

Заведующему кафедрой
фундаментальной информатики
А. Г. Смольянову
студента 4 курса
очной формы обучения
(на бесплатной основе)
направления подготовки
02.03.02 – Фундаментальная информатика
и информационные технологии
факультета математики и
информационных технологий
ФГБОУ ВО «МГУ им. Н.П. Огарёва»
Панфилкина Артема Юрьевича

заявление

Прошу разместить мою выпускную квалификационную работу на тему
«Разработка Web-сервиса учета сведений об иностранных студентах высшего
учебного заведения» в электронной библиотечной системе университета в
полном объеме.

09.06.2020г

Дата

Панфилкин А.Ю

Подпись

Заявление о самостоятельном характере выполнения работы

Я, Панфилин Артём Юрьевич, обучающийся 4 курса направления подготовки 02.03.02 Фундаментальная информатика и информационные технологии, заявляю, что в моей работе на тему «Разработка Web-сервиса учета сведений об иностранных студентах высшего учебного заведения», представленной в Государственную экзаменационную комиссию для публичной защиты, не содержится элементов неправомерных заимствований.

Все прямые заимствования из печатных и электронных источников, а также ранее защищенных письменных работ, кандидатских и докторских диссертций имеют соответствующие ссылки.

Я ознакомлен с действующим в Университете Положением о проверке работ студентов, обучающихся в ФГБОУ ВО «МГУ им. Н.П. Огарёва», на наличие заимствований, в соответствии с которым обнаружение неправомерных заимствований является основанием для отрицательного отзыва руководителя работы.

Панфилин Артём Юрьевич

Подпись обучающегося

09.06.2020

Дата

Работа представлена для проверки в Системе «Антиплагиат.ВУЗ»

09.06.2020 / Каменков Р.В.

Дата представления работы

подпись руководителя

Декану
факультета математики
и информационных технологий
Чучасву Ивану Ивановичу
студента 402 группы
направления подготовки 02.03.02
Фундаментальная информатика и ИТ
Панфилкина Артёма Юрьевича

заявление

Я, Панфилкин Артём Юрьевич, ознакомлен с пунктом 1.4 Положения о проведении государственной итоговой аттестации по образовательным программам высшего образования — программам бакалавриата, программам специалитета и программам магистратуры в ФГБОУ ВО «МГУ им. Н. П. Огарёва» о запрете во время проведения государственной итоговой аттестации иметь при себе и использовать иные средства связи, кроме средств, установленных для обеспечения качественной видеоконференции.

Дата 16.06.2020г.

Подпись 

СПРАВКА

о результатах проверки текстового документа
на наличие заимствований

Проверка выполнена в системе
Антиплагиат.ВУЗ

Автор работы	УВАРОВ ГЛЕБ СЕРГЕЕВИЧ
Факультет, кафедра, номер группы	ФМиИТ, ОП, 402
Тип работы	Выпускная квалификационная работа
Название работы	Разработка веб-сервиса потокового вещания на основе программной платформы Node
Название файла	ВКР - Акты - УваровГС - ФМИИТ - 2020.docx
Процент заимствования	10,87%
Процент цитирования	0,33%
Процент оригинальности	88,78%
Дата проверки	16:47:54 05 июня 2020г.
Модули поиска	Коллекция Фунда; Модуль поиска обязательных выражений; Коллекция Патенты; Модуль поиска перефразирований Интернет; Модуль поиска преобразования eLIBRARY.RU; Модуль поиска "МГУ им. Н. П. Огарева"; Коллекция Медицина; Модуль поиска Интернет; Коллекция Гарант; Коллекция eLIBRARY.RU; Модуль поиска переводных заимствований по интернет (Elnet); Модуль поиска переводных заимствований по eLibrary (Elnk); Периодные заимствования; Цитирование; Коллекция РГБ; Сводная коллекция ЭБС; Модуль выделения библиографических записей; Модуль поиска ИЭС "Антиплагиат"
Работу проверил	СМОЛЯНОВ АНДРЕЙ ГРИГОРЬЕВИЧ ИДП преподаватель
Дата подписи	09.06.2020г.
	 Подпись преподавателя

Чтобы убедиться
в подлинности справки,
используйте QR-код, который
сопровождает справку на сайте.



Если на вопрос, является ли обнаруженное заимствование
намеренным, система отвечает на усмотрение проверяющего.
Предоставленная информация не подлежит использованию
в коммерческих целях.

О Т З Ы В

на бакалаврскую работу студента 4 курса факультета математики и информационных технологий МГУ им. Н. П. Огарева

Панфилкина А.Ю. на тему:

Разработка Web-сервиса учета сведений об иностранных студентах высшего учебного заведения

Цель данной работы – разработка веб-сервиса учета сведений об иностранных студентах в высшем учебном заведении, работающего в любом браузере без предварительной установки, имеющего возможность добавления, редактирования и хранения данных в базе данных, а также автоматического формирования документов в форматах DOCX и XLS, которые требуются от студентов по части миграционного учёта. Также сервис должен осуществлять формирование различных отчётов.

Работа состоит из введения, трех разделов, заключения, списка использованных источников и 4 приложений.

В первой части бакалаврской работы рассматриваются особенности исследуемой области, количественный рост обучающихся из других стран по данным из открытых источников.

Вторая часть работы посвящена описанию возможностей используемых программных средств, а именно – языку программирования PHP, его библиотекам для работы с документами, языку SQL, СУБД MySQL и дополнительным инструментам.

Третья часть посвящена реализации веб-сервиса учёта сведений об иностранных студентах и созданию базы данных для работы с ней. В работе показаны основные моменты создания, а также разработки сервиса, описаны его возможности и функциональность. Результаты разработки и реализации представлены на скриншотах.

Работа над дипломным проектом показала, что Панфилкин А.Ю. умеет системно мыслить, может четко сформулировать задачи для достижения поставленной цели и успешно их решать в поставленные сроки.

Настоящая работа удовлетворяет всем требованиям, предъявляемым к бакалаврским работам, а ее автор, Панфилкин А.Ю., заслуживает оценки «отлично».

Руководитель работы

к.ф.-м.н., доцент



В.В.Каменев


ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
МОРДОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМ. Н. П. ОГАРЁВА»

Факультет математики и информационных технологий

Кафедра фундаментальной информатики

УТВЕРЖДАЮ

Зав. кафедрой
канд. физ.-мат. наук, доц.

 А. Г. Смольянов
(подпись)

«18» июня 2020 г.

БАКАЛАВРСКАЯ РАБОТА

**РАЗРАБОТКА WEB-СЕРВИСА УЧЕТА СВЕДЕНИЙ ОБ
ИНОСТРАННЫХ СТУДЕНТАХ ВЫСШЕГО УЧЕБНОГО ЗАВЕДЕНИЯ**

Автор бакалаврской работы  09.06.2020 А. Ю. Панфилин

Обозначение бакалаврской работы БР-02069964-02.03.02-14-20

Направление 02.03.02 Фундаментальная информатика и информационные
технологии

Руководитель работы

канд. физ.-мат. наук, доц.


(подпись)

09.06.2020 В. В. Каменев

Нормоконтролер

канд. техн. наук, доц.


(подпись)

11.06.2020 С.В. Гарина

Саранск
2020

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
МОРДОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМ. Н. П. ОГАРЁВА»

Факультет математики и информационных технологий

Кафедра фундаментальной информатики

УТВЕРЖДАЮ

Зав. кафедрой
канд. физ.-мат. наук, доц.

 А. Г. Смольянов
(подпись)

«25» декабря 2019 г.

ЗАДАНИЕ НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ

(в форме бакалаврской работы)

Студент Панфилкин Артем Юрьевич

1 Тема Разработка Web-сервиса учета сведений об иностранных студентах
высшего учебного заведения

Утверждена приказом № 10037-с от 25.12.2019 г.

2 Срок представления работы к защите _____

3 Исходные данные для научного исследования - литература и электронные
ресурсы по теме исследования

4 Содержание выпускной квалификационной работы

4.1 Проблематика области

4.2 Используемый инструментарий

4.3 Реализация

5 Приложение А (обязательное) Фрагмент файла «main.js»

6 Приложение Б (обязательное) Фрагмент файла «toWord.php»

7 Приложение В (обязательное) Фрагмент файла «toExcel.php»

8 Приложение Г (обязательное) Фрагмент файла «getStudents.php»

Руководитель работы
канд. физ.-мат. наук, доц.



25.12.19

подпись, дата

В. В. Каменев

Задание принял к исполнению



25.12.19

подпись, дата

А. Ю. Панфилин

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
МОРДОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМ. Н. П. ОГАРЁВА»

Факультет математики и информационных технологий
Кафедра фундаментальной информатики

УТВЕРЖДАЮ

Зав. кафедрой
канд. физ.-мат. наук, доц.

_____ А. Г. Смольянов
(подпись)

«18» июня 2020 г.

БАКАЛАВРСКАЯ РАБОТА

РАЗРАБОТКА WEB-СЕРВИСА УЧЕТА СВЕДЕНИЙ ОБ ИНОСТРАННЫХ СТУДЕНТАХ ВЫСШЕГО УЧЕБНОГО ЗАВЕДЕНИЯ

Автор бакалаврской работы (подпись) 09.06.2020 А. Ю. Панфилкин

Обозначение бакалаврской работы БР–02069964–02.03.02–14–20

Направление 02.03.02 Фундаментальная информатика и информационные
технологии

Руководитель работы

канд. физ.-мат. наук, доц. (подпись) 09.06.2020 В. В. Каменев

Нормоконтролер

канд. техн. наук, доц. (подпись) 11.06.2020 С.В. Гарина

Саранск
2020

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
МОРДОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМ. Н. П. ОГАРЁВА»

Факультет математики и информационных технологий
Кафедра фундаментальной информатики

УТВЕРЖДАЮ

Зав. кафедрой
канд. физ.-мат. наук, доц.

_____ А. Г. Смольянов
(подпись)

«25» декабря 2019 г.

ЗАДАНИЕ НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ

(в форме бакалаврской работы)

Студент Панфилкин Артем Юрьевич

1 Тема Разработка Web-сервиса учета сведений об иностранных студентах
высшего учебного заведения

Утверждена приказом № 10037-с от 25.12.2019 г.

2 Срок представления работы к защите _____

3 Исходные данные для научного исследования - литература и электронные
ресурсы по теме исследования

4 Содержание выпускной квалификационной работы

4.1 Проблематика области

4.2 Используемый инструментарий

4.3 Реализация

5 Приложение А (обязательное) Фрагмент файла «main.js»

6 Приложение Б (обязательное) Фрагмент файла «toWord.php»

7 Приложение В (обязательное) Фрагмент файла «toExcel.php»

8 Приложение Г (обязательное) Фрагмент файла «getStudents.php»

Руководитель работы

канд. физ.-мат. наук, доц.

_____ В. В. Каменев
подпись, дата

Задание принял к исполнению

_____ А. Ю. Панфилин
подпись, дата

РЕФЕРАТ

Бакалаврская работа содержит 57 страниц, 19 рисунков, 17 использованных источников, 4 приложения.

ВЕБ-СЕРВИС, ИНОСТРАННЫЕ СТУДЕНТЫ, ДОКУМЕНТЫ, PHPWORD, PHPSPREADSHEET, БАЗА ДАННЫХ.

Объектом исследования является сбор, хранение и организация данных об иностранных студентах.

Цель исследования – изучение технологий формирования документов в различных форматах, технологий создания баз данных и разработка веб-сервиса для учёта сведений об иностранных студентах.

Информационной базой работы послужила научная и учебно-методическая литература, сайты Интернет.

В результате работы изучены возможности библиотек PHPWord и PhpSpreadsheet для работы с документами, изучены основы работы с базами данных, разработан веб-сервис учёта сведений об иностранных студентах..

Степень внедрения – частичная.

Эффективность – имеет прикладное значение.

Область применения – добавление, хранение и редактирование данных об иностранных студентах, формирование различных документов.

По итогам работы созданный сервис был протестирован, его функциональность проверена.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	6
1 Проблематика области	8
2 Используемый инструментарий	12
2.1 Язык PHP	12
2.2 SQL и MySQL	14
2.3 PhpSpreadsheet	17
2.4 PHPWord	20
2.5 Дополнительные инструменты	23
3 Реализация	26
3.1 База данных	26
3.2 Программная реализация	31
ЗАКЛЮЧЕНИЕ	43
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	44
ПРИЛОЖЕНИЕ А (обязательное) Фрагмент файла «main.js»	46
ПРИЛОЖЕНИЕ Б (обязательное) Фрагмент файла «toWord.php»	52
ПРИЛОЖЕНИЕ В (обязательное) Фрагмент файла «toExcel.php»	55
ПРИЛОЖЕНИЕ Г (обязательное) Фрагмент файла «getStudents.php»	58

ВВЕДЕНИЕ

Выпускная квалификационная работа посвящена разработке веб-сервиса для учёта сведений об иностранных студентах и формировании соответствующих документов.

В настоящее время можно увидеть тенденцию роста количества иностранных студентов, обучающихся в высших учебных заведениях Российской Федерации. Развитие технологий обучения и квалификации учебного персонала приводит к большой популярности обучения граждан других стран. Работа с такими студентами и их документами в основном ведется без специализированных информационных систем в некоторых подразделениях ВУЗов, что приводит к более медленной обработке данных.

Актуальность данной работы состоит в том, что в работе учебных заведений очень важно обеспечивать постоянный учёт данных об иностранных студентах с помощью автоматизированных систем.

Целью выпускной работы является разработка веб-сервиса учета сведений об иностранных студентах в высшем учебном заведении, работающего в любом браузере без предварительной установки, имеющего возможность добавления, редактирования и хранения данных в базе данных, а также автоматического формирования документов в форматах DOCX и XLS, которые требуются от студентов по части миграционного учёта. Также сервис должен осуществлять формирование различных отчётов.

Для достижения цели необходимо решить следующие задачи:

1. Изучить существующие библиотеки для формирования документов в форматах DOCX, XLS и ознакомиться с их основными классами и методами.
2. Изучить средства работы с базами данных.
3. Создать программное средство для учёта сведений об иностранных студентах на языке PHP.
4. Протестировать созданное программное средство.

Работа состоит из введения, трех разделов, заключения, списка использованных источников и 4 приложений.

В первой части бакалаврской работы рассматриваются особенности исследуемой области, количественный рост обучающихся из других стран по данным из открытых источников.

Вторая часть работы посвящена описанию возможностей используемых программных средств, а именно – языку программирования PHP, его библиотекам для работы с документами, языку SQL, СУБД MySQL и дополнительным инструментам.

Третья часть посвящена реализации веб-сервиса учёта сведений об иностранных студентах и созданию базы данных для работы с ней. В работе показаны основные моменты создания, а также разработки сервиса, описаны его возможности и функциональность. Результаты разработки и реализации представлены на скриншотах.

1 Проблематика области

В наши дни численность иностранных студентов в России продолжает постоянно расти. Только в 2019 году количество заявок на обучение от иностранных граждан-абитуриентов выросло на 20%. Каждый год в пределах квоты, предоставляемой Правительством Российской Федерации, министерство образования России обеспечивает 15 тысячам иностранных студентов приём на бесплатное обучение.

В федеральном проекте «Экспорт образования» указано, что уже к 2024 году в России должно проходить обучение 435 тысяч иностранных граждан. С большим количеством высших учебных заведений заключены контракты, согласно которым в них должен эффективно обеспечиваться рост численности иностранных студентов. В настоящее время по данным, предоставленным Институтом статистики ЮНЕСКО, Россия находится на шестом месте в мире по привлекательности для студентов из других государств.

Если обратить внимание на данные цифровой платформы «Russia.Study», на которой существует возможность сравнивать и выбирать учебные заведения, а также подавать заявки на обучение, то можно увидеть, что почти 115 тысяч иностранных абитуриентов проявили желание получать высшее образование в ВУЗах России. Конечно, данные на разных источниках могут отличаться.

По недавно опубликованным данным погранслужбы ФСБ об иностранцах, которые приехали в Россию на учебу с начала 2019 года, численность таких студентов составляет около 181,5 тысяч человек. Несмотря на разное число студентов, рост численности иностранных абитуриентов из года в год является достаточно высоким. На рисунке 1.1 показана численность иностранных граждан, обучавшихся в высших учебных заведениях России с соответствующим процентным соотношением.

Иностранные студенты в России

Численность иностранных граждан, обучавшихся в вузах России, тыс. человек

% — доля иностранных граждан в составе всех студентов российских вузов

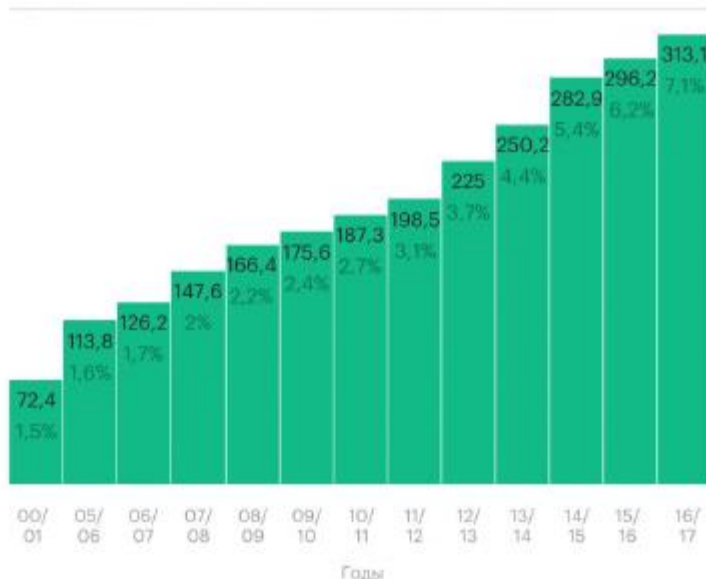


Рисунок 1.1 – Рост численности иностранных студентов в России

В качестве направленности образования, наблюдается рост интереса среди иностранных абитуриентов к техническому образованию, предоставляемому в России [2]. Помимо этого, множество студентов выбирают направления в сфере лечебного дела, стоматологии, экономики, менеджмента, а также международных отношений.

По данным министерства науки и высшего образования, в настоящее время существует более семидесяти международных соглашений о взаимном признании между Российской Федерацией и другими странами. В течение последних 5 лет были заключены соглашения с такими странами, как Франция, Италия, Сербия, Кипр, Мальта, Венгрия, Узбекистан, Камбоджа и Ангола.

На рисунке 1.2 можно ознакомиться с тем, какие уровни образования выбирают иностранные студенты в их процентном соотношении. Можем увидеть, что большую часть составляют обучение по программам бакалавриата и магистратуры.



Рисунок 1.2 – Уровни образования иностранных студентов в России

Всё вышесказанное обуславливает то, что в настоящее время существует необходимость в сохранении информации об иностранных студентах, а также в автоматическом формировании различных связанных документов, ручное заполнение которых замедляет работу отделов высших учебных заведений. Работа без использования специализированных сервисов может вызывать затруднения в оперативном получении необходимых данных, различных отчётов и прочего.

Постоянное заполнение однотипных данных – очень длительный процесс, требующий работы нескольких работников. Также существует

высокая вероятность человеческой ошибки при многократном повторном заполнении документов такого плана.

Польза разрабатываемого веб-сервиса заключается в следующих факторах:

1. Программное средство способствует быстрому формированию документов, отчётности и сортировке информации по отдельным критериям (факультет, направление, год поступления и т. п.).

2. Документация формируется в различных форматах, наиболее популярных в использовании, позволяет просматривать и редактировать. Документы доступны на скачивание сразу из веб-сервиса.

3. Сервис обеспечивает оперативный и удобный доступ к данным иностранных студентов. Позволяет хранить все данные в базе, имеет возможности их быстрого редактирования и добавления.

2 Используемый инструментарий

2.1 Язык PHP

Язык программирования PHP (PHP: Hypertext Preprocessor, с английского – «PHP: препроцессор гипертекста») является одним из наиболее распространённых и популярных языков среди web-разработчиков. Представляя собой удобный и гибкий инструмент, PHP был создан для генерирования HTML-страниц и их дальнейшего внедрения на web-серверы [3]. PHP – язык программирования, который может быть встроен непосредственно в код HTML-страниц, которые затем будут обработаны интерпретатором PHP [13]. Также данный язык позволяет работать с базами данных. В PHP поддерживается возможность работы не только с локальными, но и с удалёнными файлами. К достоинствам этого языка относят возможность его поддержки множеством хостингов и web-серверов. PHP имеет большое количество понятной и доступной документации на русском языке, он постоянно обновляется и развивается в соответствии с запросами настоящего времени.

Написанные на языке PHP скрипты выполняются на сервере, а сгенерированные ответы, в свою очередь, посылаются клиенту. PHP обладает большим количеством возможностей, имеет множество различных функциональных библиотек, которые можно использовать в решении разноплановых задач. Этот язык программирования доступен для изучения, имеет невысокий порог входа, что позволяет начинающим разработчикам быстро осваивать его возможности. Однако, несмотря на это, PHP использует множество опытных разработчиков и крупных компаний, так как он способен удовлетворить и сложным профессиональным запросам.

В настоящее время язык программирования PHP является одним из лидеров в области разработки web-приложений, где активно применяется множеством компаний. Его выбор часто обусловлен тем, что он может

взаимодействовать с множеством различных систем управления базами данных (СУБД). К таким можно отнести SQLite, MySQL, PostgreSQL, MySQLi, PDO и многие другие популярные системы. Также так как дистрибутив языка PHP является доступным для подавляющего количества различных операционных систем, например, Linux, Microsoft Windows, MacOS и т. п., его можно назвать кроссплатформенной технологией, что тоже является его достоинством.

При использовании PHP не возникает проблем с зависимостью от браузеров, так как сценарии, написанные на нём, компилируются на стороне сервера, а затем отправляются клиенту. Можно сказать, что скрипты PHP могут быть переданы любыми устройствами с браузерами, к каким относятся всем известные портативные компьютеры, сотовые телефоны, планшеты и многое другое.

Как уже было указано ранее, для языка PHP существует много различных библиотек, которые удовлетворяют самым разным потребностям разработчиков. Устанавливать такие библиотеки PHP позволяет как из кода программы, так и с помощью популярного сейчас пакетного менеджера зависимостей Composer. К его возможностям относят загрузку всех нужных расширений и библиотек за разработчика, он автоматически загружает необходимые для проекта указанные зависимости.

Работа с Composer наиболее часто осуществляется через программу командной строки, по умолчанию библиотеки устанавливаются из основного репозитория, однако можно задать и другие пути для подкачки библиотек. Стоит уточнить, что каждая библиотека устанавливается отдельно внутри рабочего проекта, а не глобально. Таким образом, все необходимые расширения ставятся в ту директорию, из которой была вызвана команда установки. Также возможности Composer позволяют обновлять установленные пакеты до новых доступных версий.

2.2 SQL и MySQL

Декларативный язык SQL (structured query language – язык структурированных запросов) – язык программирования, применяемый для создания, управления и модификации данных в реляционных базах данных, в свою очередь, управляемых различными системами управления базами данных [1]. SQL является информационно-логическим языком, его использование позволяет производить множество различных манипуляций с данными, хранящимися в базах данных. SQL является одним из самых популярных языков управления базами данных, хотя в настоящее время существует большое количество альтернатив.

Изначально SQL задумывался как пользовательский инструмент работы с данными, однако с течением времени он значительно усложнился и применяется в сложных программных реализациях. Язык имеет множество новых функций, возможностей, которыми ранее не обладал. Оставаясь практически единственным механизмом, связующим прикладное ПО и базы данных, он обогащается новыми конструкциями и объектами для хранения, становясь все больше схожим с самостоятельными языками программирования.

Сейчас SQL является совокупностью различных операторов, инструкций, а также вычисляемых функций. Используемые операторы в языке подразделяются на:

- операторы определения данных – к ним относят операторы CREATE, DROP, ALTER;
- операторы манипуляции данными – такими являются SELECT, INSERT, UPDATE, DELETE и пр.;
- операторы управления транзакциями;
- операторы управления доступа к данным.

При помощи этого языка разработчик может писать как небольшие, так и крупные, вложенные запросы, в зависимости от требований,

предъявляемых к конкретной задаче. SQL обладает рядом достоинств, к которым относят его независимость от конкретной системы управления базами данных (так как в основном тексты запросов, написанных на SQL могут быть легко перенесены из одной СУБД в другую), наличие стандартов для стабилизации языка и соответствия запросов, а также его декларативность. К недостаткам SQL часто относят возможность дублирования строк данных, названий столбцов, поддержку неопределённых значений, а также выросшую сложность языка.

Множество современных СУБД имеют специальные интерфейсы, обеспечивая пользователей и разработчиков возможностью использовать средства визуального построения запросов, что заметно ускоряет и облегчает работу с данными.

В данной работе используется система управления базами данных MySQL – свободная реляционная СУБД, наиболее часто используется для малых и средних проектов [4]. В настоящее время эта система является одной из наиболее популярных и распространённых, она является бесплатной, что обеспечивает её доступность, как для начинающих разработчиков, так и для профессиональных программистов.

MySQL считается надёжной универсальной системой, обеспечивающей высокую скорость работы. Система также считается гибкой, так как поддерживает большое количество различных типов таблиц. Гибкость системы обеспечивается и большим количеством языков программирования, тесно взаимодействующих с ней, помимо PHP к ним относят Java, C++, Python и др [12]. Работа с MySQL может осуществляться на большом количестве различных платформ.

Помимо прочих СУБД MySQL обладает рядом важных преимуществ. Отметим следующие достоинства:

- простота использования – MySQL устанавливается относительно легко, также работу с базами данных упрощает большое количество различных вспомогательных приложений и плагинов;

– обширный функционал – MySQL имеет инструментарий, который может использоваться разработчиком при реализации практически любых задач;

– безопасность – MySQL обладает множеством встроенных функций безопасности, которые изначально работают в ней по умолчанию;

– масштабируемость – система может использоваться для работы, как с малым, так и с большим объемом данных;

– скорость – за счет некоторых упрощенных стандартов системы, обеспечивается её высокая производительность.

Однако, как любое другое программное средство, MySQL имеет свои недостатки, к ним относятся:

– некоторая ограниченность функционала – систему нельзя использовать для разработки проектов со специфическими требованиями к базам;

– недостаточная надежность – несмотря на достоинства системы, в вопросах безопасности некоторых ресурсов, касающихся работы с данными, например, транзакции, MySQL уступает некоторым системам;

– низкая скорость разработки – система не всегда хватает технических возможностей, из-за чего может снижаться эффективность разработки проектов.

Для работы с базами данных в работе используется phpMyAdmin – приложение, которое является web-интерфейсом для администрирования систем управления базами данных MySQL. Его использование позволяет разработчикам запускать команды SQL, манипулировать данными, а также управлять СУБД MySQL без непосредственного ввода SQL команд.

2.3 PhpSpreadsheet

Microsoft Excel – программа для работы с электронными таблицами, созданная корпорацией Microsoft. Она предоставляет множество инструментов для экономико-статистических расчетов, графических инструментов, организует хранение и сортировку данных различного вида. На сегодняшний день Excel является одним из наиболее популярных приложений в мире (рисунок 2.3.1).



Рисунок 2.3.1 – Логотип MS Excel

В системе учета сведений об иностранных студентах некоторая информация может сохраняться на компьютеры пользователей в xls формате. Файл в формате XLS является электронной таблицей, созданной с помощью Microsoft Excel 2007 или Excel 2010. В таком файле хранятся различные данные, привязанные к определенным ячейкам. Адрес каждой ячейки определяется согласно нумерации строк и столбцов таблицы.

Зачастую отчёты необходимо сохранять в формате XLS, позволяющем манипулировать полученными данными и обрабатывать их. Для того чтобы сохранить данные в таком формате при использовании языка программирования PHP, существует специальная библиотека PhpSpreadsheet, которая является продолжением популярной библиотеки PHPExcel (больше не поддерживается).

PhpSpreadsheet позволяет производить импорт и экспорт данных в Excel, применять различные стили оформления к отчётам. Также существует возможность работы с формулами. Использование библиотеки PHPExcel позволяет разработчикам записывать данные в следующих форматах [16]:

- Excel 2007 (.xlsx) и выше;
- Excel 97 (.xls) и более поздние версии;
- Открытый формат документа / OASIS (.ods)
- HTML;
- CSV.

Ядром библиотеки является класс Spreadsheet. Он содержит ссылки на содержащиеся листы, индексы, различные параметры безопасности документов. Кроме этого, этот класс включает в себя множество других полезных методов для обработки получаемых данных [6]. Ниже перечислены несколько основных методов и свойств используемого класса:

- `getActiveSheet()` – создание листа, этот метод позволяет получить доступ к активному листу документа напрямую. Активным листом является тот, который будет активен, когда откроется книга;

- `setTitle()` – позволяет задать название листа;

- `setCellValue()` – принимает параметры координаты ячейки и данных для вывода в ячейку;

- `getColumnDimension()` – получает размерность указанного столбца;

- `getRowDimension()` – получает размерность указанной строки;

- `setAutoSize(true)` – автоматическое задание размера ячейки/столбца

и т.д.;

- `mergeCells` – позволяет объединить указанный диапазон ячеек.

Очень часто возникает необходимость оформить отчёт в определенном стиле, выделить некоторые данные, сделать выделение шрифта или применить рамку с заливкой фона для некоторых ячеек и т.д. Правильное оформление позволяет сконцентрироваться на наиболее важной информации.

Для этих целей в библиотеке PhpSpreadsheet есть целый набор стилей, которые можно применять к ячейкам в Excel.

Назначить стиль ячейке можно двумя способами:

– Применить метод `applyFromArray` для задания сразу нескольких настроек. В него передается массив настроек с множеством параметров (`fill` – заливка, `font` – шрифт, `borders` – рамка, `alignment` – выравнивание, `numberformat` – формат представления данных ячейки, `protection` – защита ячейки).

– Использовать метод класса `getStyle()` для каждого из стилей в отдельности. При его использовании задаются ячейки для установки различных стилей. Например, установка стиля шрифта (жирный, наклонный): `getFill()->getFillType()`. Чтобы задать цвет можно использовать последовательность методов `getStartColor()->setARGB()` – они позволяют задать цвет указанным ранее ячейкам. Ниже приведены примеры выравнивания ячейки по центру, а также задание шрифта.

```
$sheet->getCell('A13')->getStyle()->getAlignment()->
setHorizontal('center');
$sheet->getCell('A15')->getStyle()->getFont()->
setName('Arial')->setSize(16);
```

С помощью возможностей PhpSpreadsheet можно формировать и считывать любые отчёты в виде файлов формата Excel. Использование файлов таких форматов наиболее востребовано для хранения различных данных в табличном виде, проведения различных исследований, построения статистических отчётов и прочего. К тому же, работникам подразделений высших учебных заведений в наше время наиболее привычно использовать такие данные. Следовательно, стоит считать целесообразной возможность сохранения различной информации, хранящейся в разрабатываемом веб-сервисе в файлах данного формата.

2.4 PHPWord

PHPWord –библиотека, написанная на чистом языке PHP, она предоставляет набор классов для работы с разными форматами файлов документов, предоставляет возможности записи и чтения (рисунок 2.4.1). Текущая версия PHPWord поддерживает Microsoft Office Open XML (OOXML или OpenXML), OASIS Open Document Format для приложений Office (OpenDocument или ODF), Rich Text Format (RTF), HTML и PDF [14].



Рисунок 2.4.1 – Логотип MS Word

PHPWord является проектом с открытым исходным кодом [17]. Эта библиотека призвана стать высококачественным программным продуктом, который включает непрерывную интеграцию и модульное тестирование.

Далее приведены некоторые из возможностей, которые можно осуществить при помощи использования библиотеки PHPWord:

- задать свойства документа, например, название, тема, создатель и т.д;
- создать разделы документа с различными настройками, например, ориентация книжная/альбомная, размер страницы и нумерация страниц;
- создать верхний и нижний колонтитулы для каждой секции;
- вставить абзацы либо как простые текстовые, либо сложные (текстовые), содержащие другие элементы;
- вставить заголовки текста и оглавление;

- вставить разрывы в тексте, а также разрывы страниц;
- задать тип шрифта по умолчанию, его размер, а также стиль абзаца;
- определить пользовательские стили шрифтов (например, полужирный, курсив, цвет текста) и стили абзаца (например, расположенные по центру, многоколоночные, различные интервалы);
- вставить элементы списка как маркированные, пронумерованные или многоуровневые;
- вставить и форматировать изображения, локальные, удаленные, также возможно работать с водяными знаками страниц;
- вставить разнообразные объекты (такие как дуга, кривая, линия, прямоугольник, овал), а также графики;
- вставить и форматировать таблицы с настраиваемыми свойствами для каждой строки или ячеек (например, цвет фона, rowspan, colspan);
- вставить различные гиперссылки и сноски;
- создавать документы из шаблонов, что в этой работе является главной возможностью, и многое другое.

Использование PHPWord для работы с шаблонами осуществляется посредством использования методов «loadTemplate» и «setValue». В документе заранее указываются переменные для дальнейшей замены на данные. Ниже приведем пример использования этих средств.

```
$document = $PHPWord->loadTemplate('Template.docx');
$document->setValue('fio', $fio);
```

Основным элементом документа в MS Word является раздел. Разделом называют прямоугольную область, внутри которой помещаются остальные различные элементы страницы, например текст, изображения, таблицы и т.д. Раздел может иметь как альбомную, так и книжную ориентацию, поля (margins), настраиваемые цвета, границы и их толщину.

Также библиотека PHPWord позволяет добавлять в создаваемый документ изображения двух типов: из файла (используется метод «addImage») и динамически сгенерированное сценарием изображение (метод

«addMemoryImage»). В библиотеке поддерживаются файлы форматов изображений jpg, png, gif, tiff, bmp. Ниже приведем примеры использования этих методов:

```
$section->addImage($src[, $imageStyle]);  
$section->addMemoryImage($url[, $imageStyle]);
```

Где «\$src» и «\$url» – путь к файлу и абсолютный путь к сценарию соответственно, а «\$imageStyle» – массив с форматированием изображения.

В качестве единицы измерений значений размеров и отступов используется типографический твип. Твипом называются типографская единица измерения, которая равна 1/1440 дюйма или около 1/567 сантиметра.

Библиотека PHPWord в свою очередь содержит в своем составе специальный класс, который является конвертером основных известных единиц в твины [15]. То есть, для конвертации пикселей в твины, необходимо использовать метод класса «pixelToTwip()», который переведёт значения в необходимые единицы измерения.

В работе данная библиотека используется для создания различных документов, имеющих определённые шаблоны. PHPWord обладает огромнейшим функционалом и позволяет формировать документы MS Word различной сложности.

2.5 Дополнительные инструменты

В реализуемом сервисе используются также сопутствующие технологии, которые включают в себя различные плагины, библиотеки и т. д. Использование дополнительных инструментов упрощает пользование сервисом, увеличивает скорость и уменьшает возможность ошибок пользователей при его использовании.

AJAX

AJAX (англ. Asynchronous JavaScript and XML) – это технология взаимодействия с сервером без перезагрузки страницы [7]. Множество таблиц в веб-сервисе загружаются после выбора определенного фильтра, и так как не требуется каждый раз обновлять страницу целиком, повышается скорость работы с сервисом и удобство его использования.

В работе такой технологии можно выделить 4 основных этапа, которые перечислены ниже:

1. Пользователь вызывает AJAX. Обычно это реализуется с помощью нажатия некоторой кнопки, предлагающей получить информацию и т.п.
2. Система отправляет на сервер запрос, а также все необходимые данные. Например, может потребоваться загрузка конкретного файла или определённых сведений из базы данных.
3. Сервер получает ответ от базы и отправляет информацию в браузер.
4. JavaScript получает ответ, расшифровывает его и выводит пользователю на экран.

Для обмена данными на странице создается объект XMLHttpRequest, он выполняет функцию некоего посредника между браузером и сервером [6]. Запросы могут отправляться одним из двух типов – GET и POST.

AJAX базируется на использовании технологии динамического обращения к серверу «на лету». Действия, производимые с интерфейсом, преобразуются в операции с DOM элементами, с помощью которых

обрабатываются доступные пользователю данные, в результате чего их представление изменяется.

JQuery-Mask-Plugin

Также в работе сервиса используется плагин JQuery-Mask-Plugin, который предназначен для установки маски ввода определённому элементу «input» с помощью JavaScript кода.

Прост в установке и его настройке. Работа плагина заключается в форматировании вводимых пользователем данных в полях ввода и приведении их к виду, которое задано соответствующей маской в коде [11]. Кроме того, плагин JQuery-Mask-Plugin автоматически производит фильтрацию вводимых данных и не позволяет пользователю ввести, например, недопустимое количество символов или символов другого формата.

Плагин можно инициализировать как при помощи «data»-атрибутов полей «input», так и с помощью вызова функции \$.mask() с различными опциями. Ниже приведён пример использования масок для форматирования даты и номера телефона.

```
$('.date').mask('00/00/0000');  
$('.phone').mask('+0 (000) 000 00 00', {placeholder: "+_  
(__ ) __ _ __"});
```

jQuery

Библиотека jQuery представляет собой набор функций JavaScript, который фокусируется на взаимодействии HTML и JavaScript. Библиотека помогает легко получать доступ к любым DOM элементам, обращаться к их атрибутам и содержимому, манипулировать элементами [10]. Также библиотека jQuery предоставляет удобный API для работы с технологией AJAX.

Работу с библиотекой jQuery можно разделить на 2 типа:

1. Получение объекта jQuery с помощью функции «\$()». Например, можно передать в неё CSS-селектор и получить jQuery-объект всех элементов HTML, которые попадают под заданный критерий, далее можно работать с

ними с помощью различных методов этого объекта. Если метод не возвращает какое-либо значение, то он возвращает ссылку на jQuery объект, к которому можно применять различные последовательные методы.

2. Вызов глобальных методов у объекта «\$».

jQuery UI

jQuery UI построен поверх основной библиотеки jQuery, он предоставляет разработчику упрощенный доступ к её функциям взаимодействия. jQuery UI представляет собой набор различных взаимодействий, эффектов, виджетов и тем пользовательского интерфейса jQuery. Является хорошим решением для создания веб-приложений с высокой степенью интерактивности или, например, просто для добавления средства выбора даты в элемент управления формы.

Autocomplete

Виджеты jQuery UI Autocomplete предоставляют подсказки при вводе в поле. Например, предлагаются имена студентов, которые отображаются, когда в поле введено не менее двух символов. Источником данных является серверный скрипт, который возвращает данные, указанные с помощью функции, которая использует jQuery.ajax() [9].

3 Реализация

3.1 База данных

Для разработки веб-сервиса была создана база данных «service_db» для хранения информации о каждом студенте, некоторых документах. Её структура показана на рисунке 3.1.1.

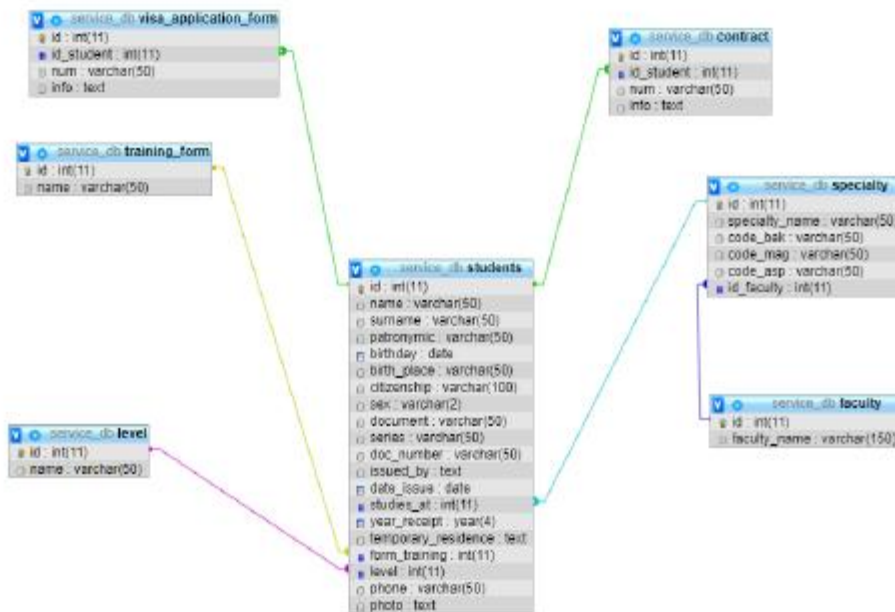


Рисунок 3.1.1 – База данных

Итак, можно увидеть следующие таблицы:

– таблица «students», предназначенная для хранения данных об иностранных студентах;

– таблица «faculty», в которой хранится список доступных факультетов, будем рассматривать на примере факультетов МГУ им. Н. П. Огарёва;

– таблица «speciality», предназначенная для хранения информации о направлениях обучения ВУЗа, с соответствующими кодами для каждого уровня обучения и факультета;

– таблица «level», в которой содержится информация об уровнях обучения (в данной работе – бакалавриат, магистратура, аспирантура);

– таблица «training_form» с данными о форме обучения (в данной работе – очная, заочная);

– таблица «contract» для хранения данных о договорах студентов;

– таблица «visa_application_form», в которой сохраняются данные из визовых анкет студентов после их заполнения;

Таблица «faculty» содержит поля «id» и «faculty_name» для хранения id и названия факультетов. Ознакомиться с таблицей можно на рисунке 3.1.2.



id	faculty_name
1	АРХИТЕКТУРНО-СТРОИТЕЛЬНЫЙ ФАКУЛЬТЕТ
2	ГЕОГРАФИЧЕСКИЙ ФАКУЛЬТЕТ
3	ФАКУЛЬТЕТ БИОТЕХНОЛОГИИ И БИОЛОГИИ
4	ФАКУЛЬТЕТ ДОВУЗОВСКОЙ ПОДГОТОВКИ И СРЕДНЕГО ПРОФЕС...
5	ФАКУЛЬТЕТ ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ
6	ФАКУЛЬТЕТ ИНОСТРАННЫХ ЯЗЫКОВ
7	ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ
8	ФИЛОЛОГИЧЕСКИЙ ФАКУЛЬТЕТ
9	ЭКОНОМИЧЕСКИЙ ФАКУЛЬТЕТ
10	ЮРИДИЧЕСКИЙ ФАКУЛЬТЕТ
11	АГРАРНЫЙ ИНСТИТУТ
12	ИНСТИТУТ МЕХАНИКИ И ЭНЕРГЕТИКИ
13	ИНСТИТУТ НАЦИОНАЛЬНОЙ КУЛЬТУРЫ
14	ИНСТИТУТ ФИЗИКИ И ХИМИИ
15	ИНСТИТУТ ЭЛЕКТРОНИКИ И СВЕТОТЕХНИКИ
16	ИСТОРИКО-СОЦИОЛОГИЧЕСКИЙ ИНСТИТУТ
17	МЕДИЦИНСКИЙ ИНСТИТУТ

Рисунок 3.1.2 – Таблица «faculty»

Таблица «speciality» содержит следующие поля:

– «id», в котором хранится id направления;

– «speciality_name» для хранения названия направления;

– «code_bak», «code_mag», «code_asp» – коды соответствующих уровней обучения;

– «faculty_id» – поле для связи с таблицей «faculty», указывает на принадлежность направления определённому факультету.

Таблица «level» имеет поля «id» – id уровня обучения и «name» – тип уровня, а именно бакалавриат, магистратура или же аспирантура.

В таблице «training_form» определены поля «id» – id формы обучения и «name» – тип формы, т. е. очная или заочная.

Таблица «contract» имеет поля «id_student» для связи с таблицей студентов, «num» для хранения номера договора и «info» для хранения данных документа в формате JSON. Таблица «visa_application_form» содержит аналогичные поля, служит для хранения данных, полученных после заполнения визовой анкеты.

Как можно увидеть, основной является таблица «students», в ней хранится основная информация о студентах. Структура таблицы показана на рисунке 3.1.3.

Опишем некоторые её поля более подробно. «name», «surname», «patronymic» отвечают за хранение ФИО студента. «birthday», «citizenship», «sex» – дата рождения, гражданство и пол студента соответственно. Далее в полях «document», «series», «doc_number», «issued_by», «data_issue» хранится информация о главном документе студента – его тип, серия, номер (если есть), кем он был выдан и когда. В поле «studies_at» хранится id направления из таблицы «speciality», оно указывает, на каком направлении факультета обучается студент. «year_receipt» хранит информацию о годе поступления, «level», «form_training» – уровень и форма обучения, хранятся в виде id из соответствующих таблиц «level» и «form_training». В поле «temporary_residence» находится адрес временной прописки. Также в таблице содержится поле «phone» для хранения контактного номера телефона и «photo» – в нём расположены пути к файлам с фото студента на сервере.

Структура таблицы		Связи	
#	Имя	Тип	
<input type="checkbox"/>	1	id	int(11)
<input type="checkbox"/>	2	name	varchar(50)
<input type="checkbox"/>	3	surname	varchar(50)
<input type="checkbox"/>	4	patronymic	varchar(50)
<input type="checkbox"/>	5	birthday	date
<input type="checkbox"/>	6	birth_place	varchar(50)
<input type="checkbox"/>	7	citizenship	varchar(100)
<input type="checkbox"/>	8	sex	varchar(2)
<input type="checkbox"/>	9	document	varchar(50)
<input type="checkbox"/>	10	series	varchar(50)
<input type="checkbox"/>	11	doc_number	varchar(50)
<input type="checkbox"/>	12	issued_by	text
<input type="checkbox"/>	13	date_issue	date
<input type="checkbox"/>	14	studies_at	int(11)
<input type="checkbox"/>	15	year_receipt	year(4)
<input type="checkbox"/>	16	temporary_residence	text
<input type="checkbox"/>	17	form_training	int(11)
<input type="checkbox"/>	18	level	int(11)
<input type="checkbox"/>	19	phone	varchar(50)
<input type="checkbox"/>	20	photo	text

Рисунок 3.1.3 – Структура таблицы «students»

Таким образом, данные в базе хранятся в виде нескольких отдельных таблиц, связанных между собой. Для работы с базой используется расширение PHP PDO, которое может представлять специфичный для неё функционал в виде некоторых стандартных функций расширения. Это расширение обеспечивает абстракцию доступа к данным.

Ниже на рисунке 3.1.4 показан пример подключения к базе данных. Для этого объявляются параметры host, название базы данных, имя пользователя, его пароль. Так же указываем порт для подключения, выставляем различные опции. Создаём новый экземпляр класса PDO по заданным параметрам.

```
$host = 'localhost';
$db   = 'service_db';
$user = 's[REDACTED]';
$pass = '9[REDACTED]';

$dsn = "mysql:host=$host:3307;dbname=$db";
$opt = [
    PDO::ATTR_ERRMODE           => PDO::ERRMODE_EXCEPTION,
    PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC,
    PDO::ATTR_EMULATE_PREPARES  => false,
];
$pdo = new PDO($dsn, $user, $pass, $opt);
```

Рисунок 3.1.4 – Подключение к базе данных

Рисунок 3.2.2 – Фрагмент формы для добавления данных о студенте

Поля формы «факультет» и «направление» представлены в виде выпадающих списков. При выборе определенного факультета автоматически подгружаются данные в поле с направлением текущего факультета при помощи технологии AJAX. Для получения данных о направлении направляем id факультета на обработку скрипту, где создаётся подключение к базе данных и SQL-запрос для выборки всех направлений данного факультета из таблицы «specialty» по значению поля «id_faculty». Затем созданный запрос выполняется, и извлечённые данные появляются на форме в виде списка.

Другие поля вводятся вручную с клавиатуры. Для полей с датой, а также для записи номера телефона определена маска ввода (mask('0000-00-

00') и mask('+7(000) 000-0000') соответственно) при помощи плагина JQuery-Mask-Plugin. Фотография студента загружается как обычный файл.

После заполнения всех полей и нажатия кнопки «Сохранить» создаётся объект FormData, который позволяет создать набор пар ключ/значение из заданных значений и передать их на дальнейшую обработку.

С помощью AJAX отправляем сформированные данные на выполнение скрипту сохранения информации о студентах в базу. Внутри него происходит сохранение файла изображения в папку «upload» на сервере, в отдельную переменную запоминается путь до него. Далее все прочие полученные поля проходят обработку при помощи функции «filter_var», которая фильтрует контент, используя определённый фильтр. Здесь применяется фильтр «FILTER_SANITIZE_STRING».

Так как в форме данные о временной прописке заполняются по отдельности (район, город и т. д.), а в базе данных они хранятся в одном поле, то для хранения адреса в виде одного поля формируется JSON-строка из соответствующих полей, связанных с ним.

Готовим SQL-запрос для записи обработанных данных в таблицу «students». Используем метод «prepare» для подготовки запроса, определяем оператор INSERT INTO, указываем нужную таблицу и поля, в которые необходимо передать перечисленные значения VALUES в следующем формате «:name, :surname» и т. п. Собираем массив, в котором указываем каким значениям VALUES соответствуют полученные ранее значения.

Далее выполняем запрос при помощи метода «execute», таким образом, добавляем запись о студенте в таблицу «students» базы данных.

Для редактирования данных о конкретном студенте необходимо перейти в раздел «Студенты->Редактирование» (рисунок 3.2.3).

Так как был осуществлен переход непосредственно к редактированию данных после выбора пункта меню (необходимо проверить наличие GET-параметра «id» в URL-адресе, в этом случае его нет), то на странице показывается форма поиска студента.

Рисунок 3.2.3 – Форма поиска студента

На форме присутствуют такие элементы как: выбор факультета, направления и строка поиска фамилии студента. Направления обучения аналогично добавлению студентов подгружаются автоматически в зависимости от выбранного факультета. Поиск по фамилии осуществляется посредством AJAX и виджета Autocomplete, то есть после ввода первых букв фамилии происходит запрос на сервер и по данным фильтров получаем список студентов, в фамилиях которых содержатся данные последовательности букв.

После выбора из списка нужного человека, на страницу подгружается часть данных для просмотра и редактирования информации о выбранном студенте (рисунок 3.2.4).


После изменения некоторых данных необходимо сохранить результат, нажав кнопку «Сохранить». Логика отправки данных на сервер аналогична запросу добавления. Для построения SQL-запроса используется оператор UPDATE, по id студента задаются (SET) обновленные значения.

РЕДАКТИРОВАНИЕ ДАННЫХ	
Данные студента	Временная прописка
Фамилия Ван	Район Мордовия
Имя Нурлан	Город Саранск
Отчество	Улицы Ульянова
Дата рождения 1991-10-17	Дом 12
Место рождения Кавказстан	Корпус
Гражданство Кавказстан	квартира 5
Пол Муж	Факультет ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ
Документ паспорт	Направление подготовки Программная инженерия
Серия 700232	Форма обучения очная
Номер	Уровень Бакалавриат

Рисунок 3.2.4 – Фрагмент формы редактирования

Для формирования различных документов перейдём на вкладку «Документы». Здесь необходимо выбрать интересующий факультет, соответствующее направление, и начать вводить имя студента, аналогично операции редактирования. После этого на странице выводится краткая информация о студенте, а также 3 карточки документа на выбор (рисунок 3.2.5). Доступные документы: временная регистрация, визовая анкета, договор.

При выборе определённого документа по необходимости нужно заполнить форму с дополнительными полями. Пример заполнения дополнительной формы при выборе документа «Договор» показан на рисунке 3.2.6. Данные о студенте автоматически будут заполнены в документе в зависимости от выбранного ранее студента.



[главная](#)
[студенты](#)
[документы](#)
[справки](#)

ДОКУМЕНТЫ

Введите данные для выбора студента

Ван Нурлан

Данные о студенте

ФИО: Ван Нурлан

Дата рождения: 1991-10-17

Дата поступления: 2019

Заполните необходимые поля для формирования документов

ВРЕМЕННАЯ РЕГИСТРАЦИЯ

Срок пребывания до

Фамилия (принимающая сторона)

Имя (принимающая сторона)

Отчество (принимающая сторона)

создать документ

Рисунок 3.2.5 – Фрагмент раздела «Документы»

ДОГОВОР

Курс

1

Срок освоения образовательной программы (продолжительность обучения) на момент подписания договора составляет

4 года

Срок обучения по индивидуальному учебному плану, в том числе ускоренному обучению, составляет

2019/2020

Стоимость обучения в

2020

учебном году составляет сумму в размере

100000

рублей.

Величина стоимости образовательных услуг за весь период обучения составляет:

000000

рублей (сумма цифрами и прописью).

ОТПРАВИТЬ

Рисунок 3.2.6 – Форма «Договор»

После нажатия соответствующей кнопки для создания документов, формируем из всех необходимых данных объект пар «ключ-значение» и отправляем их на дальнейшую обработку.

Заранее для работы были созданы шаблоны определённых документов. На тех позициях, где должен находиться динамический текст, в документе вставляется метка «`{}`», внутри фигурных скобок указывается параметр для вывода (рисунок 3.2.7).

1. ПРЕДМЕТ ДОГОВОРА

1.1. Исполнителя обязуется предоставить образовательную услугу, а Заказчик обязуется оплатить обучение по образовательной программе в пределах федерального государственного образовательного стандарта или образовательного стандарта в соответствии с учебным планом, в том числе индивидуальным, и образовательными программами Исполнителя.
Вид образовательной программы: основная образовательная программа высшего образования.
Уровень профессионального образования: `{level}`
Факультет/институт/филiaal `{faculty}`
Специальность/направление подготовки: `{specialty}`
Код `{code}`
Форма обучения: `{training_form}`
Курс: `{course}`
1.2. Срок освоения образовательной программы (продолжительность обучения) на момент подписания Договора составляет `{period}`
Срок обучения по индивидуальному учебному плану, в том числе ускоренному обучению, составляет `{training_period}`
1.3. После освоения Обучающимся образовательной программы и успешного прохождения государственной итоговой аттестации ему выдается диплом государственного образца о высшем образовании и о квалификации.

Рисунок 3.2.7 – Фрагмент шаблона документа «Договор»

Принцип заполнения шаблонов для разных документов схож. Опишем его на примере заполнения визовой анкеты. Подключаем базу данных, готовим запрос SELECT для получения последнего хранящегося номера документа, для этого обращаемся к таблице «`visa_application_form`», из неё получаем последний «`num`», и выполняем запрос. Увеличиваем номер на единицу.

Далее все полученные методом POST поля должны пройти обработку с использованием функции «`filter_var`», применяем фильтр «`FILTER_SANITIZE_STRING`». Все числовые значения приводим к `int`-формату во избежание ошибок.

Подключаем библиотеку PHPWord для работы с документом. Создаём экземпляр класс PHPWord. Используем метод «`loadTemplate`» для загрузки

шаблона данного документа. Проверяем выбор кратности визы – для этого определяем, какой параметр пришёл заполненным, на это место вставляем символ «x». Аналогично проверяется пол и выбор категории. В случае если была выбрана категория «обыкновенная», на форму подгружается дополнительный выпадающий список, таким образом, необходимо проверить и его.

Далее используем метод «setValue» для того, чтобы заполнить все параметры в шаблоне. Для заполнения полей латиницей, используем пользовательскую функцию для транслитерации кириллических символов полей surname, name, patronymic.

Также в шаблоне этого документа содержится таблица, в которую необходимо вносить данные о ближайших родственниках (рисунок 3.2.8). Для этого вносим построчно в общий массив подмассивы, хранящие информацию о каждом указанном родственнике. Для заполнения таблицы используем метод «cloneRowAndSetValues», который находит строку в таблице, идентифицированную заданным параметром, и клонирует ее столько раз, сколько записей передано в массиве.

16. Прошу внести в визу данные о моих близких родственниках (несовершеннолетние дети либо другие члены семьи, которые вписаны в документ, удостоверяющий личность заявителя):

Фамилия, имя, отчество (при наличии)	Дата рождения (дд/мм/гггг)	Гражданство	Адрес постоянного проживания
\${surname_data}	\${birthday_data}	\${citizenship_data}	\${address_data}

Рисунок 3.2.8 – Шаблон таблицы

Сохраняем документ, используя метод класса PHPWord «saveAs». Далее необходимо добавить в таблицу базы данных информацию о сгенерированном документе, используя оператор INSERT INTO. В качестве параметра «info» передаём JSON-строку с данными документа. Отправляем

пользователю ссылку на файл, который автоматически сохраняется на компьютер (рисунок 3.2.9).

Приложение к приказу
МВД России, ФСБ России, МИД России
от 27.04.2017 № 233/235/7018

ФОРМА

Регистрационный № 12

Место
для
фотографии

ВИЗОВАЯ АНКЕТА

Прошу оформить, продлить, восстановить (неуживое зачеркнуть) визу:

кратность визы: <input checked="" type="checkbox"/> однократная <input type="checkbox"/> двукратная <input type="checkbox"/> многократная	категория (вид) визы: <input checked="" type="checkbox"/> обыкновенная (<input type="checkbox"/> частная <input type="checkbox"/> гуманитарная <input type="checkbox"/> деловая <input type="checkbox"/> рабочая <input type="checkbox"/> туристическая <input type="checkbox"/> туристическая групповая <input checked="" type="checkbox"/> учебная <input type="checkbox"/> на въезд в Российскую Федерацию в целях приема в гражданство Российской Федерации) <input type="checkbox"/> транзитная <input type="checkbox"/> временно проживающего лица
--	---

в связи с обучением

О себе сообщите следующие сведения:

1. Фамилия Кутрапалли / Кутрапалли
(фамилия) (латиница)

2. Имя (имена) Раджеш / Radzhesh
(имя) (латиница)

3. Отчество (при наличии) _____ / _____
(отчество) (латиница)

4. Дата рождения 10/01/1998 5. Место рождения Индия 6. Пол: М Ж
(дд/мм/гггг) (страна, город)

7. Гражданство Индия 8. Документ, удостоверяющий личность: паспорт

Рисунок 3.2.9 – Пример сформированного документа

Для того чтобы сформировать документ формата Excel по определённым параметрам или же просмотреть информацию по студентам, необходимо перейти на вкладку «Списки». Чтобы загрузить список студентов, нужно выбрать и ввести необходимые параметры по следующим критериям:

- факультет;
- направление;
- год поступления;
- форма обучения;
- уровень обучения.

Обязательным является поле «факультет», другие выбираются опционально.

После заполнения данных и нажатия кнопки «Поиск», формируем запрос к базе данных. Объединяем таблицы «students», «speciality», «faculty», «training_form» и «level» по соответствующим id. Используем оператор SELECT для выбора всех необходимых полей из таблиц, для получения кода направления используем оператор CASE-WHEN, в котором по выбранному уровню обучения (бакалавриат, магистратура и т. п.) получаем параметр «code_bac», «code_mag» или «code_asp». Так как в некоторых таблицах названия полей одинаковые, необходимо переименовать их, используя для этого «As». Ставим условие (WHERE) для выбора только тех студентов, которые соответствуют выбранному в форме факультету, сортируем (ORDER BY) их по фамилии.

По мере выбора опциональных параметров в запрос дописываются дополнительные фильтры, например, при выборе уровня обучения в запрос запишем «AND students.level = ?». Выполняем запрос.

После этого на странице появляется список студентов, подходящих под заданные параметры поиска (рисунок 3.2.10). Из этого списка можно так же перейти детальную страницу и на редактирование отдельного студента, нажав кнопку «Просмотр/редактирование».

Если был выбран этот пункт, то изначально на странице редактирования при просмотре информации все поля недоступны, для непосредственного редактирования данных необходимо нажать кнопку «Разрешить редактирование». После её нажатия поля становятся доступными (удаляется атрибут disabled).

СПИСКИ

Год поступления: _____

ПОИСК [Очистить](#)

ФИО	Дата рождения	Пол	Гражданство	Год поступления	Факультет	Направление	Уровень	Форма обучения	Код	Телефон	
Алик Александрович	1993-10-11	М	Беларусь	2020	ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ	Физик	Бакалавриат	очная	02.03.02	+7(222) 253-6535	Проверить/Редактировать
Нурлан Бах	1991-10-17	М	Казахстан	2019	ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ	Программная инженерия	Бакалавриат	очная	09.03.04	+7(222) 253-6535	Проверить/Редактировать
Синар Сулдин	1993-11-01	М	Узбекистан	2020	ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ	Программная инженерия	Бакалавриат	очная	09.03.04	+7(222) 222-28	Проверить/Редактировать
Радони Куртаманов	1998-01-10	М	Индия	2019	ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ	Физик	магистратура	очная	02.04.02	+7(222) 222-11	Проверить/Редактировать

СОХРАНИТЬ В EXCEL

Рисунок 3.2.10 – Список студентов по заданному параметру

Для того чтобы полученные данные скачать в формате `xlsx` необходимо нажать на кнопку «Сохранить в Excel». После этого полученные в фильтре данные передаются на обработку скрипту для формирования файла Excel.

Внутри него подключаем классы `PhpSpreadSheet`, `Xlsx` и `Alignment`, полученные данные декодируются, затем создается новый объект класса `Spreadsheet`, который будет представлять собой рабочую область. Указываем в нём активный лист методом «`getActiveSheet()`» и задаём его название – «Список студентов», используя «`setTitle()`».

Теперь необходимо разметить шапку таблицы. Для этого используем метод «`setCellValue()`» и в каждую ячейку диапазона `A1:R1` вставляем названия полей, например в ячейке `A1` будет храниться ФИО студента, в ячейке `B1` – дата рождения и т.д.

Некоторые данные, которые будут выгружены в Excel, могут иметь различную длину, то есть длинные/короткие имена или адреса. Для того, чтобы они отображались корректно, необходимо выставить свойством автоматическое изменение размера столбцов, в которые они будут записаны. Для этого используем методы «getColumnDimension()» и «setAutoSize(true)» для каждого используемого столбца.

Теперь выделим шапку документа другим цветом, а также сделаем шрифт жирным для её визуального отделения. Для этого используем следующую команду:

```
$sheet -> getStyle('A1:R1')-> getFill()->
setFillType(PhpOffice\PhpSpreadsheet\Style\Fill::FILL_SOLID
->getStartColor() -> setARGB('66ff00');
```

Далее в цикле, начиная со 2 строки и используя «setCellValue()» построчно записываем значения в ячейки, например в ячейки столбца D будет вставляться информация о гражданстве, таким образом выводятся данные о каждом студенте.

После этого выровняем все записи в таблице по центру, используя «getStyle()->applyFromArray», где в качестве настроек alignment передадим следующие параметры выравнивания: Alignment::HORIZONTAL_CENTER, Alignment::VERTICAL_CENTER. Применяем эти настройки ко всему заполненному диапазону.

Теперь необходимо создать файл для записи и передать в него сформированные данные, для этого создаём новый объект класса Xlsx, указывая рабочий объект класса Spreadsheet. Используем метод «save()», указываем путь до сформированного файла. Итак, пример полученного файла формата xlsx показан на рисунке 3.2.11

	A	B	C	D	E	F	G
1	Имя	Дата рождения	Пол	Гражданство	Факультет	Направление	Уровень
2	Алексей Александрович	1995-10-11	M	Болгария	ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ	Физикт	Бакалавриат
3	Нурлан Ван	1991-10-17	M	Казахстан	ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ	Программная инженерия	Бакалавриат
4	Омар Гудон	1992-11-01	M	Узбекистан	ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ	Программная инженерия	Бакалавриат
5	Рахона Нургалиева	1995-05-18	M	Украина	ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ	Физикт	магистратура

Рисунок 3.2.9 – Фрагмент документа Excel

ЗАКЛЮЧЕНИЕ

В настоящее время все более популярным становится возможность обучения иностранных студентов в Российской Федерации. Российское образование пользуется популярностью во множестве разных стран [5]. Работа по оформлению сопутствующих документов зачастую ведется вручную, создание специализированного сервиса способствует оптимизации процессов, проводимых при такой работе.

В ходе данной работы были изучены существующие библиотеки для формирования документов в форматах DOCX, XLS, их основные классы и методы. Также были изучены различные средства работы с базами данных.

На основе изученного был разработан веб-сервис учета сведений об иностранных студентах в высшем учебном заведении на языке PHP, имеющий возможность добавления, редактирования и хранения данных в базе данных, а также автоматического формирования документов в форматах DOCX и XLS, которые требуются от студентов по части миграционного учёта.

Разработанный сервис имеет следующие возможности:

- добавление данных;
- редактирование данных;
- автоматическое формирование документов в формате docx;
- автоматическое формирование отчётов в формате xlsx.

Использование сервиса эффективнее по времени, чем рукописное заполнение документов. Также сервис предоставляет возможность быстрого предоставления и редактирования информации по заданным параметрам, что упрощает работу с данными.

Функциональность и работоспособность веб-сервиса проверена на реальных устройствах.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Грофф Дж. Р. SQL: полное руководство / Дж. Р. Грофф, П. Н. Вайнберг, Э. Дж. Оппель; пер. с англ. – М. : ООО «И.Д. Вильямс», 2015 – 960 с.
- 2 Иностранцы хотят учиться в России [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: <https://rg.ru/2019/10/28/kolichestvo-zaiavok-na-obuchenie-v-rossii-ot-inostrancev-vyroslo-na-20.html>. – Загл. с экрана.
- 3 Котеров Д. В. PHP 7 / Д. В. Котеров, И. В. Симдянов – СПб. : БХВ - Петербург, 2017 – 1088 с.
- 4 Маклафлин Б. PHP и MySQL. Исчерпывающее руководство / Б. Маклафлин. – СПб. : Питер, 2013 – 512 с.
- 5 Сколько иностранных студентов в России [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: <https://studyinrussia.ru/actual/articles/skolko-inostrannykh-studentov-v-rossii/>. – Загл. с экрана.
- 6 Создание excel файлов на PHP с помощью PhpSpreadSheet [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: <https://hard-skills.ru/other/excel-phpspreadsheet/>. – Загл. с экрана.
- 7 AJAX для новичков [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: <https://habr.com/ru/post/14246/>. – Загл. с экрана.
- 8 AJAX: С чего начать [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: https://developer.mozilla.org/ru/docs/Web/Guide/AJAX/С_чего_начать. – Загл. с экрана.
- 9 Autocomplete | jQuery UI [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: <https://jqueryui.com/autocomplete/#remote-jsonp>. – Загл. с экрана.
- 10 jQuery [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: <https://ru.wikipedia.org/wiki/JQuery>. – Загл. с экрана.

11 jQuery-Mask-Plugin – примеры использования [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: <https://givik.ru/posts/jquery-mask-plugin-primery-ispolzovaniya>. – Загл. с экрана.

12 MySQL [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: http://www.mysql.ru/docs/man/Data_Manipulation.html. – Загл. с экрана.

13 PHP: Руководство по PHP [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: <http://php.net/manual/ru/index.php>. – Загл. с экрана.

14 PHPWord [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: <https://github.com/PHPOffice/PHPWord>. – Загл. с экрана.

15 PHPWord – создание MS Word документов средствами PHP [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: <https://webformymself.com/phpword-sozdanie-ms-word-dokumentov-sredstvami-php/>. – Загл. с экрана.

16 Welcome to PHPSpreadsheets's documentation [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: <https://phpspreadsheet.readthedocs.io/en/latest/>. – Загл. с экрана.

17 Welcome to PHPWord's documentation [Электронный ресурс]. – [Б. м. : б. и.], [201–]. – Режим доступа: <https://phpword.readthedocs.io/en/latest/>. – Загл. с экрана.

ПРИЛОЖЕНИЕ А

(обязательное)

Фрагмент файла «main.js»

```
$( document ).ready(function() {
    $( "#faculty" ).change(function() {
        var id_faculty = $(this).val();
        $('select[name="speciality"]').html("");
        $.ajax({
            method:'post',
            url: 'func/getSpeciality.php',
            data: {ID:id_faculty},
            success: function(res){
                var result = jQuery.parseJSON(res);
                var html = "";
                html += '<option selected disabled>Выберите
направление</option>';
                if (result) {
                    $.each(result, function (index, value) {
                        html += '<option value="" + value.id + "">' +
value.specialty_name + '</option>';
                    });
                }
                $('select[name="speciality"]').html(html);
            }
        });
    });

    function serializeObject(t) {
        var result = {};
```

Продолжение ПРИЛОЖЕНИЯ А

```
$.each(t.serializeArray(), function(i, obj) {
    result[obj.name] = obj.value
})
return result;    }
$(input[name="phone"]).mask('+7(000) 000-0000');
$(input[name="date_issue"]).mask('0000-00-00');
$(input[name="year_receipt"]).mask('0000');
$(input[name="birthday"]).mask('0000-00-00');
$('#filter').submit(function(e) {
e.preventDefault();
var form = $(this);
$.ajax({
    method:'post',
    url: 'func/getStudents.php',
    data: form.serialize(),
    success: function(res){
        if (res) {
            var result = jQuery.parseJSON(res);
            var html = "";
            $('.findStudent').remove();
            $.each(result, function (index, value) {
                html += '<tbody><tr class="findStudent">';
                html += '<td>' + value.name + ' ' + value.surname + ' ' +
value.patronymic + '</td>';
                html += '<td>' + value.birthday + '</td>';
                html += '<td>' + value.sex + '</td>';
                html += '<td>' + value.citizenship + '</td>';
                html += '<td>' + value.year_receipt + '</td>';
```


Продолжение ПРИЛОЖЕНИЯ А

```
        html += '<td>' + value.faculty_name + '</td>';
        html += '<td>' + value.specialty_name + '</td>';
        html += '<td>' + value.TrainingLevel + '</td>';
        html += '<td>' + value.TrainingForm + '</td>';
        html += '<td>' + value.Code + '</td>';
        html += '<td>' + value.phone + '</td>';
        html += '<td><a target="_blank" href="/edit.php?id=' + value.id
+ '&dis=N">Просмотр/Редактирование</a></td>';
        html += '</tr></tbody>';
    });
    $('.students-list').append(html);
    $('.btn-block').html('<input type="button"
class="btn-excel btn btn-primary" value="Сохранить в Excel">');
    toExcel(res);
    }
    });
});
function toExcel(res)
{
    $('.btn-excel').click(function(){
        $.ajax({
            method:'post',
            url: 'func/toExcel.php',
            data: 'jsonData=' + res,
            success: function(res){
                if (res) {
                    var result = jQuery.parseJSON(res);
                    if (!result.ERROR) {
```

Продолжение ПРИЛОЖЕНИЯ А

```
        var link = document.createElement('a');
        link.setAttribute('href',result.LINK);
        link.setAttribute('download',result.FILE_NAME);
        onload=link.click();           }
    }
});
});
}
$('#form').submit(function(e){
e.preventDefault();
var inputFile = $('input[name="photo"]');
var formData = new FormData;
var info = serializeObject($(this));
var form = $(this);
formData.append('img', inputFile.prop('files')[0]);
if (info) {
    $.each(info, function(field, val) {
        formData.append(field, val);    });    }
$.ajax({
    method:'post',
    url: 'func/updateStudent.php',
    data: formData,
    dataType : "json",
    processData: false,
    contentType: false,
    success: function(res){
        $('#message').html("");

```

Продолжение ПРИЛОЖЕНИЯ А

```
        if (res.INFO) {
            $('#message').html(res.INFO);
            $('#message').addClass("alert-success");
        } else {
            $('#message').html(res.ERROR);
            $('#message').addClass("alert-danger p-2");
        }
    });
});
$('.allow-edit').click(function(){
    $('#fieldset.block').removeAttr('disabled');
})
$('#fio').autocomplete({
    source: function( request, response ) {
        $.ajax({
            url: "func/search.php",
            type: 'post',
            dataType: "json",
            data: {
                term: request.term,
                speciality: $('select[name="speciality"] option:selected').val()
            },
            success: function( data ) {
                response( data );
            }
        });
    },
    select: function (event, ui) {
        $('#fio').val(ui.item.label);
    }
});
```

Окончание ПРИЛОЖЕНИЯ А

```
        window.location.replace("/edit.php" + '?id=' + ui.item.id);
        return false;
    });
    $("#form1").submit(function(e){
        e.preventDefault();
        var INFO1 = INFO;

        var dataForm = serializeObject($(this));
        INFO1.stay_duration = dataForm.stay_duration;
        INFO1.surname_2 = dataForm.surname_2;
        INFO1.name_2 = dataForm.name_2;
        INFO1.patronymic_2 = dataForm.patronymic_2;
        $.ajax({
            method:'post',
            url: 'func/toWord.php',
            data: INFO1,
            success: function(res){
                var result = jQuery.parseJSON(res);
                if (!result.ERROR) {
                    var link = document.createElement('a');
                    link.setAttribute('href',result.LINK);
                    link.setAttribute('download',result.FILE_NAME);
                    onload=link.click();
                }
            }
        });
    });
}
```

ПРИЛОЖЕНИЕ Б

(обязательное)

Фрагмент файла «toWord.php»

```
if ($_POST['name'] && $_POST['surname']) {
    require '../lib/vendor/autoload.php';
    $name = filter_var($_POST['name'], FILTER_SANITIZE_STRING);
    $surname = filter_var($_POST['surname'], FILTER_SANITIZE_STRING);
    $patronymic = filter_var($_POST['patronymic'],
FILTER_SANITIZE_STRING);
    $birthday = filter_var($_POST['birthday'], FILTER_SANITIZE_STRING);
    $sex = filter_var($_POST['sex'], FILTER_SANITIZE_STRING);
    $type_document = filter_var($_POST['document'],
FILTER_SANITIZE_STRING);
    $series = filter_var($_POST['series'], FILTER_SANITIZE_STRING);
    $doc_number = filter_var($_POST['doc_number'],
FILTER_SANITIZE_STRING);
    $citizenship = filter_var($_POST['citizenship'],
FILTER_SANITIZE_STRING);
    $stay_duration = filter_var($_POST['stay_duration'],
FILTER_SANITIZE_STRING);
    $district = filter_var($_POST['district'], FILTER_SANITIZE_STRING);
    $city = filter_var($_POST['city'], FILTER_SANITIZE_STRING);
    $street = filter_var($_POST['street'], FILTER_SANITIZE_STRING);
    $house = filter_var($_POST['house'], FILTER_SANITIZE_STRING);
    $corp = filter_var($_POST['corp'], FILTER_SANITIZE_STRING);
    $room = filter_var($_POST['room'], FILTER_SANITIZE_STRING);
    $surname_2 = filter_var($_POST['surname_2'],
FILTER_SANITIZE_STRING);
```

Продолжение ПРИЛОЖЕНИЯ Б

```
$name_2 = filter_var($_POST['name_2'], FILTER_SANITIZE_STRING);
$patronymic_2 = filter_var($_POST['patronymic_2'],
FILTER_SANITIZE_STRING);
$phpWord = new \PhpOffice\PhpWord\PhpWord();
$file_name = 'Template1-' . date('d.m.Y') . '.docx';
$document = $phpWord-
>loadTemplate('../templates/vremennaya_registratsia.docx'); //шаблон
$birthdayArr = explode('-', $birthday);
$stay_durationArr = explode('-', $stay_duration);
$document->setValue('name', $name.' '.$patronymic);
$document->setValue('surname', $surname);
$document->setValue('citizenship', $citizenship);
$document->setValue('d', $birthdayArr[2]);
$document->setValue('m', $birthdayArr[1]);
$document->setValue('y', $birthdayArr[0]);
$document->setValue('sex', $sex);
$document->setValue('document', $type_document);
$document->setValue('series', $series);
$document->setValue('doc_number', $doc_number);
$document->setValue('district', $district);
$document->setValue('city', $city);
$document->setValue('street', $street);
$document->setValue('house', $house);
$document->setValue('corp', $corp);
$document->setValue('room', $room);
$document->setValue('d2', $stay_durationArr[2]);
$document->setValue('m2', $stay_durationArr[1]);
$document->setValue('y2', $stay_durationArr[0]);
```

Окончание ПРИЛОЖЕНИЯ Б

```
$document->setValue('name2', $name_2.' '.$patronymic_2);  
$document->setValue('surname2', $surname_2);  
$document->saveAs('../files/'.$file_name);  
}
```

ПРИЛОЖЕНИЕ В

(обязательное)

Фрагмент файла «toExcel.php»

```
require '../lib/vendor/autoload.php';
use PhpOffice\PhpSpreadsheet\Spreadsheet;
use PhpOffice\PhpSpreadsheet\Writer\Xlsx;
use PhpOffice\PhpSpreadsheet\Style\{Alignment, Fill};
if ($_POST['jsonData']) {
    $data = json_decode($_POST['jsonData'], true);
    $spreadsheet = new Spreadsheet();
    $sheet = $spreadsheet->getActiveSheet();
    $sheet->setTitle('Список студентов');
    $sheet->setCellValue('A1', 'ФИО');
    $sheet->setCellValue('B1', 'Дата рождения');
    $sheet->setCellValue('C1', 'Пол');
    $sheet->setCellValue('D1', 'Гражданство');
    $sheet->setCellValue('E1', 'Факультет');
    $sheet->setCellValue('F1', 'Направление');
    $sheet->setCellValue('G1', 'Уровень');
    $sheet->setCellValue('H1', 'Форма обучения');
    $sheet->setCellValue('I1', 'Код');
    $sheet->setCellValue('J1', 'Временная прописка');
    $sheet->setCellValue('K1', 'Год поступления');
    $sheet->setCellValue('L1', 'Документ');
    $sheet->setCellValue('M1', 'Серия');
    $sheet->setCellValue('N1', 'Номер');
    $sheet->setCellValue('O1', 'Выдан');
    $sheet->setCellValue('P1', 'Дата выдачи');
```


Продолжение ПРИЛОЖЕНИЯ В

```
$sheet->setCellValue('Q1', 'Телефон');
$sheet->getColumnDimension('A')->setAutoSize(true);
$sheet->getColumnDimension('B')->setAutoSize(true);
$sheet->getColumnDimension('C')->setAutoSize(true);
$sheet->getColumnDimension('D')->setAutoSize(true);
$sheet->getColumnDimension('E')->setAutoSize(true);
$sheet->getColumnDimension('F')->setAutoSize(true);
$sheet->getColumnDimension('G')->setAutoSize(true);
$sheet->getColumnDimension('H')->setAutoSize(true);
$sheet->getColumnDimension('I')->setAutoSize(true);
$sheet->getColumnDimension('J')->setAutoSize(true);
$sheet->getColumnDimension('K')->setAutoSize(true);
$sheet->getColumnDimension('L')->setAutoSize(true);
$sheet->getColumnDimension('M')->setAutoSize(true);
$sheet->getColumnDimension('N')->setAutoSize(true);
$sheet->getColumnDimension('O')->setAutoSize(true);
$sheet->getColumnDimension('P')->setAutoSize(true);
$sheet->getColumnDimension('Q')->setAutoSize(true);
$sheet->getStyle('A1:Q1')->getFill()->
setFillType(PhpOffice\PhpSpreadsheet\Style\Fill::FILL_SOLID)->
getStartColor()->setARGB('66ff00');
    $i = 2;
    foreach ($data as $key => $value) {
        $sheet->setCellValue('A' . $i, $value['name'] . ' ' . $value['surname'] .
'. $value['patronymic']);
        $sheet->setCellValue('B' . $i, $value['birthday']);
        $sheet->setCellValue('C' . $i, $value['sex']);
        $sheet->setCellValue('D' . $i, $value['citizenship']);
```

Окончание ПРИЛОЖЕНИЯ В

```
$sheet->setCellValue('E'.Si, $value['faculty_name']);
$sheet->setCellValue('F'.Si, $value['specialty_name']);
$sheet->setCellValue('G'.Si, $value['TrainingLevel']);
$sheet->setCellValue('H'.Si, $value['TrainingForm']);
$sheet->setCellValue('I'.Si, $value['Code']);
$sheet->setCellValue('J'.Si, $value['temporary_residence']);
$sheet->setCellValue('K'.Si, $value['year_receipt']);
$sheet->setCellValue('L'.Si, $value['document']);
$sheet->setCellValue('M'.Si, $value['series']);
$sheet->setCellValue('N'.Si, $value['doc_number']);
$sheet->setCellValue('O'.Si, $value['issued_by']);
$sheet->setCellValue('P'.Si, $value['date_issue']);
$sheet->setCellValue('Q'.Si, $value['phone']);
Si++;
}
$sheet->getStyle('A1:Q'.Si)->applyFromArray([
    'alignment' => [
        'horizontal' => Alignment::HORIZONTAL_CENTER,
        'vertical' => Alignment::VERTICAL_CENTER,
        'wrapText' => true,    ]
]);
$file_name = 'list-' . date('d.m.Y') . '.xlsx';
try {
    $writer = new Xlsx($spreadsheet);
    $writer->save('../files/' . $file_name);
} catch (PhpOffice\PhpSpreadsheet\Writer\Exception $e) {
    echo $e->getMessage();
}
```

ПРИЛОЖЕНИЕ Г
(обязательное)

Фрагмент файла «getStudents.php»

```
/*...*/  
$sqlview = 'SELECT  
    students.id, students.name, .surname, students.patronymic,  
    students.birthday, students.citizenship, students.sex,  
    students.document, students.series, students.doc_number,  
    students.issued_by, students.date_issue, students.year_receipt,  
    students.temporary_residence, students.form_training,  
    students.level, students.phone, specialty.specialty_name,  
    faculty.faculty_name, training_form.name As TrainingForm,  
    level.name As TrainingLevel,  
    (CASE students.level  
        WHEN 1 THEN specialty.code_bak  
        WHEN 2 THEN specialty.code_mag  
        WHEN 3 THEN specialty.code_asp  
        END) As Code  
FROM (((students INNER JOIN specialty  
ON students.studies_at = specialty.id)  
INNER JOIN faculty on specialty.id_faculty = faculty.id)  
INNER JOIN training_form on students.form_training =  
training_form.id)  
INNER JOIN level on students.level = level.id  
WHERE faculty.id = ? ORDER BY students.surname ASC';  
$stmt = $pdo->prepare($sqlview);  
$stmt->execute($filter_ex);
```