходный код».....	58

ВВЕДЕНИЕ

С течением времени, задача организации и проведения в высших учебных заведениях научных конференций, а также других мероприятий подобного направления, становится всё более востребованной и актуальной.

Это объясняется тем, что подобная деятельность поддерживает научный потенциал, даёт возможность студентам, преподавателям и сотрудникам проявить себя, раскрыть свои способности и таланты.

Несмотря на высокую степень развития компьютерных технологий и широкое распространение автоматизированных компьютерных средств в различные области деятельности человека, в большинстве случаев, работа по организации, ведению учета и информированию участников выполняется членами комиссии конференции вручную, что крайне неэффективно и может повлечь за собой немалое количество неточностей и проблем [1].

Использование системы поддержки проведения конференции может способствовать снижению трудовых затрат организаторов и уменьшению вероятности возникновения ошибок в процессе работы.

					<i>ВКР.090301.09.17.81</i>	Лист
Изм.	Лист	№ докум.	Подпись	Дата		5

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

Целью выпускной квалификационной работы является разработка системы поддержки проведения конференции.

Объектом работы является приложение, осуществляющее работу по сбору, обработке и анализу данных.

1.1 Описание предметной области

Проведение любой конференции – ответственное мероприятие, требующее тщательной подготовки [1].

В общем случае, процесс организации и проведения конференции можно разделить на три этапа:

- сбор заявок на участие в конференции;
- рассмотрение и утверждение заявок членами комиссии;
- проведение секционных заседаний.

В настоящее время, в большинстве случаев, для участия в конференции в качестве заявки необходимо отправлять свои данные и текст работы на электронный почтовый ящик секретариата конференции.

При таком способе организации данных, их обработка является трудоёмкой для члена комиссии, обрабатывающего поступающие заявки вручную, а также не обеспечивается даже начальный уровень безопасности контактных данных.

Наряду с другими необходимыми действиями, определенными регламентом проведения конференций, одно из основных мест занимает работа члена комиссии с почтовыми рассылками. Потенциальным участникам рассылаются информационные письма о проведении конференции и другие необходимые материалы. Эта работа выполняется членами комиссии вручную, что крайне неэффективно. Отсюда «вытекают» следующие проблемы:

- низкая скорость рассылки при большом объеме потенциальных участников и необходимости обращения к базе данных адресатов;
- большая вероятность ошибок (человеческий фактор);
- привязка к рабочему месту члена комиссии;
- слабая поддержка повторных рассылок [1].

Также, определенные сложности и неудобства может вызывать и формирование отчета об итогах проведения конференции с подсчетом количества фактических участников и другой организационной информацией.

Немаловажными факторами для обеспечения успешной организационной работы во время проведения конференции являются возможность вывода регистрационной формы заявок участников для сбора персональных данных (с учетом важности обеспечения недоступности личной контактной информации для секретаря конференции) и файла с научной работой, а также предоставление возможности членам комиссии анализировать полученные данные и управлять ими.

В процессе анализа предметной области и основополагающих задач, связанных с организацией вузовской научно-практической конференции, была выявлена потребность в создании системы, способной частично или полностью автоматизировать работу секретаря конференции и отвечающей следующим основным требованиям к функциональности:

- автоматизация массовых почтовых рассылок;
- формализованный сбор персональных данных;
- обработка полученных данных, анализ данных;
- формирование отчета.

1.2 Обзор аналогов разрабатываемой системы

Частично реализовать поставленные на этапе анализа предметной области задачи способны программы и сервисы массовых почтовых рассылок, например, Advanced Mass Sender Enterprise [2], Unisender [3], MailChimp [4].

Изображения главных экранов систем Unisender и MailChimp представлены на рисунках 1 и 2 соответственно.

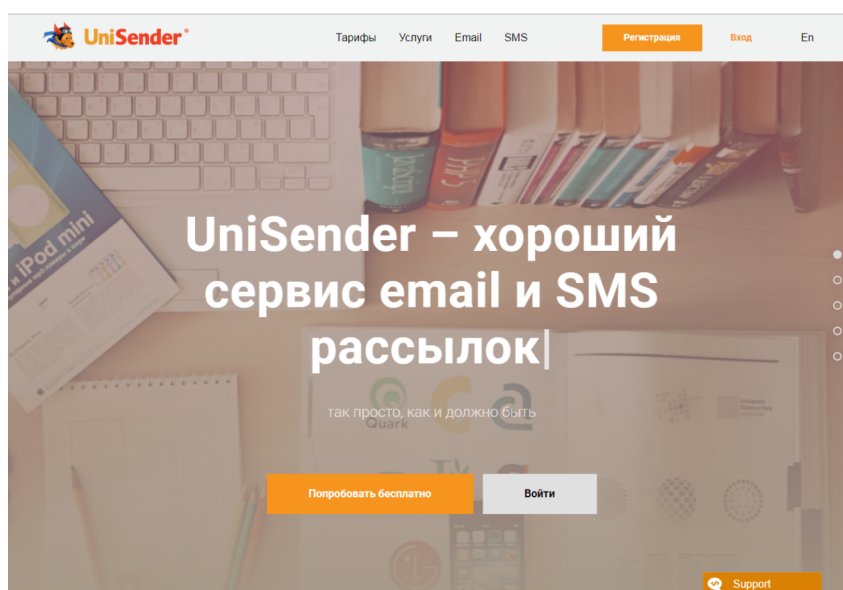


Рисунок 1 – Система Unisender

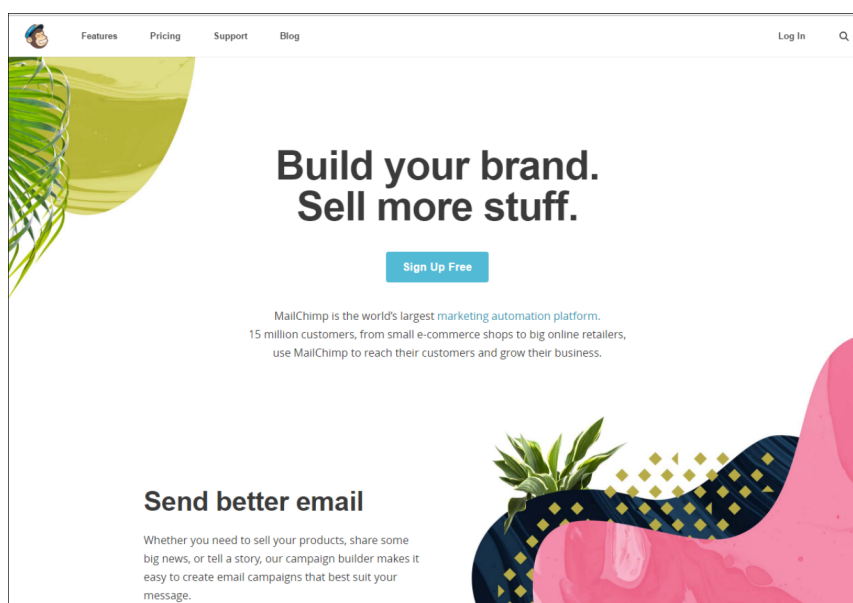


Рисунок 2 – Система MailChimp

Однако их функционала недостаточно для обеспечения широкой и качественной поддержки проведения конференции, обеспечения недоступности личной контактной информации участников для секретаря конференции, а также исключения влияния некоторых негативных факторов, например, ошибок в контактной или организационной информации.

Функционал больший, чем просто рассылку писем, предоставляют системы и сервисы, одной из которых является система YConfs (Сибирский федеральный университет), которая отчасти явилась прототипом для разрабатываемой в рамках данной выпускной квалификационной работы системы.

В функционал YConfs входят: возможность формализованного сбора данных, предоставление информационной страницы проводимой конференции и различная её настройка, обработка данных, поддержка отправки писем и прочее [5].

Однако, «упор» в данной системе сделан на работу с информационной страницей конференции, её внешним оформлением и управлением составляющими формы заявки, что не является основополагающим для задачи обеспечения снижения трудозатрат секретаря конференции.

Исходя из доступной на официальном сайте системы информации, в ней отсутствуют:

- тематическая массовая рассылка информационных писем с автоматически генерируемым содержанием;
- вывод графиков и диаграмм со статистическими данными;
- возможность отметки о присутствии участника на конференции в день проведения секционного заседания.

Сравнение основного функционала рассмотренных аналогов и разрабатываемой системы представлено в таблице 1.

					ВКР.090301.09.17.81	Лист
Изм.	Лист	№ докум.	Подпись	Дата		9

Таблица 1 – Сравнение аналогов

	Программы и сервисы массовых почтовых рассылок	Система YConfs	Разрабатываемая система
Формализованный сбор данных	-	+	+
Графики и диаграммы	-	-	+
Обработка данных	-	+	+
Разграничение прав доступа	-	+	+
Формирование отчета	-	+/-	+
Почтовая рассылка	+	+/-	+
Управление информационной страницей и её настройка	-	+	+/-

Важно отметить, что к вопросу о выборе или разработке подобной системы следует подходить с точки зрения оптимального решения прикладной задачи проведения вузовской или кафедральной научно-практической конференции, следовательно, внедрение сложного, избыточного по функционалу программного обеспечения может быть нецелесообразным.

Также, в целях обеспечения безопасности и во избежание прецедентов нарушения авторских прав или разглашения персональных данных, немаловажно учесть и тот факт, что, когда речь идет о работе с персональными данными и материалами уникальных научных трудов, следует тщательно анализировать необходимость использования программного обеспечения, разработанного сторонними организациями.

2. ОБОСНОВАНИЕ ВЫБОРА СРЕДСТВ РАЗРАБОТКИ

Выбор средств разработки реализуемой системы был сделан исходя из необходимости обеспечения следующих характеристик:

- кроссплатформенность системы;
- доступность с различного вида устройств;
- отсутствие привязки к одному рабочему месту;
- относительная простота поддержки и модификации.

HTML (Hyper TextMarkup Language) – стандартизированный язык разметки документов, на котором создаётся «каркас» внешнего вида системы.

CSS (Cascading Style Sheets) – каскадные таблицы стилей, формальный язык описания внешнего вида документа, написанного с использованием языка разметки [6].

Язык программирования JavaScript – самое распространённое средство создания браузерных интерфейсов, язык программирования с полной интеграцией с HTML/CSS, который поддерживается всеми распространёнными браузерами и включён по умолчанию [7].

Язык программирования PHP (Hypertext Preprocessor) – распространённый язык программирования, сконструированный специально для ведения Web-разработок, код может внедряться непосредственно в HTML. Достаточно прост в базовом освоении, но обладает необходимым функционалом по работе с базой данных [8].

В качестве среды разработки был выбран Chrome Developer Tools, представляющий собой набор инструментов для веб-разработки и отладки, встроенный в браузер Google Chrome.

Для хранения данных используется MySQL – компактный многопоточный сервер баз данных, характеризующийся большой скоростью,

устойчивостью и легкостью в использовании и поддерживающий язык запросов SQL [9].

Для непосредственного написания кода программных скриптов системы был использован бесплатный текстовый редактор Notepad++ 7.3.1, так как данная программа располагает широким набором опций, отличается минимальным потреблением ресурсов, а также имеет удобный интерфейс, возможности форматирования кода и подсветки текста.

Работа по разработке системы была развернута на локальном сервере, в качестве инструментария для обеспечения функционирования которого был использован базовый пакет Denwer (Денвер) – набор дистрибутивов и программная оболочка, предназначенные для удобства создания и отладки сайтов на локальном персональном компьютере без необходимости постоянного подключения к Интернету [10].

Дистрибутив данного набора был свободно скачан с официального сайта, главная страница которого представлена на рисунке 3.

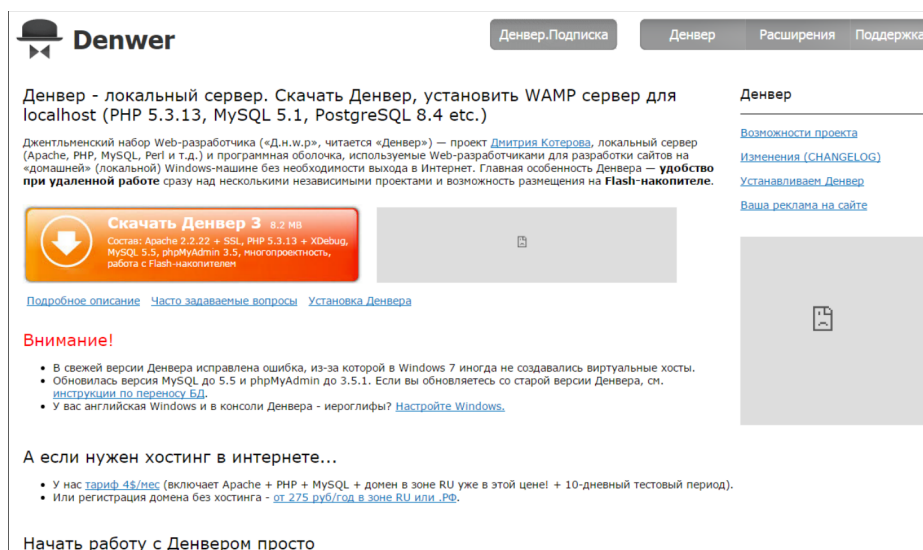


Рисунок 3 – Вид страницы сайта

При успешной установке и выполненном запуске, в браузере по адресу <http://localhost/denwer/> открывается страница, вид которой представлен на рисунке 4.

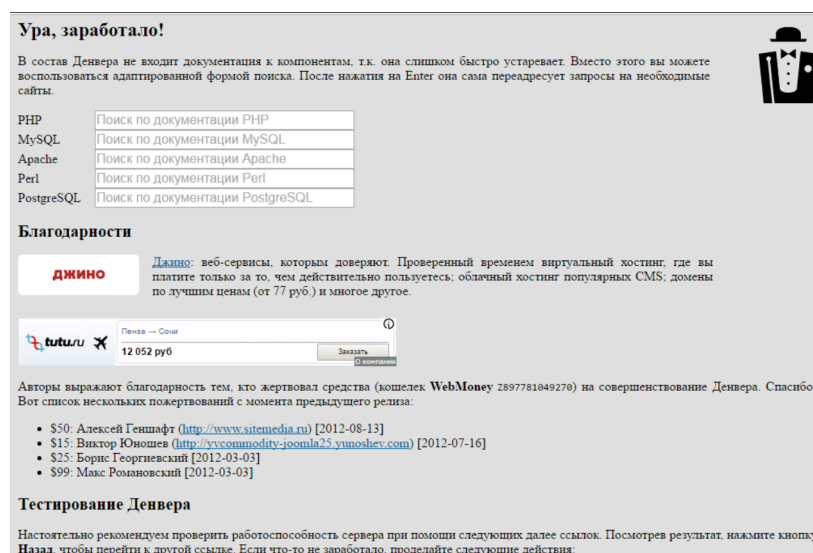


Рисунок 4 – Страница успешного запуска

Вести работу по созданию базы данных, входящих в её состав таблиц и управлению ими можно при помощи входящей в набор Денвера утилиты «phpMyAdmin», отвечающей за администрирование СУБД MySQL, вид страницы представлен на рисунке 5.

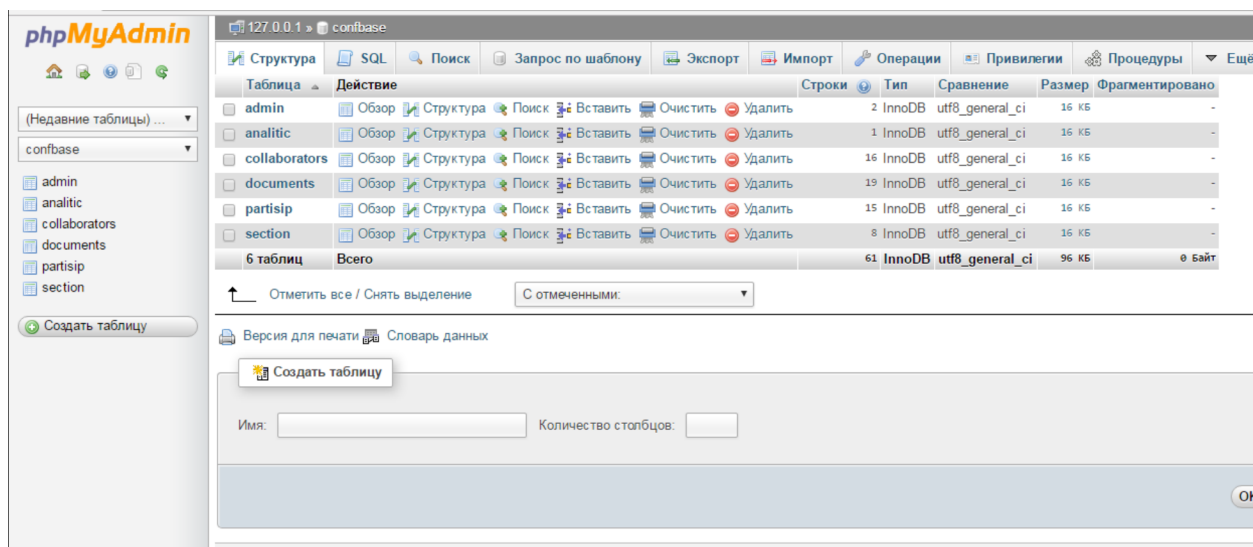


Рисунок 5 – «phpMyAdmin»

Установка комплекса интуитивно понятна, а необходимые настройки для проведения разработки системы в рамках выпускной квалификационной работы не являются сложными или излишне трудоёмкими.

3. ПРОЕКТИРОВАНИЕ СИСТЕМЫ

Разрабатываемая система предназначена для повышения эффективности проведения конференции за счет автоматизации сбора, учёта и обработки данных, помощи в планировании работы, автоматической рассылки информационных почтовых уведомлений, на основе данных из таблиц базы данных.

Работа системы поддержки проведения конференции нацелена на повышения эффективности взаимодействия с двумя основными независимыми друг от друга группами пользователей, к которым относятся:

- потенциальный участник конференции, желающий подать заявку на участие;
- администратор системы – секретарь комиссии конференции, отвечающий за работу с заявками;

Поэтому, структуру разрабатываемой системы необходимо разделить на две составляющие:

- часть системы, ориентированная на работу с заявителями – форма заявки на информационной странице конференции;
- административная часть системы, функционал которой позволит обеспечить снижение трудозатрат секретаря конференции на учёт и обработку данных;

Диаграмма вариантов использования системы представлена в приложении А.

В ходе проектирования системы поддержки организации и проведения вузовской или кафедральной научно-практической конференции к основными необходимым для реализации функциям были отнесены:

- вывод страницы с формой подачи заявок на участие в конференции и управление ей, непосредственно из административной части системы;

- организация сбора заявок на участие в виде набора персональных данных (ФИО, факультет/кафедра, соавторы работы, секция, адрес электронной почты) и текстового файла с научной работы;
- создание административной части системы с разграничением прав доступа;
- отображение списка всех полученных заявок с сокрытием персональных контактных данных и обработку заявок секретарём конференции с возможностью редактирования и удаления как заявок, так и организационной информации по конференции и по секционным заседаниям;
- реализация массовой почтовой рассылки информационных писем с автоматически генерируемым из базы данных содержимым;
- возможность отправки личных электронных почтовых сообщений участникам конференции;
- реализацию формирования отчета об итогах проведения конференции;
- формирование статистических графиков и диаграмм по данным, сформированным из запросов к базе данных.

3.1 Проектирование базы данных

Исходя из набора необходимых для реализации функций, хранение данных, обеспечивающих правильное функционирование системы решено было организовать так, как представлено на рисунке 6.

					<i>ВКР.090301.09.17.81</i>	Лист
Изм.	Лист	№ докум.	Подпись	Дата		15

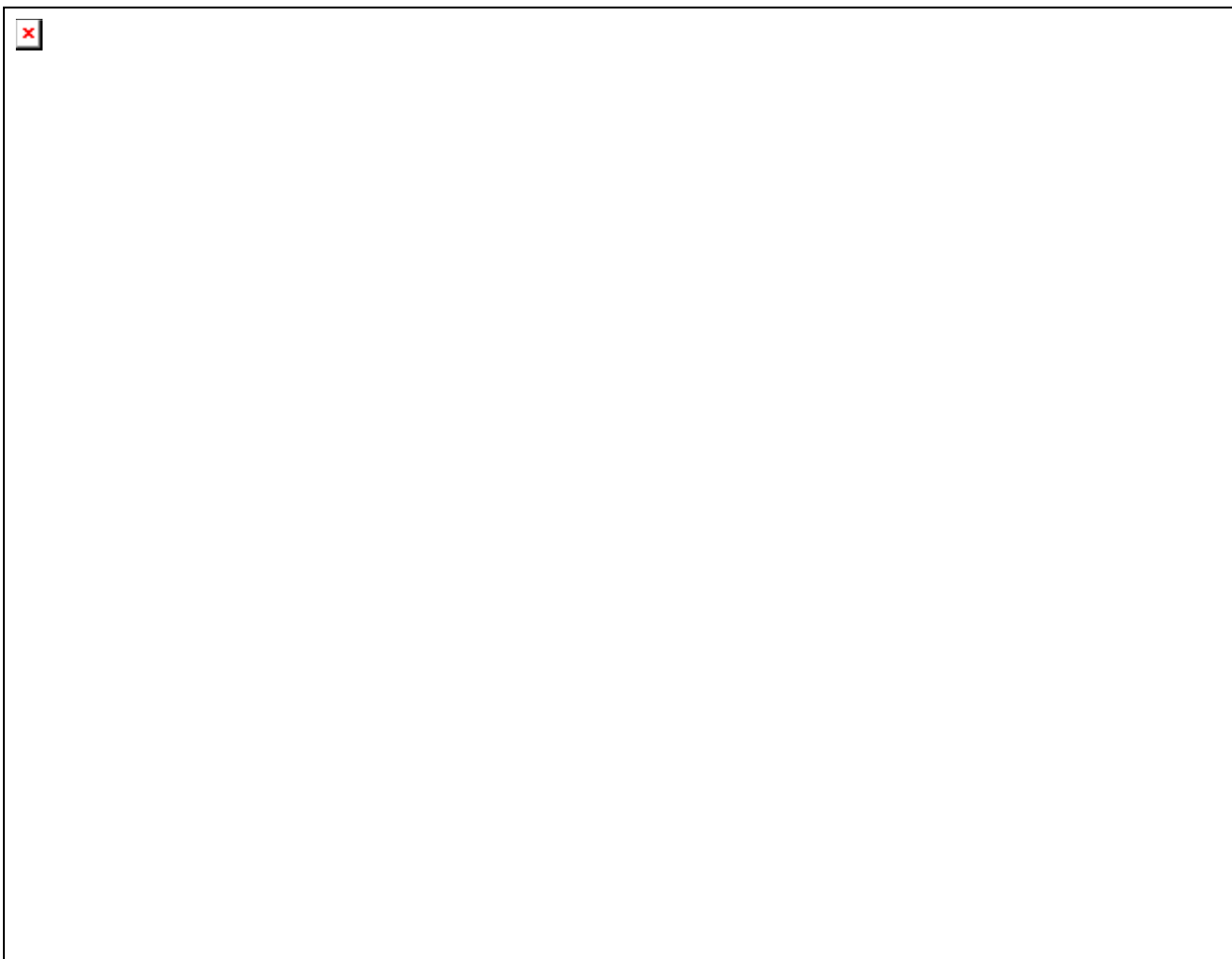


Рисунок 6 – Схема базы данных «confbase»

База данных системы «confbase» содержит 6 таблиц:

- partisip – таблица содержит данные об участниках конференции, подавших заявку.

Данные для хранения в таблице: ID, фамилия, имя, отчество, ID секции, ID документа, дата и время подачи заявки, адрес электронной почты, флаг утверждения заявки, флаг отметки о присутствии в день проведения конференции, кафедра/факультет участника;

- collaborators – таблица содержит данные о соавторах работ участников конференции, подавших заявку.

Данные для хранения в таблице: ID, ФИО соавтора, ID автора, подавшего заявку;

- section – таблица содержит данные о секциях конференции.
Данные для хранения в таблице: ID, название секции, дата проведения, время проведения, место проведения;
- documents – таблица содержит данные о файлах научных работ.
Данные для хранения в таблице: ID, название файла для хранения на сервере, название работы для отображения, путь к файлу, дата изменения файла, размер файла;
- admin – таблица содержит данные о логинах и паролях для аутентификации в административной части системы.
Данные для хранения в таблице: ID, логин администратора, пароль администратора;
- analitic – таблица содержит организационные данные.
Данные для хранения в таблице: ID, количество отправленных писем, состояние страницы с формой заявки (открыта/закрыта), название конференции, дата проведения, дата окончания приёма заявок.

4. РЕАЛИЗАЦИЯ СИСТЕМЫ

В ходе разработки системы поддержки проведения конференции при помощи выбранных средств разработки были реализованы основные функции, определенные на стадии проектирования ПО.

Весь процесс разработки можно разделить на несколько основных частей:

- реализация «каркаса» и внешнего вида страниц системы средствами HTML и CSS, нацеленная на создание простого и дружелюбного интерфейса пользователя, когда алгоритм взаимодействия с системой интуитивно понятен, не вызывает сложностей, а оформление является лаконичным и эстетически гармоничным;
- реализация функций на языке JavaScript;
- реализация «динамики» формирования страниц и логики работы системы средствами языка программирования PHP, а также взаимодействие с базой данных посредством различного вида SQL-запросов к таблицам и обработкой полученных результатов необходимым образом на языке PHP.

Схема работы системы, в которой отражены основные функции и их взаимодействие, представлена в приложении Б.

Схема компонентов системы, на которой представлены основные компоненты системы и их взаимодействие представлена в приложении В.

Файл «style.css», содержимое которого определяет внешнее оформление страниц системы представлен в приложении Г.

4.1 Реализация страницы с формой заявки

Данная часть разрабатываемой системы ориентирована на потенциальных участников конференции.

Внешний вид данной страницы схож с типичным вариантом разметки страницы абстрактного сайта.

При статусе формы заявки «открыта» на информационной странице конференции отображается форма со следующим набором элементов:

- полем ввода фамилии;
- полем ввода имени;
- полем ввода отчества;
- полем ввода факультета;
- полями ввода соавторов;
- выпадающим списком секций;
- полем для выбора файла работы;
- полем ввода почтового ящика;
- полем для отметки о согласии на обработку персональных данных;
- кнопкой отправки заявки.

Название конференции, дата проведения и дата окончания приёма заявок выводятся из соответствующих полей таблицы «analitic» базы данных, посредством запроса, реализованного в файле «index.php», код которого представлен в приложении Г.

Вид данной страницы представлен на рисунке 7.

<p>Заявка на участие в XII Международной научно-технической конференции НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И СИСТЕМЫ "НИТИС-2017"</p> <p>Сроки проведения: 21.01.2017-20.05.2017</p> <p>Приём заявок до: 01.05.2018</p> <p>Введите фамилию*</p> <p>Введите имя*</p> <p>Введите отчество</p> <p>Укажите факультет(например, ФВТ)*</p> <p>☐ Указать соавторов</p> <p>Выберите секцию*</p> <p>Не указано ▼</p> <p>Загрузить файл(docx/doc/odt)*</p> <p>Выберите файл Файл не выбран</p> <p>Укажите email(адрес электронной почты)*</p> <p>☐ Я согласен(на) на обработку персональных данных*</p> <p>Отправить заявку на участие</p> <p>* - поля, обязательные для заполнения</p>	
---	--

Рисунок 7 – Страница с формой заявки в статусе «открыта»

Также, значения полей «название конференции», «дата проведения» и «дата окончания приёма заявок» можно изменять в административной части системы.

В текстовые поля формы предполагается ввод указанной в подписях к полям информации.

Если у автора работы, заявляемой на конференцию есть соавторы, заявитель может указать их, раскрыв список полей для ввода соавторов по нажатию на checkbox рядом с соответствующей надписью.

По нажатию на кнопку «Выбрать файл» открывается окно выбора файла, где заявитель может выбрать файл со своей работой из каталогов персонального компьютера.

Важным моментом является постановка отметки в поле «Согласие на обработку персональных данных».

Это действие является обязательным, так как без согласия заявителя данные не могут быть приняты, согласно п.1, ст.6, Федерального закона от 27 июля 2006 г. N 152-ФЗ "О персональных данных": «обработка персональных данных осуществляется с согласия субъекта персональных данных на обработку его персональных данных» [11].

На странице содержатся подробные подписи-инструкции по заполнению полей для обеспечения максимальной корректности указанных в заявке данных.

Проверка введённых данных на соответствие требованиям осуществляется при каждом нажатии на кнопку отправки заявки функцией check() на языке JavaScript, реализованной в файле «index.php».

Поля «фамилия», «имя», «отчество» допускают ввод русских и латинских символов, а также «-» и «`» для написания составных имён и фамилий.

Поле «факультет» допускает ввод только русскоязычных символов.

Поля списка «соавторы» допускают ввод русских и латинских символов, «-» и «`», а также «.» для указания инициалов.

Если соавторов несколько, заполнять данные поля нужно последовательно, не оставляя пустых строк между записями.

Секцию предлагается выбрать из выпадающего списка, сформированного из результата запроса к таблице «section» базы данных или оставить её не указанной.

Поле «email» проверяется на корректность в соответствии с набором символов, допустимых при написании адреса электронной почты.

После нажатия на кнопку, подтверждающую отправку заявки, данные попадают на сервер, где записываются в соответствующие поля таблиц базы данных.

Затем, на указанный в заявке адрес электронной почты направляется информационное письмо с именным обращением.

Программный скрипт «registration.php», в котором реализована запись информации в таблицы базы данных, переданных из формы представлен в приложении Г.

При статусе формы заявки «закрыта» вместо формы заявки отображается страница, представленная на рисунке 8.

Название конференции и дата завершения приёма заявок также выводятся из результата запроса к таблице базы данных.

	<p>Приём заявок на участие в XII Международной научно-технической конференции НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И СИСТЕМЫ "НИТИС-2017"</p> <p>завершился: 01.05.2018</p> <p>Будем рады Вашему участию в наших следующих мероприятиях!</p> <p>С уважением, Организатор.</p>

Рисунок 8 – Страница с формой заявки в статусе «закрыта»

4.2 Реализация административной части

Административная часть системы является основной и в ней осуществлено большинство из тех функций, необходимость реализации которых была установлена на этапе проектирования системы.

Работа системы нацелена на достижение снижения трудозатрат потенциального пользователя административной части, поэтому данная часть системы разрабатывалась таким образом, чтобы взаимодействие с элементами меню и страниц было как можно более простым для восприятия и интуитивно понятным для рядового работника без специального уровня технических знаний.

Для входа в административную часть системы необходимо дописать /admin после основного адреса сайта.

Если ранее не была выполнена аутентификация, секретарь конференции (далее, пользователь) будет направлен на страницу входа в систему, где ему будет предложено ввести логин и пароль для аутентификации.

Функция аутентификации в административной части системы реализована на основе механизма сеансов с поддержкой автоматической аутентификации благодаря использованию cookie. Реализация данной функции в файлах «authclass.php» и «autorisation.php» и представлена в приложении Г.

Страница входа в административную часть системы представлена на рисунке 9.

					ВКР.090301.09.17.81	Лист
Изм.	Лист	№ докум.	Подпись	Дата		23

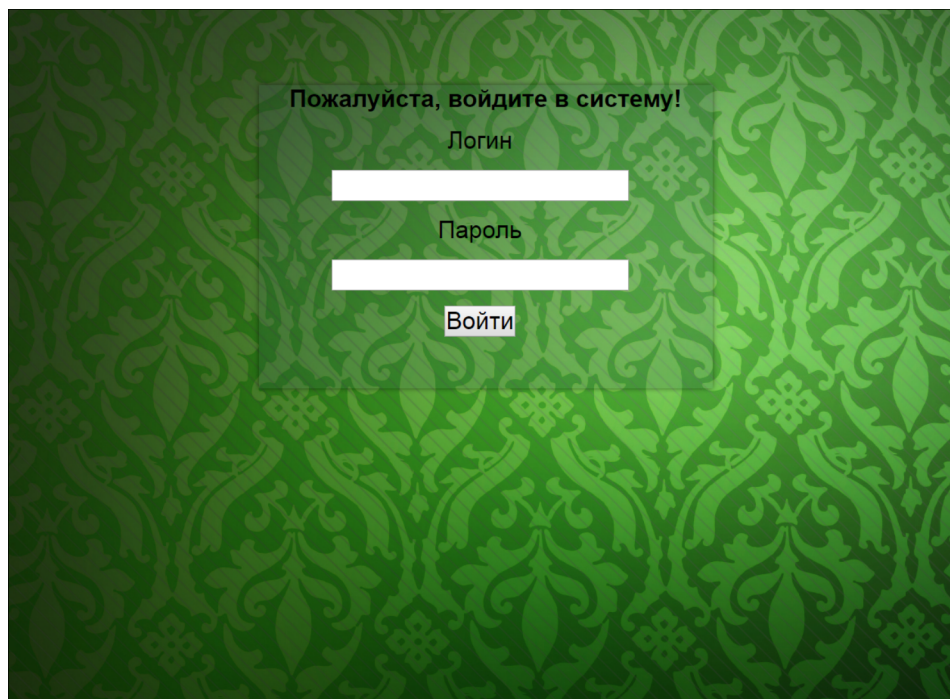


Рисунок 9 – Страница аутентификации

Если логин и пароль администратора введены неверно, вход в систему не будет осуществлён, а на странице появится надпись: «Неверный логин или пароль».

При совпадении введенных логина и пароля с теми, что хранятся в таблице «admin» базы данных, пользователь будет перенаправлен на главную страницу административной части системы «Список заявок».

Внешний вид страницы «Список заявок» с двумя уже утверждёнными заявками представлен на рисунке 10.

Тело таблицы списка заявок формируется в скрипте «spisok.php», посредством обработки и вывода результатов запросов к таблицам «partisip», «collaborate», «section», «documents» базы данных.

Код данного скрипта представлен в приложении Г.

Для списка заявок реализованы функции удаления заявок, утверждения заявок, отметки явившегося на секционное заседание участника, а также изменения секций, выбранных заявителями изначально при регистрации.

Подход к реализации данных функций осуществлялся с точки зрения максимального удобства для пользователя системы.

Функция удаления заявок необходима для того, чтобы убрать из списка случайные заявки или заявки, окончательно отклонённые при рассмотрении членами комиссии работы заявителя.

Пользователь может удалить как одну запись, так и выбрать для удаления сразу несколько записей из списка заявок, поставив галочку рядом с порядковым номером записи. После нажатия кнопки с надписью «удалить выбранные» рядом с изображением корзины произойдёт удаление записей из таблицы заявок, а также из соответствующей таблицы базы данных.

Фрагмент программного скрипта «partisipant.php» на языке PHP, в котором реализована данная функция представлен ниже:

```
if (isset($_POST["delete"])) //если нажата кнопка "удалить выбранные"
{
include ('../connection_db.php'); //подключение к БД
$deleteitem = $_POST['flagdelete']; //массив выбранных
$N = count($deleteitem); //количество выбранных

if (!($stmt = $mysqli-> prepare("SELECT * FROM `partisip`"))) {//
подготавливаемый запрос выборку

echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " .
$mysqli->error;
```

```

    }

    if (!$stmt->execute()) {
        echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error; }

    if (!$res = $stmt->get_result()) {
        echo "Не удалось получить результат: (" . $stmt->errno . ") " . $stmt->error;}

    if (!$stmt = $mysqli->prepare("DELETE FROM `partisip` WHERE `ID` = ?")) {// подготавливаемый запрос на удаление заявки
        echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;}

    if (!$stmt->bind_param("i", $IDel)) {//привязка параметров
        echo "Не удалось привязать параметры: (" . $stmt->errno . ") " . $stmt->error;
    }

    if (!$stmtdoc = $mysqli->prepare("DELETE FROM `documents` WHERE `ID` = ?")) {// подготавливаемый запрос на удаление документа данного заявителя
        echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;}

    if (!$stmtdoc->bind_param("i", $IDelldoc)) {//привязка параметров
        echo "Не удалось привязать параметры: (" . $stmt->errno . ") " . $stmt->error;
    }

    if (!$stmtcoll = $mysqli->prepare("DELETE FROM `collaborators` WHERE `par_ID` = ?")) {// подготавливаемый запрос на удаление соавт. заявителя
        echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;}

    if (!$stmtcoll->bind_param("i", $IDel)) {//привязка параметров
        echo "Не удалось привязать параметры: (" . $stmt->errno . ") " . $stmt->error;
    }

```

```

while ($rows = $res->fetch_assoc())
{
for($i=0; $i <= $N; $i++)
{
if ($deleteitem[$i]==$rows['ID'])
{ $IDel=$deleteitem[$i];
$IDeldoc=$rows['doc_ID'];
if (!$stmt->execute()) { //выполнение запроса на удаление заявки
echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}
if (!$stmt->doc->execute()) { //выполнение запроса на удаление документа
echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}
if (!$stmt->coll->execute()) { //выполнение запроса на удаление соавторов
echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
} } }
$stmt->close();
$mysqli->close();
header("Location: index.php");
}

```

Функция утверждения заявок является очень важной в реализации логики работы всей системы в целом.

Её использование напрямую влияет на результаты массовых информационных почтовых рассылок, проводимых после закрытия формы заявок на странице конференции.

Также, функция «утверждения заявок» влияет на отображение списка утверждённых участников, распределённых по секционным заседаниям в разделе «Секции».

Данные в сформированном по итогам проведения конференции отчете также зависит от числа утверждённых участников.

Изначально, в строке каждой новой записи в списке заявок отображается значок «желтый круг», означающий, что заявка находится на рассмотрении и еще не была утверждена.

После ознакомления членов комиссии конференции с текстом присланной участником работы и принятия решения об утверждении заявки данного участника, администратор, кликом на значок, упомянутый выше, меняет его. При этом, меняется и значение соответствующего поля записи в таблице «partisip» базы данных.

Обозначение сменится на значок «Галочка», а вся строка с данными заявки участника будет выделена зеленым цветом - это означает, что заявка утверждена.

Для возможности исправления ошибочного утверждения заявок участников была также реализована обратная функция – «отмена утверждения заявки на участие».

Немаловажной с точки зрения автоматизации процесса организации и проведения конференции является возможность сделать отметку о явке участника на секционное заседание в день проведения или о факте его выступления.

В разрабатываемой системе данная функция аналогично функции «утверждение заявки» влияет на работу других функций, например, на массовую почтовую рассылку писем с благодарностью за участие и на формирование данных в итоговом отчете.

Реализована функция следующим образом: в последнем столбце каждой строки списка находится поле для отметки, при клике на которое на экране появляется сообщение «Принято», а само поле меняет своё состояние на отмеченное.

Средствами асинхронного запроса к базе данных, реализованного в файлах «participant.php» и «come.php», соответствующее поле в таблице «partisip» базы данных также меняет своё значение.

Поле «явка» в измененном и увеличенном состоянии означает, что данный участник был отмечен как присутствующий или фактически выступивший на секционном заседании проводимой конференции.

Фрагменты программного скрипта, в котором реализованы описанные выше функции представлены в приложении Г.

Если пользователь не указал секцию при заполнении заявки или члены комиссии после ознакомления с текстом работы участника приняли решение о смене секции, в которой будет выступать участник, секретарь конференции может изменить указанную ранее секцию участника.

Страница редактирования, на которой пользователь может изменить секцию участника по результату рассмотрения его заявки представлена на рисунке 11.

№	Фамилия	Имя	Отчество	Секция
1	Петров	Иван	Иванович	"Методы математического моделирования в научных исследованиях"
2	Петров	Сидр	Иванович	"Методы математического моделирования в научных исследованиях"
3	Иванов	Иван	Иванович	"Геоинформационные технологии"
4	Васильев	Сидр	Федорович	"Системы автоматизированного проектирования"
5	Елисеева	Светлана		"Системы автоматизированного проектирования"
6	Васильев	Сергей	Иванович	"Сети ЭВМ и телекоммуникации"
7	Иванов	Иван	Иванович	"Программное и аппаратное обеспечение информационно-вычислительных систем"
8	Куринов	Степан	Федорович	"Управление в социальных, экономических и технических системах"
9	Фурманов	Александр	Игоревич	"Программное и аппаратное обеспечение информационно-вычислительных систем"
10	Нерогова	Татьяна	Олеговна	"Проектирование и администрирование информационных систем"
11	Иванов	Сергей	Сидорович	Не указано

Рисунок 11 – Редактирование секций в заявках

Секцию для изменения предлагается выбрать из выпадающего списка всех заявленных на конференции секциях, сформированного по результату запроса к таблице «section» базы данных.

Первой в списке стоит текущая секция потенциального или утверждённого участника.

При выборе нужной секции из списка и нажатии на кнопку с изображением галочки произойдёт изменение значения соответствующего поля таблицы «particip» базы данных.

Из режима редактирования можно выйти без изменений, нажав на кнопку с изображением крестика.

По нажатию на название работы участника на странице «Список заявок», пользователю будет предложено скачать файл работы участника для дальнейшего ознакомления членов комиссии с содержанием научной работы и принятия решения об утверждении заявки на участие в конференции.

На каждой странице административной части слева отображается меню с кнопками перехода к соответствующим страницам системы:

- «Заявки»;
- «Секции»;
- «Рассылки»;
- «Отчетность».

А также кнопки изменения состояния формы заявок: открыть и закрыть.

В верхней части меню выводится имя(логин) администратора системы.

В меню также отображены:

- информация об общем количестве принятых заявок на текущий момент;
- число уже утверждённых заявок;
- дата, когда необходимо закрыть форму приёма заявок.

Ниже представлен скрипт «left_menu.php», в котором реализовано меню системы:

```

<div class="menu">
<?
$nowlogon=getLogin();//получаем логин пользователя
    echo "<div class='thisadmin'> <img width='30px' height='30px'
src='/img/korona.png'>Пользователь: ".$nowlogon."</div>";?>
        </br><p>                <b>Всего                заявок:                </b>
<?include('sum_partsip.php')?></p>//кол-во заявок всего
        <p>                <b>Утверждено                заявок:
</b><?include('sum_partsip_app.php')?></p>//кол-во заявок утверждённых
        //пункты меню
        <a href="partisipant.php"><button class="abutton b_partisip"
></button><label></br>Список заявок</br></label></a>

        <a href="sections.php"><button class="abutton b_section"
></button></br><label>Секции</label></a></br>
        <a href="mail.php"><button class="abutton b_mail"
></button></br><label>Рассылки</label></a></br>
        <a href="report.php"><button class="abutton b_otchet"
></button></br><label>Отчетность</label></a>

        <?include ('../connection_db.php'); //подключение к БД
        $quer = "SELECT * FROM analitic";
        $result = $mysqli->query($quer);
        $row = $result->fetch_assoc();

        if ($row[is_open]==1)
        { $isreg='Открыта';}
        else

```

```

        {$isreg='Закрыта';}
        echo "<div class=" "> </br> <b>Форма заявки:</b> ".$isreg."</div>";
        ?>

        <form action="" method="get " class="b_form">

        <input class="abutton b_close" name="is_closereg" type="submit" value="
"></br><label>Закрыць</label>

        </form>

        <form action="" method="get" class="b_form">

        <input class="abutton b_open" name="is_openreg" type="submit"
value=""></br><label>Открыць</label>

        </form>

        <?

        echo "<div class=" "> </br> <b>Форма заявок должна быть закрыта </b>
".$row['conf_date_open']."</div>";

        ?>

        <? include ('../connection_db.php');

        if (isset($_GET["is_openreg"])) { //если нажата кнопка «Открыть»

                $qwe="UPDATE `analytic` SET `is_open`= '1' ";

                $mysqli->query($qwe);

        header("Location: index.php");

        }

        if (isset($_GET["is_closereg"])) { //если нажата кнопка «Закрыць»

                $qwe="UPDATE `analytic` SET `is_open`= '2' ";

                $mysqli->query($qwe);

                header("Location: index.php");

        } ?> </div>

```

По нажатию на кнопку меню «Секции» пользователю открывается страница «Секции», на которой содержится список наименований секционных заседаний проводимой конференции.

В данном разделе реализованы функции:

- добавления новых секций;
- удаления секций;
- редактирования организационной информации по месту, времени и дате проведения каждого секционного заседания.

Добавление и удаление секции защищено дополнительной авторизацией и доступно только пользователям, которые знают соответствующий логин и пароль, записанные в базе данных.

На данной странице также реализована возможность просмотра списка участников с названиями их работ с утвержденными заявками для каждой секции по клику на checkbox рядом с названием.

Страница редактирования организационной информации по секционным заседаниям и просмотр списка утверждённых участников для каждой из секций отображены на рисунках 12 и 13 соответственно.

Пользователь: myadmin

Редактирование

Всего заявок: 11
Утверждено заявок: 5

Список заявок

Секции

Рассылки

Отчетность

№	Секция	Место проведения	Дата	Время
1	"Проектирование и админис"	ПГУ, корпус 7, аудитория 21	08.09.2017	14.00
2	"Системы автоматизированн"	ПГУ, корпус 7, аудитория 21	07.09.2017	13.00
3	"Методы математического м"	ПГУ, корпус 7, аудитория 21	09.09.2017	15.00
4	"Программное и аппаратное"	ПГУ, корпус 7, аудитория 21	09.09.2017	14.00
5	"Управление в социальных."	ПГУ, корпус 7, аудитория 31	07.09.2017	12.00
6	"Тесинформационные технс"	ПГУ, корпус 7, аудитория 31	08.09.2017	13.00
7	"Сети ЭВМ и телекоммуник"	ПГУ, корпус 7, аудитория 21	09.09.2017	12.15

Форма заявки: Открыта

Закрыть

Открыть

Форма заявок должна быть закрыта
01.05.2018

Пользователь:
myadmin

Всего заявок: 11
Утверждено заявок: 5

Список заявок

Секции

Рассылки

Отчетность

Форма заявки: Открыта

Закреть Открыть

Форма заявок должна быть закрыта
01.05.2018

Секции

№	Секция	Место проведения	Дата	Время
1	☐ "Проектирование и администрирование информационных систем"	ПГУ, корпус 7, аудитория 210	08.09.2017	14.00
2	☒ "Системы автоматизированного проектирования" Васильев Сидр Федорович - Научная работа	ПГУ, корпус 7, аудитория 210	07.09.2017	13.00
	Елисеева Светлана - Научная работа олен документ			
3	☒ "Методы математического моделирования в научных исследованиях" Петров Иван Иванович - Научная работа	ПГУ, корпус 7, аудитория 210	09.09.2017	15.00
	Петров Сидр Иванович - Научная работа олен документ			
4	☐ "Программное и аппаратное обеспечение информационно-вычислительных систем"	ПГУ, корпус 7, аудитория 210	09.09.2017	14.00
5	☐ "Управление в социальных, экономических и технических системах"	ПГУ, корпус 7, аудитория 310	07.09.2017	12.00
6	☐ "Геоинформационные технологии"	ПГУ, корпус 7, аудитория 310	08.09.2017	13.00
7	☒ "Сети ЭВМ и телекоммуникации" Васильев Сергей Иванович - Научная работа	ПГУ, корпус 7, аудитория 215	09.09.2017	12.15

- удалить

- редактировать все

Добавить новое направление

Одно из ведущих мест в функционале реализованной системы занимает работа с электронной почтой, а именно, с массовыми информационными рассылками по участникам конференции.

Всего в системе реализовано 4 различных вида способа работы с почтовыми рассылками:

1. Отправка письма с именным обращением при подаче заявки на участие в конференции с информационным содержанием.

Пример такого письма представлен на рисунке 14;

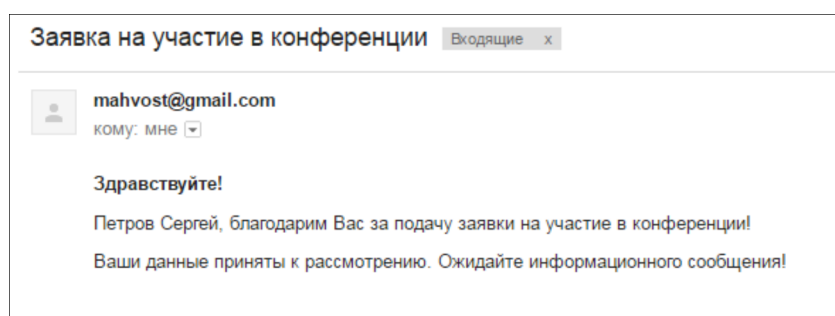


Рисунок 14 – Информационное письмо

2. Массовая рассылка с автоматически генерируемым содержанием письма на основе следующих данных, полученных из результата запроса к базе данных:

- ФИО участника;
- Наименование, дата, место и время проведения секционного заседания;

Данный вид рассылки осуществляется по участникам, заявки которых утверждены и становится доступной только после закрытия формы регистрации заявок участников.

Пример такого письма представлен на рисунке 15;

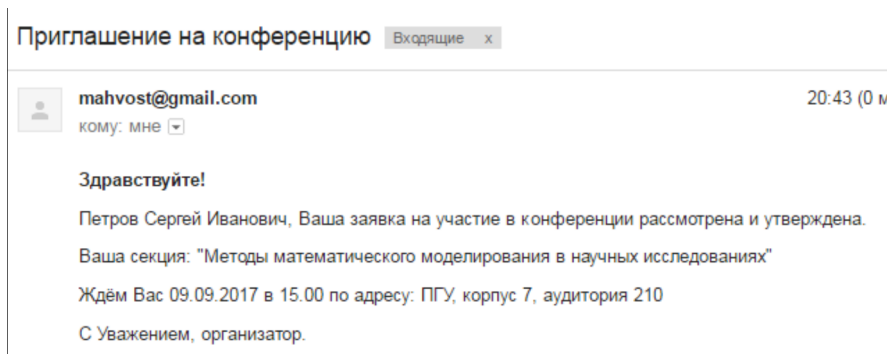


Рисунок 15 – Информационное письмо

3. Массовая рассылка с благодарностью за участие в конференции с автоматически генерируемым содержанием письма на основе данных (ФИО участника) полученных из результата запроса к базе данных, рассылаемая по участникам, фактически присутствовавшим на конференции. Пример такого письма представлен на рисунке 16;

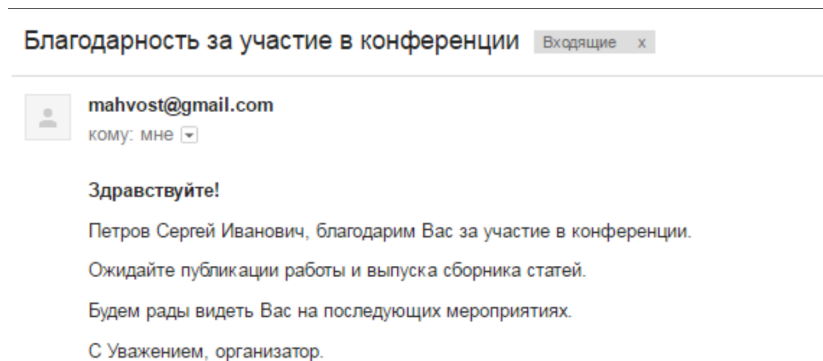


Рисунок 16 – Письмо с благодарностью за участие

4. Личное почтовое сообщение с именным обращением (ФИО участника, взятое из результата запроса к базе данных) и индивидуальным информационным текстом, набранным администратором системы.

Организация почтовых рассылок в разрабатываемой системе выполнена таким образом, что пользователю административной части системы недоступны личные контактные данные участников, а именно адреса электронной почты.

При нажатии на кнопку меню «Рассылки» происходит переход на страницу «Управление почтовыми рассылками». На которой пользователь может осуществлять различные почтовые рассылки участникам конференции.

Вид данной страницы при состоянии формы заявок «Открыта» с недоступными кнопками массовых рассылок представлен на рисунке 17.

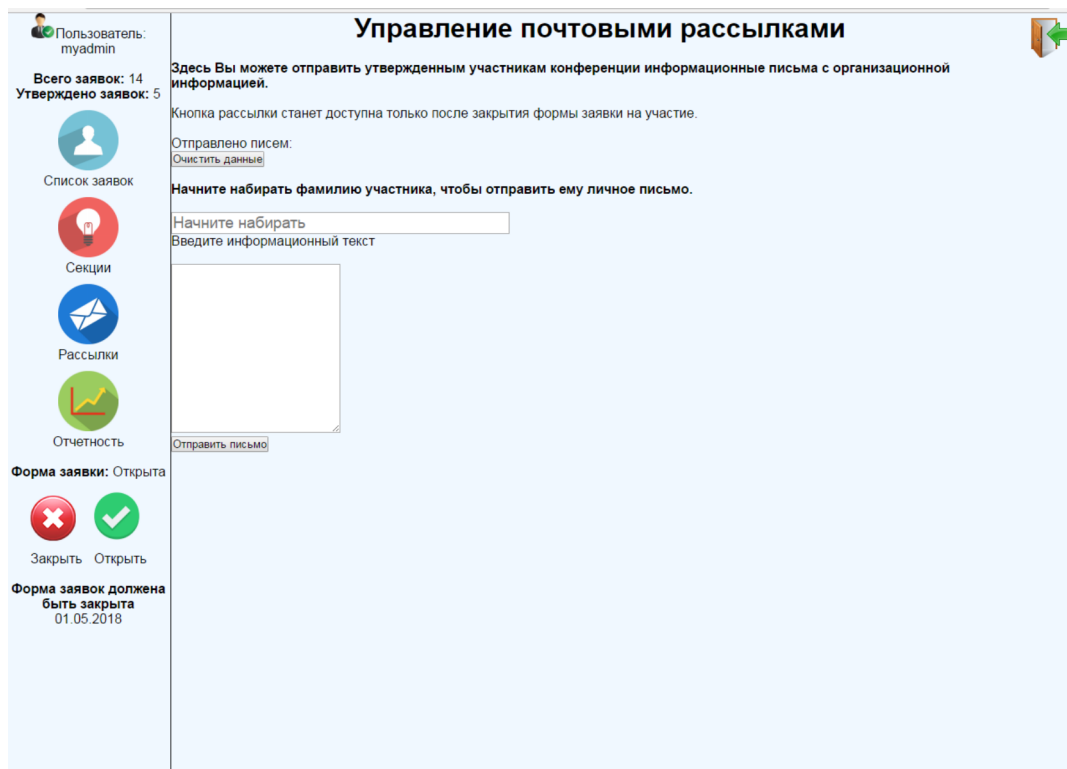


Рисунок 17 – Страница «Управление почтовыми рассылками»

Вид данной страницы с доступными кнопками рассылки при состоянии формы заявок «Закрота» представлен на рисунке 18.

 Пользователь:
myadmin

Управление почтовыми рассылками



Всего заявок: 11
Утверждено заявок: 5


Список заявок


Секции


Рассылки


Отчетность

Здесь Вы можете отправить утвержденным участникам конференции информационные письма с организационной информацией.

 - разослать письма с местом, датой и временем проведения его секционного заседания.

Вы можете отправить участникам конференции письма с благодарностью за участие.

 - разослать письма с благодарностью участникам, присутствовавшим на секционных заседаниях.

Отправлено писем: _____
[Очистить данные](#)

Начните набирать фамилию участника, чтобы отправить ему личное письмо.

Форма заявки: Закрыта

Закрыть Открыть

Форма заявок должна быть закрыта
01.05.2018

Скрипты «mail.php», «mail_org.php», «finalymail.php», «search.php» реализующие работу с почтовыми рассылками посредством функции mail() и асинхронный запрос поиска нужного адресата для 4-го вида рассылок представлены в приложении Г.

- указать или изменить название проводимой конференции;
- указать или изменить сроки проведения конференции;
- указать или изменить дату окончания приёма заявок на участие в конференции;
- открыть отчет по итогам конференции, данные для которого формируются по специально подготовленным результатам запросов к базе данных;
- ознакомиться со статистической информацией в виде диаграмм «Распределение поданных заявок по секциям», «Количество

поданных заявок по факультетам», «Количество поданных заявок по дням».

Внешний вид страницы «Отчетная информация» представлен на рисунке 19.

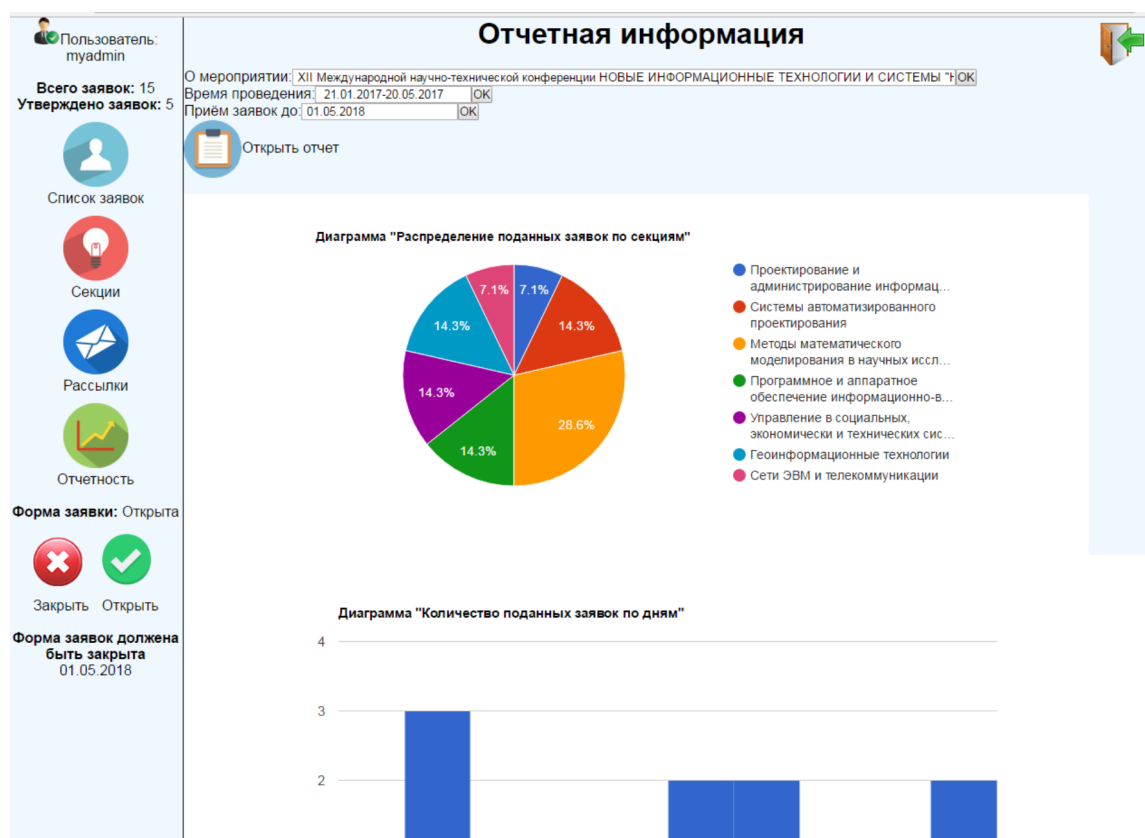


Рисунок 19 – Страница «Отчеты»

При переходе на данную страницу диаграммы отображаются в области просмотра не полностью. Чтобы ознакомиться со всеми тремя представленными в системе диаграммами необходимо «прокрутить» страницу вниз.

Для построения и вывода красочных и понятных диаграмм были использованы функции многофункционального набора инструментов для визуализации данных «Google Chart Tools API» [12].

Наборы данных для построения реализованных диаграмм, отображающихся в данном разделе системы, сформированы посредством

выполнения запросов к таблицам «partisip», «section» базы данных «confbase».

Для более наглядного отображения и простоты анализа представленных на диаграммах, данных для реализации были выбраны два вида: PieChart (круговая) и ColumnChart.

Круговая диаграмма «Распределение поданных заявок по секциям» содержит данные по общему количеству заявок, поданных на участие в каждой заявленной секции, проводимой в рамках конференции.

На «круге» диаграммы количество заявок отображено в процентах, а при наведении на отдельную секцию выводится еще и точное число поданных заявок.

Список секций выведен в легенду. Данная диаграмма представлена на рисунке 20.

Диаграмма "Распределение поданных заявок по секциям"

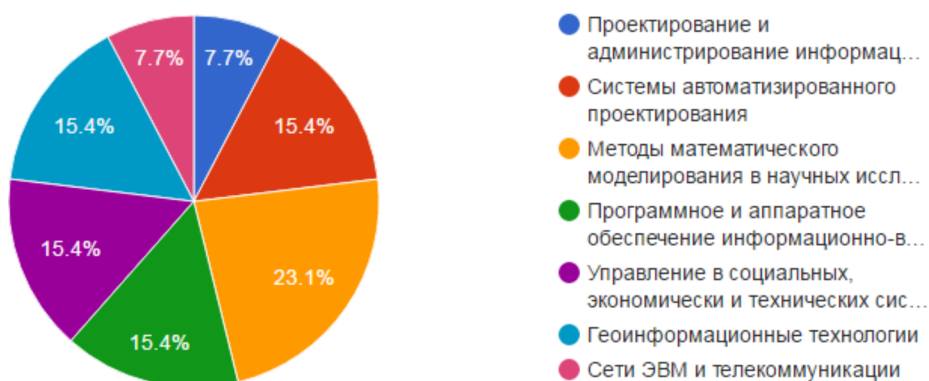


Рисунок 20 – Диаграмма «Распределение поданных заявок по секциям»

Диаграмма «Количество поданных заявок по дням» содержит данные по общему количеству поданных заявок за каждый день приёма заявок на участие в конференции.

Дни, в которое не было подано ни одной заявки на диаграмме не отображаются. Количество заявок отражено на вертикальной оси диаграммы, а даты на горизонтальной оси.

Данная диаграмма представлена на рисунке 21.

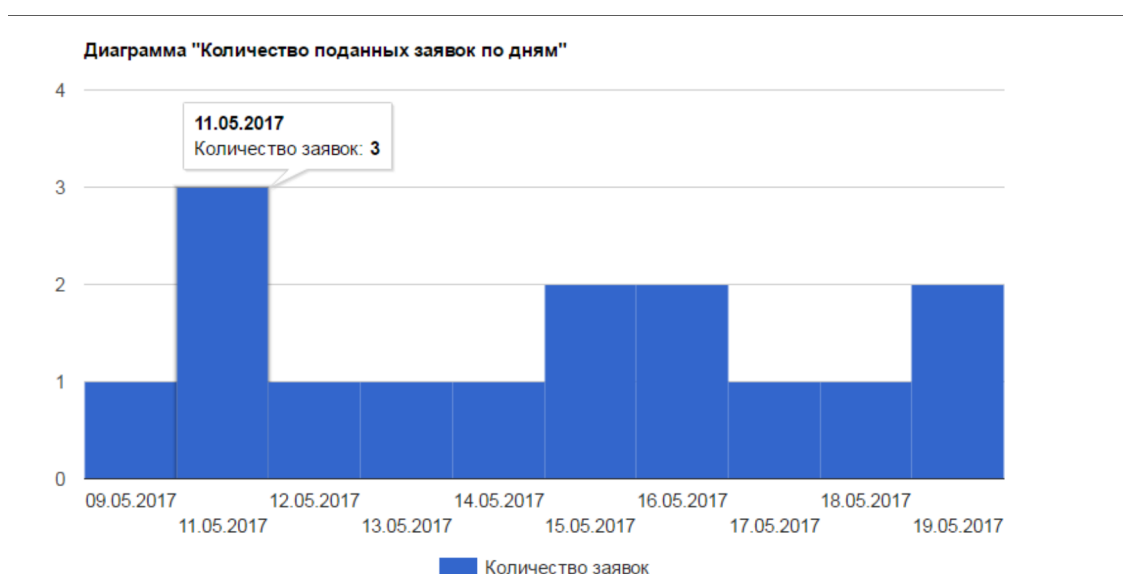


Рисунок 21 – Диаграмма «Количество поданных заявок по дням»

Круговая диаграмма «Количество поданных заявок по факультетам» содержит данные по общему количеству заявок, поданных от каждого факультета, указанного в заявках участников.

На «круге» диаграммы количество заявок отображено в процентах, а при наведении на отдельную секцию выводится еще и точное число поданных заявок.

Список факультетов, указанных в заявках участников представлен в легенде.

Данная диаграмма представлена на рисунке 22.

Диаграмма "Количество поданных заявок по факультетам"

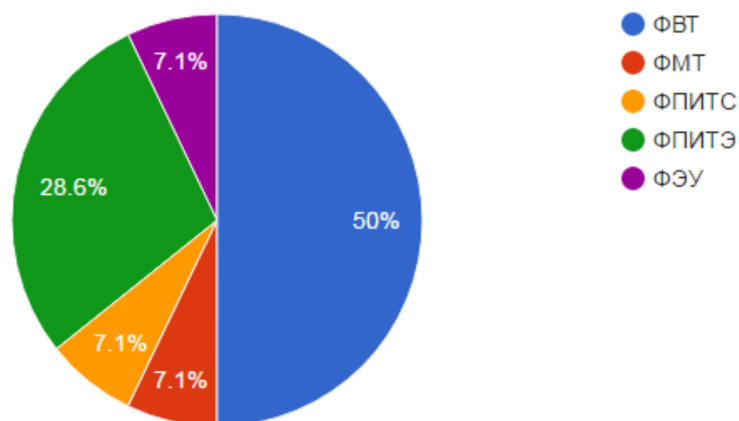


Рисунок 22 – Диаграмма «Количество поданных заявок по факультетам»

Входные данные для формирования графиков имеют формат JSON (JavaScript Object Notation) - простой формат обмена данными, удобный для чтения и написания как человеком, так и компьютером, основанный на подмножестве языка программирования JavaScript [13].

Пример одного из скриптов, в которых сформированы входные данные для отображаемых в системе диаграмм, представлен ниже. Диаграмма "Количество поданных заявок по дням" – «data_chart2.php»:

```
<?
// include ('../connection_db.php');
// подготавливаемый запрос
if (!($stmt = $mysqli-> prepare("SELECT COUNT(name) AS
count,SUBSTR(date_reg, 1, 10) date_reg FROM partisip GROUP BY
(SUBSTR(date_reg, 1, 10))")))
{
    echo "Не удалось подготовить запрос: (" . $mysqli->errno
. ") " . $mysqli->error;
}
if (!$stmt->execute()) {
```

```

        echo "Не удалось выполнить запрос: (" . $stmt->errno . ") "
    . $stmt->error;
    }
    if (!($res = $stmt->get_result())) {
        echo "Не удалось получить результат: (" . $stmt->errno . ") " .
    $stmt->error;}

```

```

    $data = '{"cols": [{ "id":"","label":"Дата","pattern":"","type":"string"},
        { "id":"","label":"Количество
заявок","pattern":"","type":"number"} ],"rows": [;

```

```

        while ($row = $res->fetch_assoc())
        { $count = $row["count"];
            $data .= '{"c":[{"v":'.$row["date_reg"].'}, {"v":'.$count.'}],';
        }
    $data=rtrim($data, ',');
    $data .= ']}';

```

```

    echo ($data);
?>

```

По нажатию на кнопку «Открыть отчет» формируется пример отчета по итогам проведенной конференции.

Отчетные данные сформированы специальными запросами к базе данных, а при копировании текста отчета в стороннюю программу по работе с текстовыми файлами, например, MS Word, разметка данного документа сохраняет свою структуру.

Скрипт формирования отчета «report_text.php» представлен в приложении Г.

Страница со сформированным отчетом представлена на рисунке 23.

ОТЧЕТ

о профориентационном мероприятии

ХII Международной научно-технической конференции НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И СИСТЕМЫ
"НИТИС-2017"

Организатор: Пензенский государственный университет

Содействие в организации оказывали:

Цель мероприятия:

- развитие теоретических и прикладных исследований в области информационных технологий и систем, установления научных и деловых связей между учеными, специалистами и бизнесменами разных стран;
- информирование студентов о новейших результатах НИР предприятия ОПК, возможность магистрантам доложить результаты своей научной работы.

Основание для проведения: приказ Ректора ПГУ

Время проведения: 21.01.2017-20.05.2017

Количество участников(очных): 14

Было проведено пленарное заседание и 7 секционных заседаний(я):

- Секция 1. "Проектирование и администрирование информационных систем".
- Секция 2. "Системы автоматизированного проектирования".
- Секция 3. "Методы математического моделирования в научных исследованиях".
- Секция 4. "Программное и аппаратное обеспечение информационно-вычислительных систем".
- Секция 5. "Управление в социальных, экономических и технических системах".
- Секция 6. "Геоинформационные технологии".
- Секция 7. "Сети ЭВМ и телекоммуникации".

Было заслушано 9 доклада(ов). По результатам работы конференции был издан сборник статей объемом __ п.л., индексируемый в РИНЦ. В работе конференции принимали активное участие студенты и магистранты.

В рамках конференции был проведен конкурс студенческих докладов, победители были награждены грамотами и призами. Оплата затрат на проведение конференции была выполнена из внебюджетных средств.

Дата _____

Подпись _____

Рисунок 23 – Отчет на странице в системе

Представленные в отчете данные: название конференции, время проведения, количество участников, список секционных заседаний, а также количество заслушанных докладов формируются посредством различных запросов к таблицам «partisip», «analitic», «section», «collaborator» базы данных «confbase».

Страница документа «MS Word» со скопированным в неё сформированным в системе отчетом представлена на рисунке 24.

					ВКР.090301.09.17.81	Лист
						45
Изм.	Лист	№ докум.	Подпись	Дата		

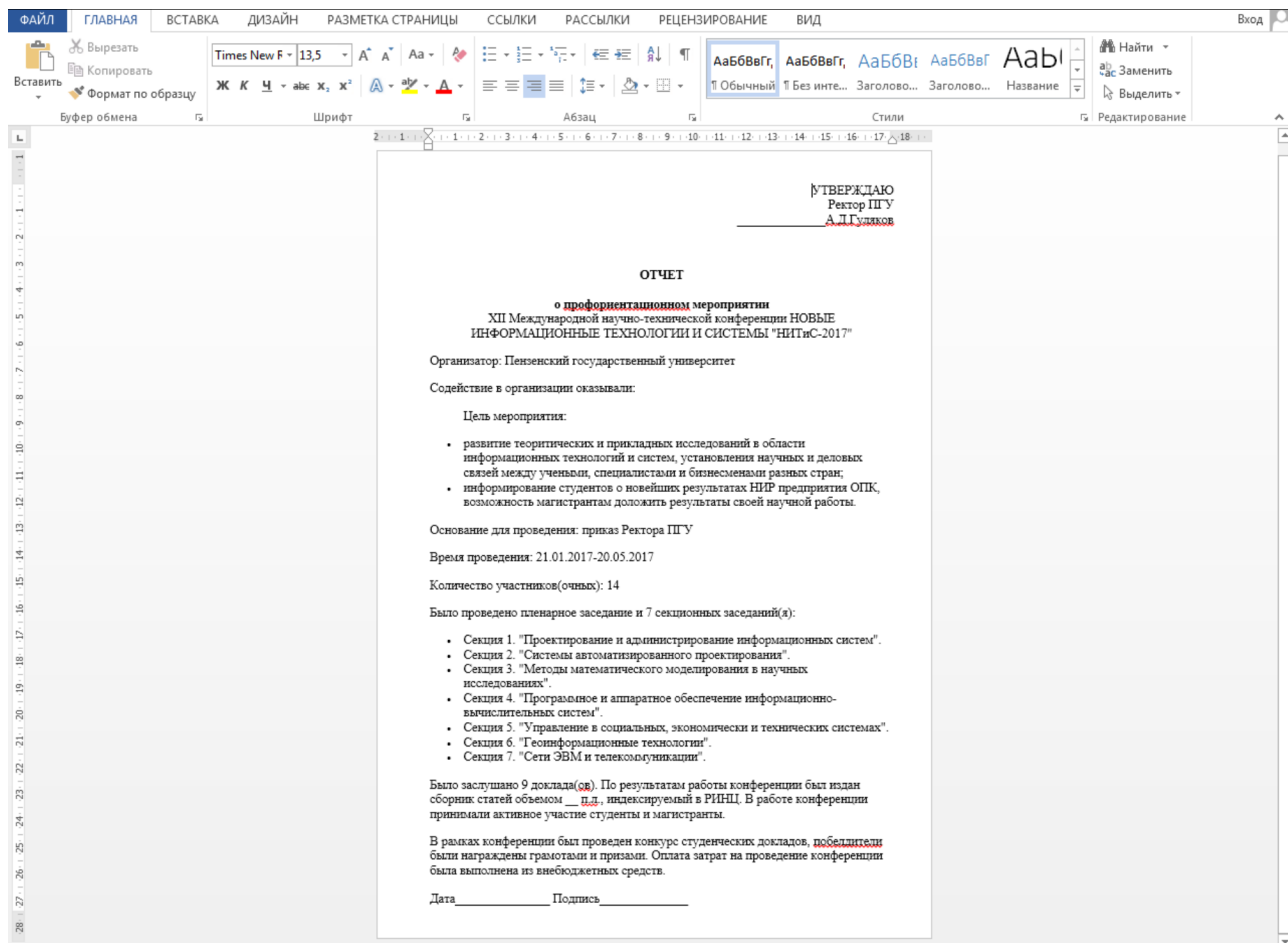


Рисунок 24 – Отчет в «MS Word»

Для удобства последующей печати отчета, рекомендуется привести его к такому виду, чтобы текст занимал только одну страницу документа. Для этого необходимо изменить значения отступов от края листа страницы, что может сделать любой пользователь с начальным уровнем навыков владения ПК.

Для удобства работы с системой, аутентификация пользователя сохраняется и при закрытии браузера, что снимает необходимость повторного ввода логина и пароля администратора при каждом последующем запуске браузера для работы с системой.

При нажатии на кнопку «Выйти» в правом верхнем углу происходит очищение и удаление задействованных в реализации аутентификации глобальных переменных. Пользователь перенаправляется на страницу аутентификации в системе.

5. ТЕСТИРОВАНИЕ СИСТЕМЫ

В ходе проверки правильного функционирования разработанной в рамках данной ВКР системы поддержки проведения конференции были протестированы все основные заявленные и реализованные в программе функции, рассмотренные в разделе 4 – «РЕАЛИЗАЦИЯ СИСТЕМЫ».

Тестирование проводилось на персональном компьютере с операционной системой Windows 7 в браузере Google Chrome (версия 58.0.3029.110).

Результаты проведенного тестирования совпали с ожидаемыми, значит, система работает корректно и выполняет реализованные функции.

В качестве примера приведем процесс тестирования функции подачи заявки на участие в конференции, так как она является одной из основных в разработанной системе.

В открывшейся для подачи заявки на участие в конференции странице заполняем форму с данными (фамилия, отчество, факультет, соавторы, адрес электронной почты), в соответствии с подписями-подсказками, выбираем секцию из выпадающего списка и загружаем пример файла с научной работой с одноимённым названием в формате .docx.

Для проверки правильной работы функции по отслеживанию неправильно заполненных полей формы намеренно оставляем пустым поле «имя», допускаем ошибку в адресе электронной почты и не ставим галочку согласия на обработку персональных данных.

Заполненная таким образом форма представлена на рисунке 25.

Введите фамилию*

Иванов

Введите имя*

Введите отчество

Иванович

Укажите факультет(например, ФВТ)*

ФВТ

☒ Указать соавторов

Введите ФИО соавторов работы(например, Иванов И.И.)

Петров О.Г.

Примечание: если соавторов несколько, не оставляйте пустых строк между ними.

Выберите секцию*

"Геоинформационные технологии"

Загрузить файл(docx/doc/odt)*

Выберите файл | Научная работа.docx

Укажите email(адрес электронной почты)*

y.kurdenkov@gmail.com

☐ Я согласен(на) на обработку персональных данных*

ОК

* - поля, обязательные для заполнения

Рисунок 25 – Некорректно заполненная форма

После нажатия на кнопку отправки заявки, на экране ожидаемо появляется сообщение «Необходимо ввести имя!» сигнализирующее об ошибке в заполнении поля «введите имя*» формы. Так как данное поле обозначено символом «*», что означает его обязательное заполнение, а оно было оставлено пустым.

Вид страницы с примером такого сообщения представлен на рисунке 26.

Необходимо ввести имя!

OK

Введите отчество

Иванович

Укажите факультет(например, ФВТ)*

ФВТ

☒ Указать соавторов

Введите ФИО соавторов работы(например, Иванов И.И.)

Петров О.Г.

Примечание: если соавторов несколько, не оставляйте пустых строк между ними.

Выберите секцию*

"Геоинформационные технологии"

Загрузить файл(docx/doc/odt)*

Выберите файл Научная работа.docx

Укажите email(адрес электронной почты)*

y.kurdenkov@gmail.com

☐ Я согласен(на) на обработку персональных данных*

OK

* - поля, обязательные для заполнения

Рисунок 26 – Сообщение об ошибке в заполнении

После последовательного исправления данных в некорректно заполненных полях на правильные по «указанию» соответствующих сообщений об ошибках, получаем форму, значения в полях которых полностью соответствуют рекомендованным.

По нажатию на кнопку отправки заявки получаем ожидаемое сообщение об успешном принятии данных, что продемонстрировано на рисунке 27.

Данные приняты! Пожалуйста, подождите несколько секунд, чтобы загрузился файл!

ОК

Введите отчество

Иванович

Укажите факультет(например, ФВТ)*

ФВТ

☒ Указать соавторов

Введите ФИО соавторов работы(например, Иванов И.И.)

Петров О.Г.

Примечание: если соавторов несколько, не оставляйте пустых строк между ними.

Выберите секцию*

"Геоинформационные технологии"

Загрузить файл(docx/doc/odt)*

Выберите файл

Научная работа.docx

Укажите email(адрес электронной почты)*

y.kurdenkova@gmail.com

☒ Я согласен(на) на обработку персональных данных*

ОК

* - поля, обязательные для заполнения

Рисунок 27 – Корректно заполненная форма

После этого страница с формой перезагружается и очищается, а все данные из полей заявки отправляются на сервер, где записываются в соответствующие поля таблиц «partisip», «section», «collaborators», «documentc» базы данных.

Убедиться в правильности работы функции можно открыв административную часть системы, так как указанные ранее данные в виде «заявки» отображаются в последней строке списка заявок на странице «Список заявок» системы, что видно из рисунка 28.

ЗАКЛЮЧЕНИЕ

В рамках выполнения задания на выпускную квалификационную работу была спроектирована и реализована система поддержки проведения конференции, отвечающая основным требованиям, выявленным на этапе анализа предметной области и выполняющая заявленные для реализации функции, такие как: формализованный сбор персональных данных, хранение данных с возможностью дальнейшего использования, редактирования и удаления, автоматизация почтовых рассылок, формирование отчета по итогам проведения конференции и демонстрацию диаграмм на основе данных из базы данных. Так, работа выполнена в полном объеме в соответствии с ТЗ.

В рамках работы по сопровождению системы возможны следующие доработки и улучшения:

- учет данных по нескольким конференциям одновременно;
- возможность непосредственно из системы осуществлять публикации в электронный сборник работ, утверждённых по итогам проведения конференции;
- расширение функционала по управлению структурой информационной страницы конференции и формы заявки на участие.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1) Бурукина И.П. Использование инструментов интернет - маркетинга при организации конференций/ Курденкова Я.Г., Кириллов А.И.// Университетское образование (МКУО-2016): сб. статей – Пенза, 2016. – С. 90-92.
- 2) Business Software Products: программы для e-mail рассылок и директ маркетинга [Электронный ресурс]. URL: <http://www.bspdev.ru/> (дата обращения: 13.02.2017).
- 3) MailChimp: Send better email. Sell more stuff [Электронный ресурс]. URL: <https://mailchimp.com/> (дата обращения: 13.02.2017).
- 4) UniSender: хороший сервис массовых Email и СМС рассылок [Электронный ресурс]. URL: <https://www.unisender.com/ru/> (дата обращения: 13.02.2017).
- 5) Сибирский федеральный университет: о системе YConfs [Электронный ресурс]. URL: <http://yconfs.sfu-kras.ru/> (дата обращения: 13.02.2017).
- 6) Бурукина И.П. Возможности платформы HTML5/ Курденкова Я.Г.// Университетское образование (МКУО-2016): сб. статей – Пенза, 2015. – С. 214-216
- 7) Javascript.ru: Современный учебник Javascript [Электронный ресурс]. URL: <https://learn.javascript.ru/> (дата обращения: 02.04.2017)
- 8) Руководство по PHP [Электронный ресурс]. URL: <https://secure.php.net/manual/ru/> (дата обращения: 20.03.2017)
- 9) Л. Веллинг, Разработка веб-приложений с помощью PHP и MySQL, 4-е издание /Л. Томсон. – Изд-во Вильямс: 2010.
- 10) Denwer: Денвер - локальный сервер [Электронный ресурс]. URL: <http://www.denwer.ru/> (дата обращения: 20.03.2017).

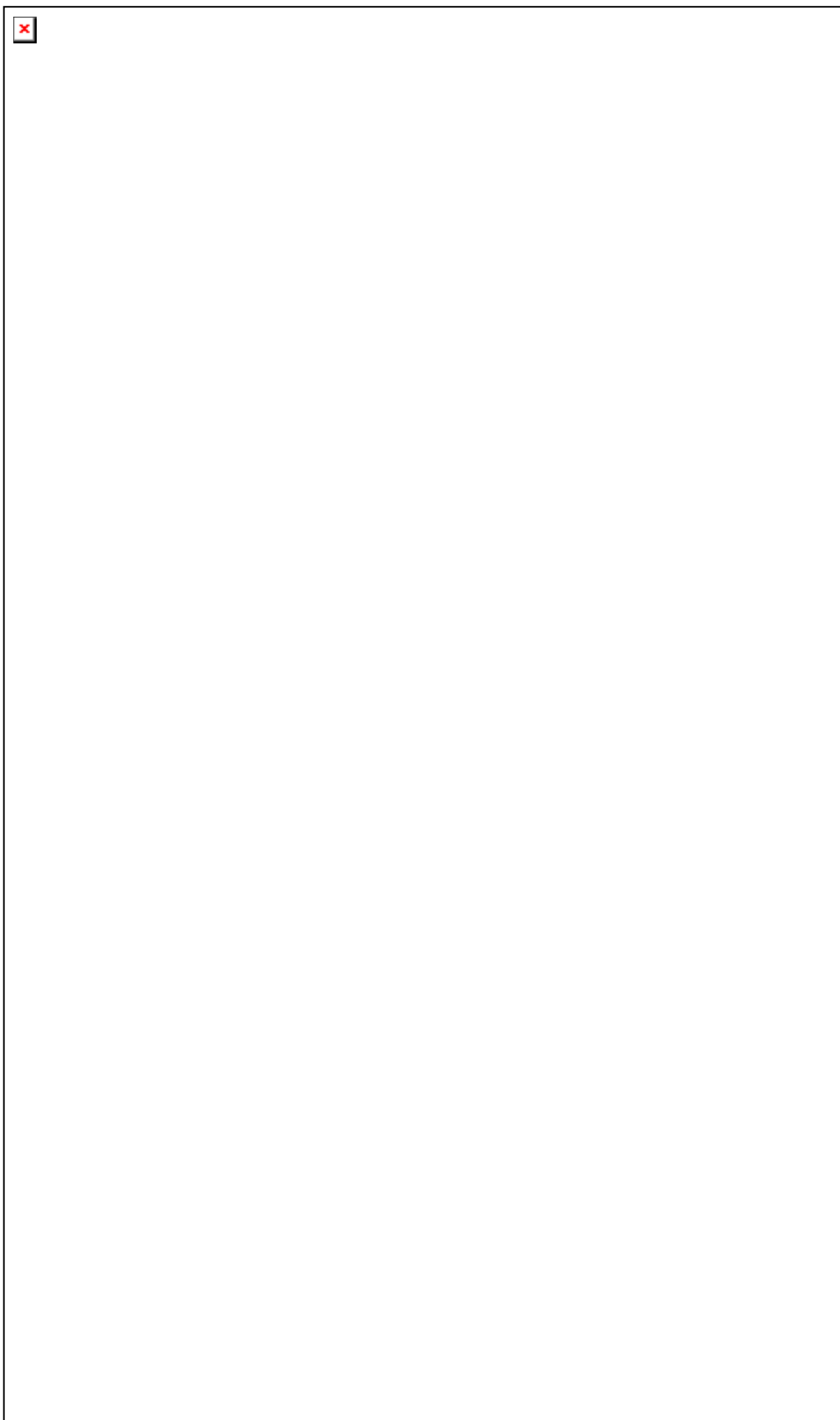
- 11) Федеральный закон от 27.07.2006 N 152-ФЗ (ред. от 22.02.2017) "О персональных данных" // "Российская газета". - N 165. - 29.07.2006.
- 12) Google Charts: Interactive charts for browsers and mobile devices. [Электронный ресурс]. URL: <https://developers.google.com/chart/> (дата обращения: 02.05.2017)
- 13) Введение в JSON [Электронный ресурс]. URL: <http://www.json.org/json-ru.html> (дата обращения: 02.05.2017)

					<i>ВКР.090301.09.17.81</i>	Лист
Изм.	Лист	№ докум.	Подпись	Дата		54

ПРИЛОЖЕНИЕ А
«Диаграмма вариантов использования»



ПРИЛОЖЕНИЕ Б
«Схема работы системы»



ПРИЛОЖЕНИЕ В
«Схема компонентов системы»



ПРИЛОЖЕНИЕ Г

«Исходный код»

style.css

/*часть системы со страницей заявки*/

*{

margin:0;

padding:0;

}

body{

font-family:arial;

}

.header{

height:150px;

border-bottom:2px solid;

}

.wrapper{

min-height: 100vh;

height: 1px;

}

/*для страницы в статусе «закрыта»*/

.closegeg

{

width:70%;

min-height:700px;

margin-left:300px;

margin-left:300px;

border-left:2px solid;

padding-bottom:30px;

text-align:center;

}

.main_main-content{

width:70%;

margin-left:300px;

border-left:2px solid;

padding-bottom:30px;

text-align:center;

}

.label{

					ВКР.090301.09.17.81	Лист
						58
Изм.	Лист	№ докум.	Подпись	Дата		

```

        text-align:center;
        padding-top:20px;

    }
    /*форма заявки*/
    .form{
        padding-top:40px;
        text-align:center;
        margin: 0 auto;
        width:400px;
    }

    .input_field{
    }
    /*кнопка отправки заявки*/
    .main_ok
    {
        background:RGBA(173, 255, 47,0.5);
    }
    .main_ok:hover
    {
        background:RGBA(100, 200, 47,0.5);
    }
    .field-inner{
        display:block;
    }
    .input_label{
        margin-top:30px;
        display:inline-block;
        margin-right:10px;
    }
    .input_field-inner{

        display:block;
        font-size:14pt;
        width:400px;

    }
    .options
    {
        font-size:12px;
    }

```

```

.footer
{
    height:200px;
    border-top:1px solid;
}
/* административная часть системы */
*{
    margin:0;
    padding:0;
}
body{
    font-family:arial;
    background:RGBA(240, 248, 255,1) ;

}
a{
    text-decoration:none;
}
a:visited
{
    color:black;
}
h1 {

    text-align:center;
    padding-bottom:20px;
}
.fon{
    background-image: url('/img/fon.jpg');
}

/*форма авторизации*/
.auth{
    margin:0 auto;
    margin-top:100px;
    width:600px;
    height:400px;
    box-shadow: 0 0 10px rgba(0,0,0,0.5);
    position:relative;
    background:RGBA(47, 79, 79,0.2);

}
.ahello{

```

```

}
.alabel{
    font-size:24pt;
    margin-top:15px;
    margin-bottom:5px;
}

/*кнопки*/
.abutton{
    border:none;
    width:72px;
    height:72px;
    margin-top:10px;
    font-size:13pt;
    background:RGBA(173, 255, 47,0.5);
}

/*отчеты*/
.rep_b
{
    background: url('../img/report.png')no-repeat;
    width:64px;
    height:64px;
    border:none;
    vertical-align:middle;
}

/*рассылки*/
.b_mail
{
    background: url('../img/email.png')no-repeat;
}

/*секции*/
.b_section
{
    background: url('../img/section.png')no-repeat;
}

/*список заявок*/
.b_partisip
{
    background: url('../img/partisip.png')no-repeat;
}

```

```

/*другие кнопки*/
.b_otchet
{
    background: url('../img/otchet.png')no-repeat;
}
.b_open
{
    background: url('../img/open.png')no-repeat;
}
.b_close
{
    background: url('../img/close.png')no-repeat;
}
.b_form
{
    display:inline-block;
}
.search{
}
.search_result{
    list-style-type: none;
    box-shadow: 0 0 10px rgba(10, 60, 50,0.5);
    color:RGB(10, 60, 50);
}
.search_result li:hover{
    background:RGBA(47, 79, 79,0.1);
    cursor:pointer;
}
.sear{
}

.authfields{
    position:absolute;
    top:10%;
    left:16%;
    display:block;
}

```

```

        text-align:center;
    }
    .b_redact
    {
        vertical-align:middle;
        width:48px;
        height:48px;
        background: url('../img/redact.png')no-repeat;
    }
    .b_ok
    {
        background: url('../img/ok.png')no-repeat;
    }
    .b_neok{
        background: url('../img/neok.png')no-repeat;
    }
    .b_del
    {
        background: url('../img/delete.png')no-repeat;
    }
    .b_gomail
    {
        background: url('../img/gomail.png')no-repeat;
    }
    .main-content{

        width:80%;

    }

    /*ЛЕВОЕ МЕНЮ*/
    .menu{
        text-align:center;
        width:15%;
        float:left;
        height: 100vh;
        //border-right: 1px solid;
        margin-right:2px;
        display:inline-block;

        //box-shadow: 0 0 10px rgba(0,0,0,0.5);
    }
    .content-inner{

```

```

width:100%;
text-align:center;

}
.text_to_send
{
    height:200px;
    width:200px;
}
.ne_ok
{
    width:22px;
    height:22px;
    border:none;
    text-align:center;
    vertical-align:middle;
    background: url('../img/vopros.png')no-repeat;
}
.ok{
    width:22px;
    height:22px;
    border:none;
    text-align:center;
    vertical-align:middle;
    background: url('../img/button_ok.png')no-repeat;
}
.ne_ok , .ok, .b_gomail, .b_del, .b_mail, .b_section, .b_partisip, .b_otchet, .b_redact, .b_open,
.b_close:hover{
    cursor:pointer;
}
.come{
    background:RGBA(173, 255, 47,0.5);
    border-left:1px solid;
}
.comes
{
    width:22px;
    height:22px;
    border:none;
    outline:none;
    text-align:center;
    vertical-align:middle;
    background: url('../img/button_ok.png')no-repeat;
}

```

```

}

/*утверждённая заявка*/
.vuuu{
    border:3px solid;
    width:20px;
    height:20px;
    background:RGBA(173, 255, 47,0.4);
}
td{
    font-size:14px;
    padding-right:10px;
    border-bottom: 1px solid;
}
.table-header
{
    font-weight:bold;
}

/*видимая табличная часть*/
.visible-content{
    height:70vh;
    overflow:scroll;
    background:#ffffff;
    box-shadow: 0 2px 10px rgba(0,0,0,0.5);
}

.update-button{
    display:inline-block;
}
.delete-button
{
    display:inline-block;
}

}
.partisipants{
    min-height:100vh;
    border-left: 1px solid;
    display:inline-block;
}
.sections{
    min-height:100vh;
    border-left: 1px solid;

```

```

        display:inline-block;
    }
    .chart
    {
    }
    /*кнопка выхода*/
    .exit{
        width:48px;
        height:48px;
        position:absolute;
        right:10px;
        border:none;
        top:5px;
        font-size:20pt;
        background: url('../img/goout.png')no-repeat;
    }

```

index.php

```

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
<script src="//ajax.googleapis.com/ajax/libs/jquery/3.1.0/jquery.min.js"></script>
<link rel="stylesheet" type="text/css" href="css/style.css">
<title>Главная</title>
<script type="text/javascript">
function testStr(str)//проверка на допустимые символы
{
var pattern = new RegExp(/^[A-яA-zЁё\.\- ]/);

if (pattern.test(str)) {
return false;}
else{
return true;}
}
function testStrC(str)
{
var pattern = new RegExp(/^[A-яЁё\.\- ]/);

if (pattern.test(str)) {
return false;}
else{

```

```

return true;}
}
function check(){//проверка на корректность введенных в поля данных
//отдельные элементы структуры документа
var sname=document.registration.sname.value;
var name=document.registration.name.value;
var pname=document.registration.pname.value;
var sect=document.registration.sect.value;
var email_adress=document.registration.email_adress.value;
var chek_personal = document.registration.chek_personal;
var files = document.registration.files.value;
var parts_place=document.registration.parts_place.value;
var collaborat = document.getElementsByName('collaborat');

var check_ok=true;//флаг на отсутствие ошибок

if (sname!="")//проверка фамилии
{
if (sname.length<2 || sname.length>40)
{alert('Укажите ваше фамилию (2-40 символов)!');
document.registration.sname.focus();
check_ok=false;
return false;
}
if(!testStr(sname))
{alert('Необходимо ввести корректную фамилию!');return false;}
}
else
{alert('Необходимо ввести фамилию!');
document.registration.sname.focus();
check_ok=false;
return false;
}

//проверка имени
if (name!="")
{if (name.length<2 || name.length>40)
{alert('Укажите ваше имя (2-30 символов)!');
document.registration.name.focus();
check_ok=false;
return false;
}
if(!testStr(name))

```

```

{alert('Необходимо ввести корректное имя !');return false;}
}
else
{alert('Необходимо ввести имя!');
document.registration.name.focus();
check_ok=false;
return false;
}

//проверка отчества
if (pname!="")
{if (pname.length<2 || pname.length>40)
{alert('Укажите ваше отчество (2-30 символов)!');
document.registration.pname.focus();
check_ok=false;
return false;
}
if(!testStr(pname))
{alert('Необходимо ввести корректное отчество !');return false;}
}

//проверка соавторов
for(i=0;i<collaborat.length;i++){
if (collaborat[i].value!="")
{
{if (collaborat[i].length<2 || collaborat[i].length>40)
{alert('Укажите ваше ffgfgfg (2-30 символов)!');
document.registration.collaborat[i].focus();
check_ok=false;
return false;
}
if(!testStrC(collaborat[i].value))
{alert('Необходимо ввести корректное ФИО соавторов !');return false;}
}
}
}

//проверка факультета
if (parts_place!="")
{if (parts_place.length<2 || parts_place.length>40)
{alert('Укажите ваш факультет (2-40 символов)!');
document.registration.sname.focus();
check_ok=false;
return false;
}
}

```

					ВКР.090301.09.17.81	Лист
						69
Изм.	Лист	№ докум.	Подпись	Дата		

```

}
else{alert('Вы не дали согласие на обработку персональных данных!');
document.registration.chek_personal.focus();
check_ok=false;
return false;}
//если ошибок не обнаружено
if(check_ok===true)
{
alert('Данные приятны! Пожалуйста, подождите несколько секунд, чтобы загрузился файл!');
}
}
</script>
<script type="text/javascript">      //развертка списка соавторов по клику
function showOrHide(cb, cat) {
cb = document.getElementById(cb);
cat = document.getElementById(cat);
if (cb.checked)//если был активирован чекбокс
{ cat.style.display = "block";}
else
{cat.style.display = "none";}
}

</script>
</head>
<body>
<div class="header">
</div>
<div class="wrapper ">
<div class="main_main-content">
<?

include ('connection_db.php');//соединение с БД
//запрос к базе с орг. данными
$query = "SELECT is_open, conf_name, conf_date, conf_date_open FROM analitic";
$result = $mysqli->query($query);
$row = $result->fetch_assoc();
if($row['is_open']==2)//если форма заявки закрыта
{
header('Location:cluserreg.php');//перенаправить на страницу «приём заявок окончен»
}
echo("<h3 class='label'>Заявка на участие в ".$row['conf_name']. "</h3><br><h4>Сроки
проведения: ".$row['conf_date']. "</h4>
<br><h4>Приём заявок до: ".$row['conf_date_open']. "</h4>");

```

```

?>
<div class="form" align="center">
//форма ввода данных с полями
<form name="registration" action="registration.php" enctype="multipart/form-data"
method="POST" onsubmit="return check();">
<div class="input_label input_field-inner">
<label class="field-inner">Введите фамилию*</label>
<input class="input_field-inner" type="text" name="sname"/>
</div>
<div class="input_label input_field-inner">
<label class="field-inner">Введите имя*</label>
<input class="input_field-inner" type="text" name="name"/>
</div>
<div class="input_label input_field-inner">
<label class="field-inner">Введите отчество</label>
<input class="input_field-inner" type="text" name="pname"/>
</div>
<div class="input_label input_field-inner">
<label class="field-inner">Укажите факультет(например, ФВТ)*</label>
<input class="input_field-inner" type="text" name="parts_place"/>
</div>
</br>
<div class=" ">
<input type="checkbox" id="1000" onchange="showOrHide(1000, 2000);">Указать соавторов

<p id="2000" style="display: none;">
<label style="font-size:10pt;">Введите ФИО соавторов работы(например, Иванов
И.И)</label></br>
<input type="text" class="collaborat" name="collaborat[]"></br>
<input type="text" class="collaborat" name="collaborat[]"></br>
<input type="text" class="collaborat" name="collaborat[]"></br>
<input type="text" class="collaborat" name="collaborat[]"></br>
<input type="text" class="collaborat" name="collaborat[]"></br>
<input type="text" class="collaborat" name="collaborat[]"></br>
<input type="text" class="collaborat" name="collaborat[]"></br>
<label style="font-size:10pt;">Примечание: если соавторов несколько, не оставляйте пустых
строк между ними.</label>
</p>
</div>
<div class="input_label input_field-inner">
<label class="field-inner">Выберите секцию*</label>
<?php

```

```

include ('connection_db.php');?>
<select class="input_field-inner" name="sect">
//формирование выпадающего списка секций
<?php
$query = "SELECT * FROM section";

if ($result = $mysqli->query($query))
while ($row = $result->fetch_assoc()){
?>
<option class="options" value="<?=$row['ID']?>"> <?=$row['section_name']?> </option>
<?
}
$mysqli->close();
?>
</select></div>
<div class="input_label input_field-inner">
<label class="field-inner">Загрузить файл(docx/doc/odt)*</label>
<input name="MAX_FILE_SIZE" type="hidden" value="100000000">
<input class="input_field-inner" name="files" type="file">
</div>
<div class="input_label input_field-inner">
<label class="field-inner">Укажите email(адрес электронной почты)*</label>
<input class="input_field-inner" type="text" name="email_adress"/>
</div>

<div class="input_label input_field-inner">
<input class=" " type="checkbox" name="chek_personal" value ="1"/>
<label class="">Я согласен(на) на обработку персональных данных*</label>

</div>
<input class="input_label input_field-inner main_ok" type="submit" name="upload" value="OK"/>
</br>
</form>
<span class="label">* - поля, обязательные для заполнения</span></br></br>
</div>
</div>
<div class="footer"></div>
</div>
</body>
</html>

```

registration.php

```
<?php
include ('connection_db.php');//соединение с БД
//секция
$sect = htmlspecialchars($_POST['sect']);
//$referral = htmlspecialchars($_POST['referral']);
//имя
$name = htmlspecialchars($_POST['name']);
$name = htmlspecialchars($_POST['sname']);//фамилия
$name= htmlspecialchars($_POST['pname']);//отчество
//адрес электронной почты
$email = htmlspecialchars($_POST['email_adress']);
$loadtime=date('d.m.Y-H:i:s');//время подачи заявки
$collab = $_POST['collaborat'];//соавторы
//факультет
$parts_place = htmlspecialchars($_POST['parts_place']);
//данные для письма
//тема
    $subject ="=?utf-8?B?". base64_encode("Заявка на участие в конференции"). "?=";
//текст сообщения в который на место переменных вставятся значения
    $message = "<html><body><p><b>Здравствуй!</b></p><br><p>".$sname." ".$name.",
благодарим Вас за подачу заявки на участие в конференции!</p>
        <p>Ваши данные приняты к рассмотрению. Ожидайте информационного
сообщения!</p></body></html>";
    $header="From: Conferention";
        $header.="\\nContent-type: text/html; charset=\\\"utf-8\\\"";
//функция отправки сообщения
    mail($email, $subject, $message, $header);

header('Content-Type: text/html; charset=UTF-8');
//функция транслитерации для названия файла с работой
function rus2translit($string) {
    $converter = array(
        'а' => 'a', 'б' => 'b', 'в' => 'v',
        'г' => 'g', 'д' => 'd', 'е' => 'e',
        'ё' => 'e', 'ж' => 'zh', 'з' => 'z',
        'и' => 'i', 'й' => 'y', 'к' => 'k',
        'л' => 'l', 'м' => 'm', 'н' => 'n',
        'о' => 'o', 'п' => 'p', 'р' => 'r',
        'с' => 's', 'т' => 't', 'у' => 'u',
        'ф' => 'f', 'х' => 'h', 'ц' => 'c',
        'ч' => 'ch', 'ш' => 'sh', 'щ' => 'sch',
        'ь' => 'i', 'ы' => 'y', 'ъ' => 'i',
```

```

'э' => 'e', 'ю' => 'yu', 'я' => 'ya',

'A' => 'A', 'Б' => 'B', 'В' => 'V',
'Г' => 'G', 'Д' => 'D', 'Е' => 'E',
'Ё' => 'E', 'Ж' => 'Zh', 'З' => 'Z',
'И' => 'I', 'Й' => 'Y', 'К' => 'K',
'Л' => 'L', 'М' => 'M', 'Н' => 'N',
'O' => 'O', 'П' => 'P', 'Р' => 'R',
'С' => 'S', 'Т' => 'T', 'У' => 'U',
'Ф' => 'F', 'Х' => 'H', 'Ц' => 'C',
'Ч' => 'Ch', 'Ш' => 'Sh', 'Щ' => 'Sch',
'Ъ' => 'I', 'Ы' => 'Y', 'Ь' => 'I',
'Э' => 'E', 'Ю' => 'Yu', 'Я' => 'Ya',
);
return strstr($string, $converter);
}

//Каталог загрузки файлов
$uploadDir = '/home/labatest11.ru/www/upload/';

//Вывод ошибок
$error = array();

//Коды ошибок загрузки файла
$errorUpload = array(
    0 => 'Ошибок не возникло, файл был успешно
загружен на сервер.',
    1 => 'Размер принятого файла превысил
максимально допустимый размер, который задан директивой upload_max_filesize
конфигурационного файла php.ini.',
    2 => 'Размер загружаемого файла превысил
значение MAX_FILE_SIZE, указанное в HTML-форме.',
    3 => 'Загружаемый файл был получен только
частично.',
    4 => 'Файл не был загружен.',
    6 => 'Отсутствует временная папка. Добавлено в
PHP 4.3.10 и PHP 5.0.3.'
);

//Определяем типы файлов для загрузки
$fileTypes = array(
    //'jpg' => 'image/jpeg',
    //'png' => 'image/png',

```

```

        // 'gif' => 'image/gif',
        'doc'=>'application/vnd.openxmlformats-
officedocument.wordprocessingml.document',
        'docx'=>'application/msword',
        'odt'=>'application/vnd.oasis.opendocument.text'
    );

    //Если нажата кнопка загрузить
    if(isset($_POST['upload']))
    {
        //Проверяем пустые данные или нет
        if(!empty($_FILES))
        {
            //Проверяем на ошибки
            if($_FILES['files']['error'] > 0)
                $err[] = $errUpload[$_FILES['files']['error']];

            //Проверяем тип файла для загрузки
            if(!in_array($_FILES['files']['type'], $fileTypes))
                $err[] = 'Данный тип файла <b>'. $_FILES['files']['type'] .</b> не
подходит для загрузки!';

            $nametmp= rus2translit($_FILES['files']['tmp_name']);
            $namename = rus2translit($_FILES['files']['name']);
            //Если нет ошибок то грузим файл
            if(empty($err))
            {
                //дата изменения
                $datemod= date ("F d Y H:i:s.",
filemtime($_FILES['files']['tmp_name']));
                //размер файла в килобайтах
                $sizefile=(($_FILES['files']['size']/1024);

                //название для вывода в систему
                $pub_n=$_FILES['files']['name'];

                //тип файла
                $type = pathinfo($_FILES['files']['name']);
                $pub_name=rtrim($pub_n, $_FILES['files']['type']);
                $namef = $uploadDir .'/'.$namename;
                move_uploaded_file($_FILES['files']['tmp_name'],$namef);
                $stat = stat($namef);
                $datemod= date ("d.m.Y-H:i:s.", $stat['mtime']);
                $DirforDown ='/'.'upload/'.'$namename.';

            }
        }
    }

```

```

else
    echo implode('<br>', $err);
}

}

//подготавливаемый запрос к БД
if (!$stmt = $mysqli->prepare("INSERT INTO `documents`(`doc_name`,`pub_name`,`doc_place`,`doc_size`,`doc_timemod`)
    VALUES (?,?,?,?,?)")) {
    echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}
//подготовленные параметры
if (!$stmt->bind_param("sssis", $namename,$pub_name,$DirforDown,$sizefile,$datemod)) {
    echo "Не удалось привязать параметры: (" . $stmt->errno . ") " . $stmt->error;
}
//выполнение запроса
if (!$stmt->execute()) {
    echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}

$query = "SELECT ID FROM `documents` WHERE `doc_timemod` = '$datemod'";

if ($result = $mysqli->query($query)){
    $rowdoc = $result->fetch_assoc();
}

//подготавливаемый запрос к БД
if (!$stmt = $mysqli->prepare("INSERT INTO `partisip`(`sname`,`name`,`pname`,`section_ID`,`doc_ID`,`date_reg`,`email`,`is_verif`,`appear`,`parts_place`)
    VALUES (?,?,?,?,?,?,?,?,?)")) {
    echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}
$noverif = false;
$noappear = false;
//подготовленные параметры
if (!$stmt->bind_param("sssiissbbs",
    $sname,$name,$pname,$sect,$rowdoc['ID'],$loadtime,$email,$noverif,$noappear,$parts_place)) {
    echo "Не удалось привязать параметры: (" . $stmt->errno . ") " . $stmt->error;
}
//выполнение запроса

if (!$stmt->execute()) {
    echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}

```

```

}

//запрашиваем из базы последнего добавленного участника, чтобы получить id для соавторов
$query = "SELECT * FROM `partisip` ORDER BY partisip.ID DESC";

        if ($result = $mysqli->query($query)){
            $rowpartsip = $result->fetch_assoc();

        }

if (!$stmt = $mysqli->prepare("INSERT INTO `collaborators`(`Fname`,`par_ID`)VALUES
(?,?)")) {
    echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}

/* подготавливаемый запрос, вторая стадия: привязка и выполнение */
if (!$stmt->bind_param("si", $collaboratp,$rowpartsip['ID'])) {
    echo "Не удалось привязать параметры: (" . $stmt->errno . ") " . $stmt->error;
}
$i=0;
if(strlen($collab[0])==0)//если нет соавторов записываем прочерк в БД
{
    $collab[0]="-";
}
while(strlen($collab[$i])!=0 )
{
    //записываем текущего соавтора в переменную
    $collaboratp=$collab[$i];
    if (!$stmt->execute()) {
        echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
    }

    $i++;
}
$stmt->close();
$mysqli->close();
//перенаправляем на главную страницу
header ('Location: index.php');

?>

```

authclass.php

```
<?
session_start(); //Запускаем сессии
function isAuth() {
if (isset($_COOKIE["id"]))
{
if ($_COOKIE["id"]!="")
{
return true;}}
else{
if (isset($_SESSION["is_auth"])) { //Если сессия существует
return $_SESSION["is_auth"]; //Возвращаем значение переменной сессии is_auth (хранит true
если авторизован, false если не авторизован)
}
else return false; //Пользователь не авторизован, т.к. переменная is_auth не создана
}
}
function auth($login, $password) {
include ('./connection_db.php');
if (!($stmt = $mysqli-> prepare("SELECT * FROM admin"))) {
echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}

if (!$stmt->execute()) {
echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}

if (!($res = $stmt->get_result())) {
echo "Не удалось получить результат: (" . $stmt->errno . ") " . $stmt->error;}
while($row = $res->fetch_assoc())
{
if($row['login']== $login)
{
if (crypt($password, $row['password']) == $row['password'])
{
$_SESSION["is_auth"] = true; //Делаем пользователя авторизованным
$_SESSION["login"] = $login; //Записываем в сессию логин пользователя
setcookie("id",$login, time()+60*60*24*30);

return true;
}
else
{ //Логин и пароль не подошел
```

```

$_SESSION["is_auth"] = false;
return false;
}}}
}
/**
 * Метод возвращает логин авторизованного пользователя
 */

function getLogin() {
if (isAuth()) { //Если пользователь авторизован
if (isset($_SESSION["is_auth"])) {return $_SESSION["login"];}
else{ //Возвращаем логин, который записан в сессию
return $_COOKIE["id"];}
}
}

function out() {
if (isset($_SESSION["is_auth"]))
{$_SESSION = array(); //Очищаем сессию
session_destroy();
} //Уничтожаем

setcookie("id","", time()-999);
}
?>

```

autorisation.php

```

<?php
/*$passwod='a1234z';
$login='myadmin';
$hash = crypt('$passwod');*/

include ('authclass.php');

if(!getLogin()){
    nologin();
}
?>

<form action="" method="get">
<input class="exit" name="is_exit" type="submit" value="">
</form>

<?
function nologin(){

```

```

header("Location: index.php");
}
if (isset($_GET["is_exit"])) { //Если нажата кнопка выхода

    out(); //Выходим
    header("Location: index.php"); //перенаправляем на страницу авторизации после выхода
}

?>

```

«spisok.php»

```

<?php
//порядковый номер участника в таблице
$number=1;
include ('../connection_db.php');
header('Content-Type: text/html; charset=UTF-8');
function get_partisip($par_ID)
{
include ('../connection_db.php');
if (!$stmt = $mysqli-> prepare("SELECT Fname FROM collaborators, partisip
WHERE collaborators.par_ID = partisip.ID AND partisip.ID=? ORDER BY collaborators.ID")) {
echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}
if (!$stmt->bind_param("i", $par_ID)) {
echo "Не удалось привязать параметры: (" . $stmt->errno . ") " . $stmt->error;
}
/* подготавливаемый запрос, вторая стадия: привязка и выполнение */

if (!$stmt->execute()) {
echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}

if (!$res_part = $stmt->get_result()) {
echo "Не удалось получить результат: (" . $stmt->errno . ") " . $stmt->error;}

while($partisip = $res_part->fetch_assoc())
{

echo($partisip['Fname']);
echo("<br>");
}}

```

```

if (!$stmt = $mysqli->prepare("SELECT date_reg, sname, name, pname, appear,
is_verif, parts_place, section_name, doc_name, doc_size, doc_place, doc_timemod, pub_name FROM
partisip,
section, documents WHERE partisip.section_ID = section.ID AND partisip.doc_ID = documents.ID
ORDER BY partisip.ID"))) {
echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}

/* подготавливаемый запрос, вторая стадия: привязка и выполнение */

if (!$stmt->execute()) {
echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}

if (!$res = $stmt->get_result()) {
echo "Не удалось получить результат: (" . $stmt->errno . ") " . $stmt->error;}

$quer = "SELECT ID FROM partisip";
$result = $mysqli->query($quer);

while ($row = $res->fetch_assoc())//пока есть записи
{
$rowid = $result->fetch_assoc();
//формируем строку таблицы
echo "<form></form><tr>
<td><form action='\" method='GET'><input type='hidden' name='appid' value='\".$rowid['ID'].\"\">
<input class='ne_ok' name='appear' type='submit' value='\".$row['appear'].\"\"></form></td>
<td> <input id='\". $number.\" type='checkbox' name ='flagdelete[' value='\". $rowid['ID'].\"\"/> </td>
<td>\". $number. \"</td><td width='10%'>\". $row['date_reg']. \"</td>
<td>\". $row['sname'].\"<br>\".$row['name'].\"<br>\".$row['pname'].\"</td>
<td > \". $row['parts_place'].\"</td>
<td width='20%'> \". $row['section_name'].\"</td>
<td> <a href='\".$row['doc_place'].\"\">
.$row['pub_name'].\"</a><br>\".$row['doc_size'].\"КБ<br>\".$row['doc_timemod'].\"</td>
<td width='10%'>\", get_partisip($rowid['ID']),\"</td>
<td class='come'><input class='is_come' id='\". $row['is_verif'].\" type='checkbox' name ='is_come'
value='\". $rowid['ID'].\"\"/></td>
</tr>";
$number++;
//номер участника увеличить на единицу для следующего
}

```

```
$stmt->close();
$mysqli->close();
```

```
?>
```

report_text.php

```
//пример отчета по итогам конференции
```

```
<html style="padding:0;margin:0;"><body style="margin:0 200px;">
<head>
```

```
</head>
```

```
<body>
```

```
<div align="right">
```

```
<p align="">УТВЕРЖДАЮ</br> Ректор ПГУ</br>
```

```
_____ А.Д.Гуляков</p></div></br></br>
```

```
<div align="center"><b>ОТЧЕТ</br></br>о профориентационном мероприятии</b></br>
<?
```

```
include ('../connection_db.php');
```

```
$query = "SELECT * FROM analitic";
```

```
$result= $mysqli->query($query);
```

```
$nameconf = $result->fetch_assoc();
```

```
echo("<span align='center'>".$nameconf['conf_name']."</span>");
```

```
?>
```

```
</div>
```

```
<div align="" style="height:600px">
```

```
<p>Организатор: Пензенский государственный университет</p>
```

```
<p>Содействие в организации оказывали:</p>
```

```
<ul>Цель мероприятия:
```

```
<li>
```

развитие теоритических и прикладных исследований в области информационных технологий и систем, установления научных и деловых связей между учеными, специалистами и бизнесменами разных стран;

```
</li>
```

```
<li>
```

информирование студентов о новейших результатах НИР предприятия ОПК, возможность магистрантам доложить результаты своей научной работы.

```
</li>
```

```
</ul>
```

```
<p>Основание для проведения: приказ Ректора ПГУ</p>
```

```
<p>Время проведения:
```

```
<?
```

```
$query = "SELECT * FROM analitic";
```

					ВКР.090301.09.17.81	Лист
Изм.	Лист	№ докум.	Подпись	Дата		82

```

$result= $mysqli->query($query);
$nameconf = $result->fetch_assoc();
echo("<span align='center'>".$nameconf['conf_date']."</span>");
?>
</p>
<p>
Количество участников(очных):
<?
if (!$stmt = $mysqli->prepare("SELECT * FROM partisip WHERE partisip.is_verif=1 ")) {
    echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}
if (!$stmt->execute()) {
    echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}

$res = $stmt->get_result();
$row_cnt = $res->num_rows;

if (!$stmt = $mysqli->prepare("SELECT * FROM collaborators, partisip WHERE
partisip.is_verif=1 AND par_ID = partisip.ID ")) {
    echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}
if (!$stmt->execute()) {
    echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}

$rescol = $stmt->get_result();
while ($rescolcou = $rescol->fetch_assoc())
{
    if ($rescolcou['Fname']!= '-')
    {
        $row_cnt++;
    }
}
echo("<span align='center'>".$row_cnt."</span>");
?>
</p>

<p>
Было проведено пленарное заседание и
<?
if (!$stmt = $mysqli->prepare("SELECT * FROM section")) {
    echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}

```

```

}
if (!$stmt->execute()) {
    echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}

$res = $stmt->get_result();
$row_cnt = ($res->num_rows)-1;
$rows = $res->fetch_assoc();
echo("<span align='center'>".$row_cnt." секционных заседаний(я):</span>");
$number=1;
echo("<ul>");
while ($rows = $res->fetch_assoc())
{
    echo("
    <li>Секция ".$number.". ".$rows['section_name']. "</li>");
    $number++;
}
echo("</ul>");
?>

</p>
<p>
Было заслушано
<?
if (!$stmt = $mysqli->prepare("SELECT * FROM partisip WHERE partisip.is_verif=1 ")) {
    echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}
if (!$stmt->execute()) {
    echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}

$res = $stmt->get_result();
$row_cnt = $res->num_rows;
echo("<span align='center'>".$row_cnt."</span>");
?>
доклада(ов). По результатам работы конференции был издан сборник статей объемом __
п.л.,
индексируемый в РИНЦ. В работе конференции принимали активное участие студенты и
магистранты.
</p>
<p>

```

В рамках конференции был проведен конкурс студенческих докладов, победители были награждены грамотами и призами. Оплата затрат на проведение конференции была выполнена из внебюджетных средств.

</p>

</div>

Дата_____Подпись_____

</body>

<html>

mail.php

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<link rel="stylesheet" type="text/css" href="../css/style.css">

<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>

<script src="//ajax.googleapis.com/ajax/libs/jquery/3.1.0/jquery.min.js"></script>

<title>Управление почтовыми рассылками</title>

</head>

<body>

<?php

include ('authorisation.php');//проверка на авторизацию

include ('left_menu.php');//подключение левого меню

?>

<div class="main-content partisipants">

<h1 class="hello">Управление почтовыми рассылками</h1>

<p>Здесь Вы можете отправить утвержденным участникам конференции информационные письма с организационной информацией.</p></br>

<?>

//если форма заявки открыта

if (\$row[is_open]==1)

{echo("<p>Кнопка рассылки станет доступна только после закрытия формы заявки на участие.</p>");}

else

{echo "<button class='b_redact b_gomail' ></button> - разослать письма с местом, датой и временем проведения его секционного заседания.</br>";

echo "</br><p>Вы можете отправить участникам конференции письма с благодарностью за участие.</p>";

					ВКР.090301.09.17.81	Лист
Изм.	Лист	№ докум.	Подпись	Дата		85

```
echo "<a href='finalymail.php'><button class='b_redact b_gomail' ></button> - разослать письма
с благодарностью участникам, присутствовавшим на секционных заседаниях.</a>";}
```

```
if (isset($_GET["is_ok"])) { //Если нажата кнопка разослать
include ('../connection_db.php');
$qwe="UPDATE `analytic` SET `email_count`=' '";
$mysqli->query($qwe);
```

```
header("Location: mail.php");
}
?>
```

```
<?include ('../connection_db.php');
$quer = "SELECT * FROM analytic";
$result = $mysqli->query($quer);
$row = $result->fetch_assoc();
```

```
echo "<div class='mailcount'> </br> Отправлено писем: ".$row['email_count']."</div>";
```

```
?><form action="" method="get">
<input class="" name="is_ok" type="submit" value="Очистить данные">
</form>
<div>
<br><p><b>Начните набирать фамилию участника, чтобы отправить ему личное
письмо.</b></p></br>
<form action="" method="post">
<input type="text" name="referral" placeholder="Начните набирать" value="" class="input_field-
inner sear" autocomplete="off">
<ul class="search_result"></ul>
<input type="hidden" class="id_part" name="id_partis" value="0">
<p>Введите информационный текст</p></br>
<textarea class="text_to_send" name="text_to_send"></textarea> </br>
<input type="submit" name="send" value="Отправить письмо">
</form>
</div>
<script>
$(function(){

// поиск участника
$('.sear').bind("change keyup input click", function() {

if(this.value.length >= 1){
$.ajax({
type: 'post',
```

```

url: "search.php", //Путь к обработчику
data: {'referral':this.value},
response: 'text',
success: function(data){
$(".search_result").html(data).fadeIn(); //Выводим полученные данные в списке
})
}
else{
$(".search_result").fadeOut();
})

$(".search_result").hover(function(){
$(".sear").blur(); //Убираем фокус с input
})

//При выборе результата поиска, прячем список и заносим выбранный результат в input
$(".search_result").on("click", "li", function(){
s_user = $(this).text();
id_user = $(this).attr("id");

$(".sear").val(s_user).attr('disabled'); //деактивируем input
$(".search_result").fadeOut();
$('.id_part').val(id_user);
})))
</script>
</div>
<? if (isset($_POST["send"]))//если нажата кнопка отправить
{
include ('./connection_db.php');//подключение к БД
$to_id = $_POST['id_partis'];//ID участника
$text_to_send = $_POST['text_to_send']; //текст сообщения

if (!$stmt = $mysqli-> prepare("SELECT name, pname, email FROM partisip WHERE ID=?"))
{//запрос к БД
echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}

if (!$stmt->bind_param("i",$to_id)) {
echo "Не удалось привязать параметры: (" . $stmt->errno . ") " . $stmt->error;
}
if (!$stmt->execute()) {
echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}
}

```

```

if (!$res = $stmt->get_result()) {
echo "Не удалось получить результат: (" . $stmt->errno . ") " . $stmt->error;}

$formail = $res->fetch_assoc();
$email = $formail['email'];
$subject = "?utf-8?B?". base64_encode("Конференция. Вопросы по заявке"). "?=";
$message = "<html><body><p><b>Здравствуйте!</b></p><br><p>".$formail['name'].
".$formail['pname'].",</p><p>".$text_to_send."</p><p>С Уважением, организатор.
</p></body></html>";
$header="From: Conferention";
$header.="<br>Content-type: text/html; charset=?utf-8?";

mail($email, $subject, $message, $header);
}
?>
</body>
</html>

```

search.php

```

<?php
include('../connection_db.php');//подключение к БД

if(!empty($_POST["referral"])){ //Принимаем данные

    $referral = $_POST["referral");//записываем данные
    //запрос к таблице с участниками БД
    $db_referral = $mysqli -> query("SELECT * FROM partisip WHERE partisip.sname LIKE
    '%$referral%'");
    //запрос к таблице с документами БД
    $db_referral_doc = $mysqli -> query("SELECT * FROM partisip ,documents WHERE
    partisip.sname LIKE '%$referral%' AND documents.ID = partisip.doc_ID ORDER BY
    partisip.ID");
    while ($row = $db_referral -> fetch_assoc()) {
        $rowd = $db_referral_doc -> fetch_assoc();
        //формируем данные на вывод
        echo "<n<li id='".$row['ID']."'>".$row['sname']. " ".$row['name']. " ".$row['pname']. " -
        ".$rowd['pub_name']. "</li>"; // $row["name"] - имя поля таблицы
    }?>

```

mail_org.php

```
<? include ('../connection_db.php');
//запрос к таблице с орг.информацией
$query = "SELECT * FROM analitic";
if (!$stmt = $mysqli->prepare("UPDATE `analitic` SET `email_count`= ?")) {
echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}
/* подготавливаемый запрос, вторая стадия: привязка и выполнение */

if (!$stmt->bind_param("i", $count)) {
echo "Не удалось привязать параметры: (" . $stmt->errno . ") " . $stmt->error;
}
if ($result= $mysqli->query($query))
{
$formail_count = $result->fetch_assoc();
}
$count=$formail_count['email_count'];

//запрос к таблице с участниками
if (!$stmtmp = $mysqli-> prepare("SELECT  sname,name,pname, email, section_name,
section_place, section_time, section_date FROM partisip,
section WHERE partisip.section_ID = section.ID AND partisip.appear = 1 ORDER BY
partisip.ID")) {
echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}
/* подготавливаемый запрос, вторая стадия: привязка и выполнение */
if (!$stmtmp->execute()) {
echo "Не удалось выполнить запрос: (" . $stmtmp->errno . ") " . $stmtmp->error;
}
if (!$res = $stmtmp->get_result()) {
echo "Не удалось получить результат: (" . $stmtmp->errno . ") " . $stmtmp->error;}

while($formail = $res->fetch_assoc())
{
//данные для отправки в сообщении
$name = $formail['sname'];//фамилия участника
$name = $formail['name'];//имя
$name = $formail['pname'];
$section_name = $formail['section_name'];//наименование секции
//дата проведения секционного заседания
$date = $formail['section_date'];
$time = $formail['section_time'];//время
```

```

$place = $formail['section_place'];//место проведения
$email = $formail['email'];
$subject = "=?utf-8?B?". base64_encode("Приглашение на конференцию"). "=?";
$message = "<html><body><p><b>Здравствуй!</b></p></br><p>".$sname." ".$name."
".$sname.", Ваша заявка на участие в конференции рассмотрена и утверждена.</p>
<p>Ваша секция: ".$section_name." </p><p>Ждём Вас ".$date." в ".$time." по адресу: ".$place."
</p><p>С Уважением, организатор. </p></body></html>";
$header="From: Conferention";
$header.="<nContent-type: text/html; charset=\"utf-8\"";

mail($email, $subject, $message, $header);
$count++;
$stmt->execute();
}
header("Location: mail.php");?>

```

finalymail.php

```

<? include ('../connection_db.php');
//запрос к таблице с орг. информацией
$query = "SELECT * FROM analitic";
if (!$stmt = $mysqli->prepare("UPDATE `analitic` SET `email_count`=?")) {
echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}

/* подготавливаемый запрос, вторая стадия: привязка и выполнение */

if (!$stmt->bind_param("i", $count)) {
echo "Не удалось привязать параметры: (" . $stmt->errno . ") " . $stmt->error;
}
if ($result= $mysqli->query($query))
{
$formail_count = $result->fetch_assoc();
}
$count=$formail_count['email_count'];

//запрос к таблице с участниками
if (!$stmt = $mysqli-> prepare("SELECT  sname,name,pname, email FROM partisip WHERE
partisip.appear = 1 AND partisip.is_verif = 1 ORDER BY partisip.ID")) {
echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}

/* подготавливаемый запрос, вторая стадия: привязка и выполнение */

```

```

if (!$stmt->execute()) {
echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}

if (!$res = $stmt->get_result()) {
echo "Не удалось получить результат: (" . $stmt->errno . ") " . $stmt->error;}

while($formail = $res->fetch_assoc())
{
$name = $formail['sname'];
$name = $formail['name'];
$pname = $formail['pname'];
$email = $formail['email'];
$subject ="=?utf-8?B?". base64_encode("Благодарность за участие в конференции"). "?=";
$message = "<html><body><p><b>Здравствуйте!</b></p><br><p>".$sname." ".$name."
".$pname.", благодарим Вас за участие в конференции.</p>
<p>Ожидайте публикации работы и выпуска сборника статей. </p><p>Будем рады видеть
Вас на последующих мероприятиях. </p><p>С Уважением, организатор.
</p></body></html>";
$header="From: Conferention";
$header.="<br>Content-type: text/html; charset='utf-8'";

mail($email, $subject, $message, $header);
$count++;
$stmt->execute();
}
header("Location: mail.php");

?>

```

Фрагменты скрипта partisipant.php

```

...
<?
    if (isset($_GET["appear"])) //если нажата кнопка утвердить
    { include ('../connection_db.php');//подключение к БД
    $idd=$_GET["appid"];
    //запрос к БД на получение записи с нужным ID
    if (!$stmt = $mysqli->prepare("SELECT appear FROM partisip WHERE ID=?")) {
        echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
    }
}

```

```

/* подготавливаемый запрос, вторая стадия: привязка и выполнение */

if (!$stmt->bind_param("i",$idd)) {
    echo "Не удалось привязать параметры: (" . $stmt->errno . ") " . $stmt->error;
}

if (!$stmt->execute()) {
    echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}
$res = $stmt->get_result();
$row = $res->fetch_assoc();
if ($row['appear']==0){
    $tr=1;
}
else
{$tr=0;}
if (!$stmt = $mysqli->prepare("UPDATE partisip SET appear= ? WHERE ID = ?")) {
    echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}

/* подготавливаемый запрос, вторая стадия: привязка и выполнение */

if (!$stmt->bind_param("ii", $tr,$idd)) {
    echo "Не удалось привязать параметры: (" . $stmt->errno . ") " . $stmt->error;
}

if (!$stmt->execute()) {
    echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}

    header("Location: index.php");
}

...
<script>//постановка отметки о присутствии
$( '.is_come').change(function(){

// Проверка стоит галочка или нет
if ($(this).is(':checked')) {

    var chk = $(this).attr("name");
    var chkVal = $(this).attr("value");

```

```

// запрос к БД
$.ajax({
    url: 'come.php',
    type: 'post',
    data: { 'is_come':this.value},
    //beforeSend: function() { alert(chk+ ' '+chkVal); },
    success: function(data) { alert('Принято!');window.location.reload()
        }
    });
}
});
</script>

```

```

</div></br><a href="upload.php"> <button class=" b_redact"></button> -
редактировать все</a>
</div></div></div>
</body>
</html>

```

come.php

```

<?
include ('../connection_db.php');
$upload= $_POST['is_come'];
if (!$stmt = $mysqli->prepare("UPDATE `partisip` SET `is_verif` = 1 WHERE `ID` = ?")) {
echo "Не удалось подготовить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}
/* подготавливаемый запрос, вторая стадия: привязка и выполнение */
if (!$stmt->bind_param("i", $upload)) {
echo "Не удалось привязать параметры: (" . $stmt->errno . ") " . $stmt->error;
}
if (!$stmt->execute()) {
echo "Не удалось выполнить запрос: (" . $stmt->errno . ") " . $stmt->error;
}

?>

```