**МИНОБРНАУКИ РОССИИ**

**федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Балтийский государственный технический университет «ВОЕНМЕХ» им. Д.Ф. Устинова»**

**(БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  | | |  | | | | | | | | | |
| Факультет | | И | Заведующий кафедрой | | | | |  | | И5 | | |
|  | | индекс |  | | | | |  | | шифр кафедры | | |
| Выпускающая кафедра | | И5 | Скулябина О.В. | | |  |  | | | | | |
|  | | индекс | Фамилия И.О. | | |  | подпись | | | | | | |
| Группа | | И976 | «\_\_\_\_\_» | |  | | | | | | 2021 г. | |
|  | номер группы | | |  |  | |  | | | | |  | | |

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
ПО ПРОГРАММЕ БАКАЛАВРИАТА**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Молодцовой Юлии Сергеевны | | | | | | | | |
| Фамилия, имя, отчество обучающегося в родительном падеже | | | | | | | | |
| **На тему** | Разработка программного обеспечения для импорта данных | | | | | | | |
| в базу данных ИС | | | | | | | | |
|  | | | | | | | | |
| **Направление подготовки** | | | 09.03.04 |  | Программная инженерия | | | |
|  | | | индекс направления | | | |  | полное наименование направления |
|  | | | | | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Профиль** | | Разработка программно-информационных систем | | |
|  | | | наименование профиля |
|  | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Консультант:** | | | | |  | | | |  | **Руководитель:** | | |  | | | | | | |
| *если утвержден приказом* | | | | | подпись | | | |  |  | | | подпись | | | | | | |
|  | | |  | |  | | | |  |  | | | | |  | | Бондарев Е.С. | | |
| ученая степень, ученое звание | | | | | Фамилия ИО | | | |  | ученая степень, ученое звание | | | | | | Фамилия ИО | | | |
| « » |  |  | | | | |  | 2021 г. |  | « » |  |  | | | | | |  | 2021 г. |
|  | | | | | |  | | |  | **Обучающийся:** | | | |  | | | | | |
|  | | | |  | |  | | |  |  | | | | |  | | Молодцова Ю.С. | | |
|  | | | | | |  | | |  | подпись | | | | |  | | Фамилия ИО | | |
|  |  |  | | | | |  |  |  | « » |  |  | | | | | |  | 2021 г. |

**РЕФЕРАТ**

Пояснительная записка 52 с., 30 рис., 2 табл., 15 источн., 1 прил.

ЭЛЕКТРОННЫЙ ДОКУМЕНТООБОРОТ, УЧЕБНЫЕ ПЛАНЫ, ВЕДОМОСТИ, БИЗНЕСС-ПРОЦЕСС ОБУЧЕНИЯ, ИМПОРТ ДАННЫХ, PDF, RTF, БАЗА ДАННЫХ, ЯЗЫКИ ПРОГРАММИРОВАНИЯ

Целью данной выпускной квалификационной работы является разработка программного обеспечения для импорта данных в базу данных ИС.

Объектом исследования являются методы импорта данных из документов в формате PDF и RTF.

Предметом исследования является алгоритм извлечения данных из учебных планов и ведомостей с последующим импортом в базу данных ИС.

В ходе выполнения выпускной квалификационной работы был произведен анализ предметной области, выявлены требования к разрабатываемому приложению и определены задачи. На основе полученных данных, было разработано программное обеспечение, соответствующее поставленным требованиям. Разработанное программное обеспечение позволяет импортировать данные из учебных планов и ведомостей в базу данных информационной системы.

**СОДЕРЖАНИЕ**

[ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ 5](#_Toc74828929)

[ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ 6](#_Toc74828930)

[ВВЕДЕНИЕ 7](#_Toc74828931)

[1 Анализ предметной области 9](#_Toc74828932)

[1.1 Электронный документооборот 9](#_Toc74828933)

[1.2 Бизнес-процесс обучения 9](#_Toc74828934)

[1.3 Функциональные требования к разрабатываемому ПО 14](#_Toc74828935)

[2 Проектирование ПО 16](#_Toc74828936)

[2.1 Структура PDF 16](#_Toc74828937)

[2.2 Структура RTF 19](#_Toc74828938)

[2.3 Требования к данным БД 22](#_Toc74828939)

[2.4 Соотнесение данных из документов с полями БД 27](#_Toc74828940)

[2.4.1 Направление, профиль, квалификация, форма обучения 28](#_Toc74828941)

[2.4.2 Дисциплина, форма аттестации, номер семестра 28](#_Toc74828942)

[2.4.3 Группа, форма аттестации 29](#_Toc74828943)

[2.4.4 Специальный идентификатор ведомости №1 29](#_Toc74828944)

[2.4.5 Специальный идентификатор ведомости №1 30](#_Toc74828945)

[2.4.6 Содержание учебного плана 30](#_Toc74828946)

[2.4.7 Студент 31](#_Toc74828947)

[2.5 Выбор средств разработки 31](#_Toc74828948)

[Язык программирования С++ 31](#_Toc74828949)

[Язык C# 32](#_Toc74828950)

[Язык Python 32](#_Toc74828951)

[Выбор языка программирования 32](#_Toc74828952)

[3 Программная реализация 34](#_Toc74828953)

[3.1 Модуль обработки PDF 34](#_Toc74828954)

[3.2 Модуль обработки RTF 38](#_Toc74828955)

[3.3 Работа с SQL 40](#_Toc74828956)

[3.4 Результат работы ПО 41](#_Toc74828957)

[ЗАКЛЮЧЕНИЕ 44](#_Toc74828958)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 45](#_Toc74828959)

[ПРИЛОЖЕНИЕ А 47](#_Toc74828960)

[Акт о внедрении 47](#_Toc74828961)

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

В настоящей пояснительной записке применяют следующие сокращения и обозначения:

PDF (англ. Portable Document) – формат файла

RTF (англ. Rich Text Format) – формат файла

БД – база данных

ИС – информационная система

ПО – программное обеспечение

ЭДО – электронный документооборот

ЯП – язык программирования

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

В настоящей пояснительной записке применяют следующие термины с соответствующими определениями:

Документооборот — движение документов в организации с момента их создания или получения до завершения исполнения или отправления; комплекс работ с документами: прием, регистрация, рассылка, контроль исполнения, формирование дел, хранение и повторное использование документации, справочная работа.

Машиночитаемый документ — документ, пригодный для автоматического считывания содержащейся в нём информации, записанный на магнитных, оптических и других носителях информации.

Электронный документ (ЭД) — документ, созданный с помощью средств компьютерной обработки информации, подписанный электронной цифровой подписью (ЭЦП) и сохранённый на машинном носителе в виде файла соответствующего формата.

Электронная цифровая подпись (ЭЦП) — аналог собственноручной подписи, являющийся средством защиты информации, обеспечивающим возможность контроля целостности и подтверждения подлинности электронных документов.

ВВЕДЕНИЕ

В настоящее время большая часть документооборота в организациях ведётся в электронном формате. Высшие учебные заведения не являются исключением.

Сведения об обучающихся, учебных планах и результатов освоения дисциплин в БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова хранятся в различных информационных системах, однако для экспорта данных используются файлы-выгрузки в электронном виде в форматах PDF и RTF. Разрабатываемое программное обеспечение для импорта данных призвано автоматизировать процесс оцифровки документов, необходимых для работы ИС «Личный кабинет преподавателя».

Объектом исследования являются методы импорта данных из документов в формате PDF и RTF.

Предметом исследования является алгоритм извлечения данных из учебных планов и ведомостей с последующим импортом в базу данных ИС.

Целью данной выпускной квалификационной работы является разработка программного обеспечения для импорта данных в базу данных ИС.

Для достижения поставленной цели, необходимо решить следующие задачи:

* Провести аналитический обзор структуры документов форматов PDF и RTF;
* Спроектировать ПО;
* Выбрать инструмент разработки;
* Разработать и протестировать ПО.

Выпускная квалификационная работа содержит введение, три раздела и заключение.

В первом разделе проводится анализ предметной области, рассматриваются процессы электронного документооборота и документооборота вуза, проводится анализ способов синхронизации с ИС.

Во втором разделе анализируется общая структура документов форматов PDF и RTF, рассматриваются шаблоны обрабатываемых документов, обозначаются требования к данным БД, описываются требования к соответствию данных и обосновывается выбор средств разработки.

В третьем разделе приводится результат реализации и тестирования ПО.

# Анализ предметной области

## Электронный документооборот

Ведение документации в бумажном формате постепенно уходит в прошлое. Большая часть организаций переходит на электронный документооборот, поскольку он позволяет сэкономить временные и финансовые ресурсы.

Электронный документооборот (ЭДО) – организационно-техническая система, обеспечивающая процесс создания, управления доступом и распространения электронных документов в компьютерных сетях, а также обеспечивающая контроль над потоками документов в организации [1].

ЭДО может быть реализован в полном или частичном формате. Если полный формат предполагает отказ от всех бумажных носителей, то частичный формат позволяет документировать часть процессов в бумажном виде. Выбор формата зависит от целей и потребностей каждой конкретной организации.

## Бизнес-процесс обучения

Концепция, в соответствии с которой вуз представляется в виде набора бизнес-процессов, а управление его деятельностью – как управление бизнес-процессами, была разработана в конце 1980 гг.

Под бизнес-процессом понимается специфически упорядоченная совокупность работ, действий во времени и пространстве с указанием начала и конца и точным определением входов и выходов. Примером образовательного бизнес-процесса является выпуск специалистов по определенной специальности и форме обучения, а экземпляр процесса – выпуск по отдельной специализации.

Обучающиеся университета осваивают программу подготовки согласно учебному плану. Учебный план - это совокупность дисциплин, реализующей кафедры, формы аттестации, периода аттестации и формата аудиторной нагрузки. Каждый семестр студенты изучают определенный набор дисциплин.

Пример структуры учебного плана представлен на рисунке 1:



Рисунок 1 – Структура учебного плана

Для повышения качества образования и в соответствии с положением о текущем, рубежном контроле успеваемости и промежуточной аттестации обучающихся [2] в БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова предусмотрен контроль успеваемости как во время семестра, так и по его итогам.

Контроль во время семестра делится на рубежный контроль успеваемости (рубежная аттестация) и текущий контроль успеваемости (текущая аттестация).

Рубежный контроль успеваемости – контроль знаний обучающихся по итогам половины семестра, определяющий глубину и полноту знаний, умений и навыков обучающихся по окончании изучения части или половины материала дисциплины.

Текущий контроль успеваемости – контроль успеваемости обучающихся, определяющий степень усвоения обучающимися теоретической и практической части дисциплины, производящийся в дискретные временные интервалы лектором и (или) преподавателем(ями), ведущими лабораторные работы и практические занятия по дисциплине.

Данные мероприятия проводятся согласно графику [3]:

* текущая аттестация (текущий контроль успеваемости) проводится на 6 и 14 неделях семестра,
* рубежная аттестация (рубежный контроль) проводится на 10 неделе семестра.

По завершению семестра проводится промежуточная аттестация в виде экзаменационной сессии. Во время экзаменационной сессии преподаватели производят оценку знаний обучающихся согласно рабочим программам дисциплин.

В настоящий момент для обеспечения проведения рубежной, текущей и промежуточной аттестации используются ведомости. Пример ведомости аттестации изображен на рисунке 2. Ведомость - документ установленного образца, который формируется деканатом (или структурным подразделением, наделенных полномочиями деканата).

В аттестационной ведомости указывается номер группы, название дисциплины, реализующая кафедра и пофамильный список студентов с полями для заполнения аттестации.

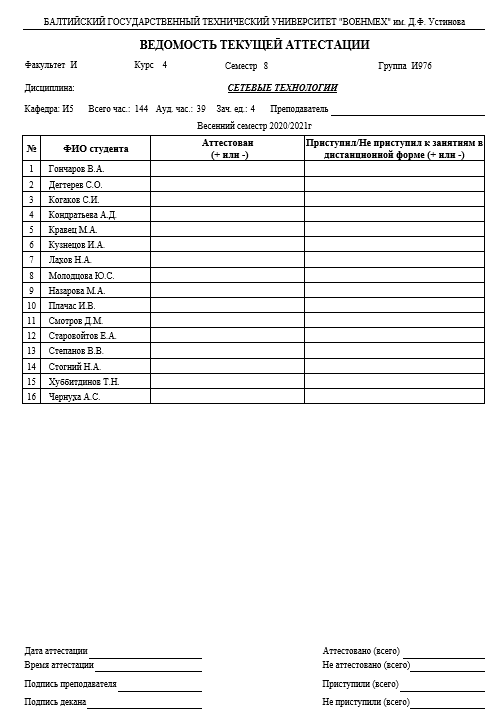


Рисунок 2 – Пример аттестационной ведомости

В экзаменационной ведомости указывается номер группы, название дисциплины, реализующая кафедра, форма аттестации и пофамильный список студентов с указанием номеров зачетных книжек, с полями для внесения оценок. Пример экзаменационной ведомости изображен на рисунке 3.

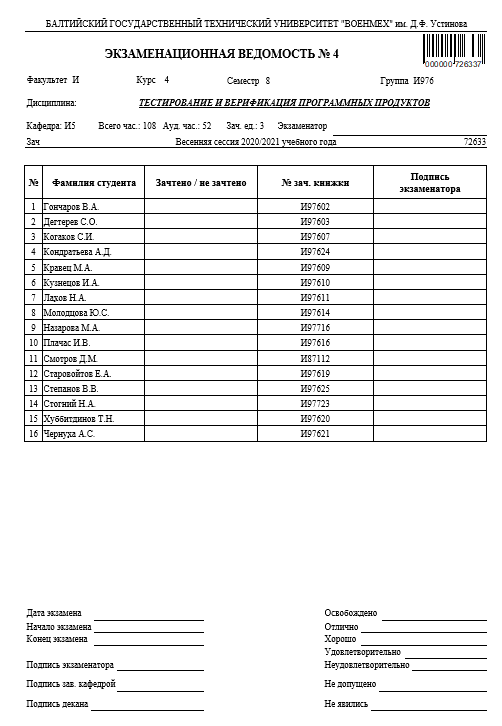


Рисунок 3 – Пример экзаменационной ведомости

Бизнес-процесс проведения любой аттестации можно описать следующим образом:

1. деканат (или структурное подразделение, наделенное полномочиями деканата) подготавливает ведомости из подсистемы ИС УМУ «Деканат» и печатает их;
2. распечатанные ведомости подписываются деканом или заместителем декана;
3. ведомости группируются и отдают в физическом виде по кафедрам секретарям;
4. на кафедре секретарями ведомости раздаются преподавателям для заполнения;
5. преподаватели заполняют ведомости, сдают обратно на кафедру;
6. секретарь кафедры собирает ведомости, проверяет их количество и относит обратно в деканат;
7. полученные ведомости в деканате сортируются по учебным группам и раздаются по заместителям декана;
8. заместители декана оцифровывают ведомости при необходимости;
9. заместители декана производят аналитику и готовят необходимые отчетные документы.

Во время работы в дистанционном режиме в 2020-2021 годах из-за сложной эпидемиологической ситуации в РФ данный бизнес-процесс был изменен:

Бизнес-процесс проведения любой аттестации можно описать следующим образом:

1. деканат (или структурное подразделение, наделенное полномочиями деканата) подготавливает ведомости из подсистемы ИС УМУ «Деканат»;
2. ведомости группируются по кафедрам в архивы и рассылаются по корпоративной почте;
3. на кафедре секретарями ведомости рассылаются на почту преподавателям;
4. преподаватели заполняют ведомости, отправляют обратно на почту кафедры;
5. секретарь кафедры собирает ведомости, проверяет их количество и отправляет обратно в деканат;
6. полученные ведомости в деканате сортируются по учебным группам и отправляются по заместителям декана;
7. при возможности заместители декана оцифровывают ведомости в автоматизированном режиме, остальное вносят вручную;
8. заместители декана производят аналитику и готовят необходимые отчетные документы.

На рисунке 4 изображен пример ведомости, составленной в ИС УМУ.

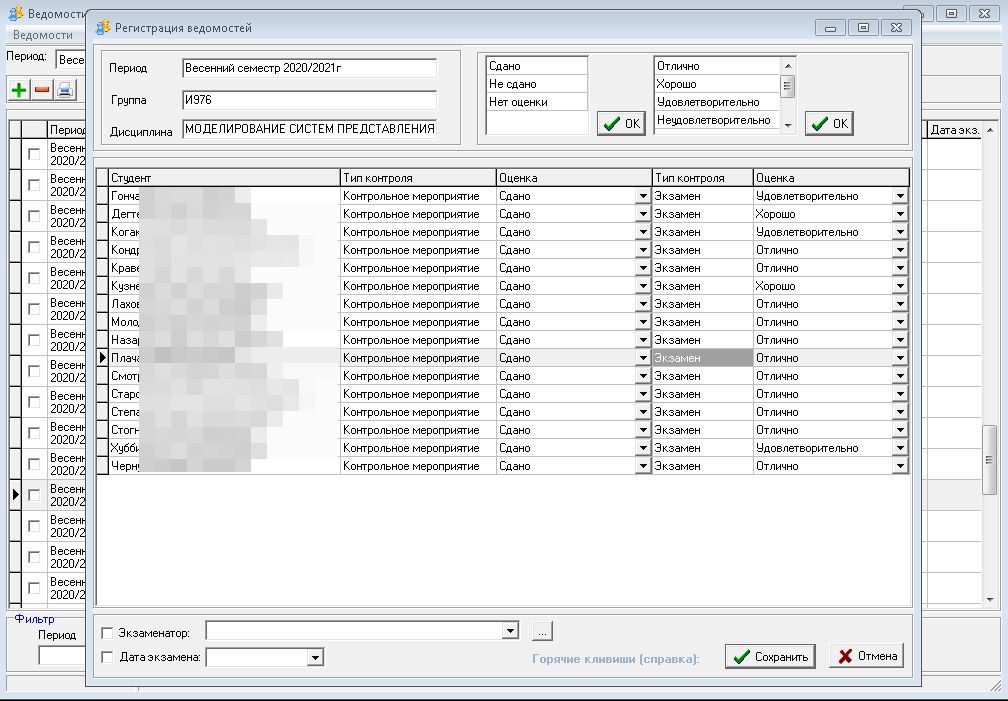


Рисунок 4 – Пример ведомости в ИС УМУ

Бизнес-процесс проведения аттестаций громоздок и не оптимален. Данные задачи решает ИС «Личный кабинет преподавателя», позволяющую проводить аттестации без лишних трудозатрат, однако для ее работы необходимы исходные данные (учебные планы, списки студентов и др.), которые в настоящий момент невозможно получать в автоматизированном формате из-за загруженности разработчиков ИС вуза.

## Функциональные требования к разрабатываемому ПО

Задача разрабатываемого ПО заключается в автоматизации процесса извлечении данных из учебных планов и ведомостей, хранящихся в форматах PDF и RTF, и импорта их в базу данных ИС.

Синхронизация данных – устранение различий между двумя копиями данных. Предполагается, что ранее эти копии были одинаковыми, а затем одна из них, либо обе были независимо изменены.

Методы синхронизации данных:

1. экспорт/импорт – обмен изменёнными данными между копиями;
2. репликация – перенос изменений с «главной копии» на все остальные;
3. API – использование внешних интерфейсов.

Для синхронизации данных разрабатываемое программное обеспечение будет использовать метод импорта/экспорта. Данные из ИС УМУ экспортируются в виде документов формата PDF и RTF, далее извлеченная из них информация импортируется в базу данных ИС.

Итоговые функциональные требования к разрабатываемому ПО:

* выбор обрабатываемого документа;
* извлечение данных из учебных планов и ведомостей, хранящихся в формате PDF и RTF;
* импорт данных в базу данных ИС.

# Проектирование ПО

Ввиду того, что данные для импорта хранятся в документах форматов PDF и RTF, прежде всего необходимо рассмотреть их внутренние структуры.

## Структура PDF

PDF *(Portable Document Format)* – переносимый формат документов, разработанный компанией Adobe Systems, используется как основа для создания электронных изданий в среде программного пакета Adobe Acrobat. Документы формата PDF могут быть представлены на любых аппаратных платформах и операционных системах. Файл PDF часто представляет собой комбинацию векторной графики, текста и растровой графики.

Структуру файла формата PDF можно представить в виде схемы, приведенной на рисунке 5. Эта структура обеспечивает быстрый доступ к любой части файла и содержит механизм для его изменения.

|  |
| --- |
|  |
|  |  |

Рисунок 5 - Структура файла формата PDF

Обычный PDF файл содержит четыре раздела:

* заголовок *(header)*;
* «тело файла» *(body)*;
* таблицу перекрестных ссылок *(cross-reference table)*;
* поиск таблиц, объектов и ссылок *(trailer)*.

Заголовок содержит номер спецификации PDF, используемой в файле.

Тело файла содержит последовательность косвенных объектов, входящих в состав публикации. Объекты – это компоненты публикации, такие как: страницы, изображения, шрифты.

Таблица перекрестных ссылок содержит информацию о каждом объекте в файле в виде одной строки описания места объекта в файле.

Trailer позволяет программному приложению, при чтении PDF файла, быстро находить таблицу перекрестных ссылок и специальные объекты. Приложения должны читать PDF файл с конца.

PDF публикация может быть описана как иерархия объектов, содержащихся в «теле» PDF файла. Структура PDF публикации или документа представлена на рисунке 6.

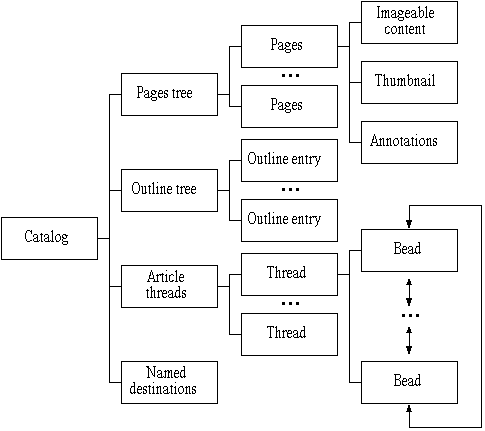


Рисунок 6 - Структура публикации в формате PDF

Основными объектами в этой иерархии являются таблицы dictionary. Каждая страница документа включает ссылки к своим изображениям, миниатюрам и комментариям, которые появляются на странице.

Каталог *(Catalog)* – объект типа dictionary, являющийся корневым узлом документа. Он содержит ссылки на «дерево» страниц в документе, ссылку на дерево объектов, представляющих схему документа (*bookmarks*, или *outline*), ссылки на статьи и список *named destinations*.

Доступ к страницам документа открывается через дерево узлов, названное – деревом страниц *(Pages tree)*. Это дерево определяет порядок страниц в документе.

Страницы (pages) являются объектами типа dictionary, ключи которого описывают текст, содержащийся на одной странице и изображения.

PDF документ может включать миниатюрные схемы страниц. Миниатюра *(Thumbnail)* задается значением ключа *Thumb* объекта страницы.

Комментарии *(Annotations)* – это заметки или другие объекты, которые связаны со страницей, но описываются отдельно от дескриптора страницы.

Дерево закладок (bookmarks tree) предоставляет пользователю возможность иметь доступ к различным видам публикации по имени.

Место назначения *(Destinations)*: комментарии и закладки могут определять место назначения, которое состоит из страницы, места на странице, и масштаба показа страницы.

Дерево имен (Name tree) похоже на дерево страниц, но «листья» содержат пары string (имен) и объектов. Такое дерево применяется для организации поименованных мест назначения. Оно состоит из узлов трех видов: корень, промежуточное, листья

Статьи (article threads): публикация может включать несколько статей (article thread), каждая из которых, в свою очередь, может содержать несколько фрагментов.

Формы (Acrobat Forms) в Acrobat состоят из полей. Основные свойства полей: тип, имя, значение.

Спецификация файла (File specification) определяет место расположения файла на диске. Не зависит от операционной системы.

Шрифт (Font)представлен в PDF формате как объект типа dictionary, определяющий тип, имя, кодировку, и другую информацию.

*XObjects* – поименованные ресурсы. PDF поддерживает три типа таких объектов: *Images*, *Forms* и *PostScript language fragments*.

Form XObjects – это описание любого текста, графики или простых изображений, которые могут многократно воспроизводиться на различных страницах или в разных местах одной страницы. Также представлено в виде stream.

Графическое состояние (Graphics state) – используется для описания графических параметров объекта. Визуализация графического объекта определяется такими параметрами, как: толщина линии, шрифт и другими.

*Special Graphics State* содержит параметры, которые применяются ко всем типам графических объектов. PDF обеспечивает создание *graphics state stack* для сохранения и восстановления состояний (оператор *q*).

*Special Graphics State operators*: *q* – сохранение текущего состояния в стеке состояний, *Q* – восстановление графического состояния из стека, *concat* – модифицирует CTM *(Current Transformations Matrix)*.

Контуры (Path) используются для представления линий, кривых и областей. Сегментами могут быть прямые линии или кривые (кривые Безье). Операторы: *MoveTo*, *LineTo* – добавляет линию к контуру, *CurveTo* – добавляет кривую к контуру, *closepath* – «закрывает» контур и добавляет линию из текущей точки к началу контура, *fill* – заполняет контур и др.

*PDF text object* состоит из операторов, которые определяют символьные строки, перемещения текущей точки, и «текстовые состояния».

*Text string operators* – эти операторы прорисовывают текст на странице [4,5].

## Структура RTF

RTF (Rich Text Format) - это проприетарный формат файла документа с опубликованной спецификацией, разработанный Microsoft Corporation с 1987 по 2008 год для межплатформенного обмена документами с продуктами Microsoft. До 2008 года Microsoft опубликовала обновленные спецификации для RTF с основными редакциями версий Microsoft Word и Office. Формат RTF применяется для представления электронных изданий, созданных в среде MS Office и приложений операционной системы Windows. Обычно он выполняет функции обменного формата.

RTF — это классический контейнер с вложенными друг в друга контейнерами помельче. Пример обобщенной структуры файла RTF приведён на рисунке 7. Каждый такой контейнер обычно начинается с управляющего слова или символа, предваряемого знаком «обратный слэш». Определен фирмой Microsoft как стандартный формат для обмена текстовыми документами.

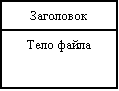


Рисунок 7 – Обобщенная структура RTF

В состав заголовка файла входит идентификатор RTF файла, команда выбора гарнитуры, команда выбора шрифта по умолчанию и таблица доступных шрифтов, таблица цветов (RGB) и таблица стилей.

В теле файла размещаются данные публикации в виде двоичных кодов ASCII. В формате RTF используются только символы, представимые стандартом ASCII, а также MAC- и PC- символьного набора. Кроме текста, файл в RTF-формате содержит команды управления, которые также реализованы с использованием этого стандарта.

Иногда формат RTF называют языком разметки текста. Это связано с тем, что этот формат содержит операторы, управляющие слова и символы и другие элементы описания размещения текста и графики.

Команды управления разделяются на управляющие слова (control words) и управляющие символы (control symbols).

Управляющее слово представляет собой последовательность символов с разделителем (delimiter) в конце:

\набор\_символов <разделитель>

Перед управляющим словом вводится обратная косая черта “\” (backslash). В качестве разделителей могут использоваться следующие символы:

* пробел (space), причем этот символ относится к управляющему слову;
* цифра или знак “-“;
* все символы, отличные от цифр и букв (эти символы не относятся к управляющему слову).

В формате RTF для задания управляющей последовательности символов используются буквы от “А” до “Z” и от “а” до “z”, а также цифры от “0” до “9”. Национальные символы к управляющей информации не относятся.

В качестве управляющих символов используются отдельные буквы. Перед каждым управляющим символом вводится обратная косая черта “\”:

В настоящее время определены только некоторые из этих символов. Поэтому при чтении неизвестные символы могут быть пропущены.

В формате RTF существует возможность объединять отдельные последовательности в группы при помощи скобок: “{” – начало группы, “}” – конец группы.

Такие группы создаются, например, при описании сносок, колонтитулов и так далее. Если символы “\”, “{“ или “}” размещаются внутри обычного текста, то перед ними ставится обратная косая черта.

Это позволяет программе, читающей RTF-последовательность, распознать, что символ не следует интерпретировать как управляющий.

В составе формата RTF имеются операторы для задания начертания и способа выделения символов при создании редакторских примечаний; форматирование секции *(section formatting);*команды задания расстояния между строк; операторы нумерации строк; операторы задания атрибутов колонтитулов; команды выравнивания; операторы форматирование абзаца *(paragraph formatting)*; установки параметров отступа первой строки абзаца *(first line indent)*; операторы задания атрибутов рамки вокруг абзаца; операторы форматирования таблиц и др.

Набор операторов рассматриваемого формата ориентирован на описание сложного текстового материала, включающего иллюстрации, таблицы, диаграммы и другие элементы печатных изданий [6].

## Требования к данным БД

На вход ПО должны подаваться данные шаблонного вида. На рисунках 8, 9 изображен стандартный вид обрабатываемых документов типа «Учебный план».



Рисунок 8 –Титульный лист учебного плана

На титульном листе учебного плана хранятся следующие данные: название направления, профиля подготовки, выпускающей кафедры, срока и периода обучения, виды реализуемой профессиональной деятельности выпускника, а также календарный график. Календарный график помогает отслеживать сроки обучения внутри учебного года и определять сессионные периоды.

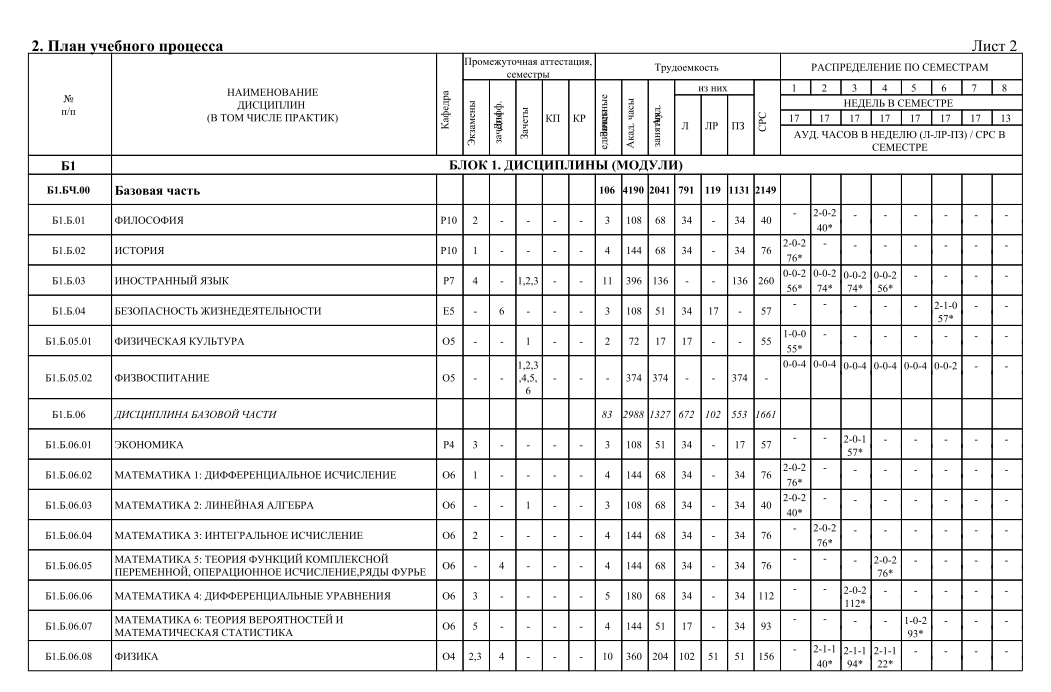


Рисунок 9 – Страница учебного плана, содержащая таблицу

На последующих листах учебного плана хранятся название реализуемых дисциплин, обеспечивающая кафедра (важно отличать обеспечивающая кафедру от выпускающей), форма аттестации по каждой из дисциплин (это может быть экзамен, зачет, зачет с оценкой, и дополнительно курсовая работа и курсовой проект).

Трудоемкость дисциплины показывает формат преподавания дисциплины (есть ли лекционные, практические и лабораторные занятия), нагрузку (как в зачетных единицах, так и аудиторной нагрузке).

Важно отметить, что многие дисциплины преподаются несколько семестров, причем это не обязательно последовательные семестры. Данные сущности хранятся вместе, так как в последствии в отчетных документах эти данные суммируются.

На рисунке 10 изображен стандартный вид документов типа «Экзаменационная ведомость».

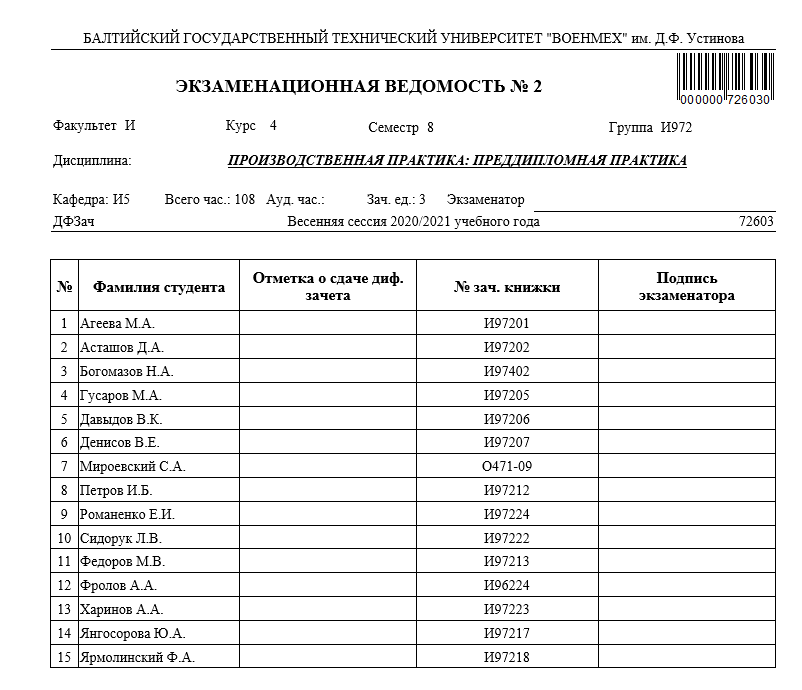


Рисунок 10 – Экзаменационная ведомость

В экзаменационных и зачетных ведомостях указываются пофамильные списки студентов, номера зачетных книжек, номер группы, обеспечивающая кафедра, название дисциплины, сессионный период, форма аттестации, курс и семестр, а также служебные поля для обратной синхронизации данных в ИС УМУ.

Служебные поля содержат идентификаторы дисциплины и номера ведомости и печатаются белым текстом над таблицей.

На рисунке 11 представлена схема используемой в ИС «Личный кабинет преподавателя» базы данных.

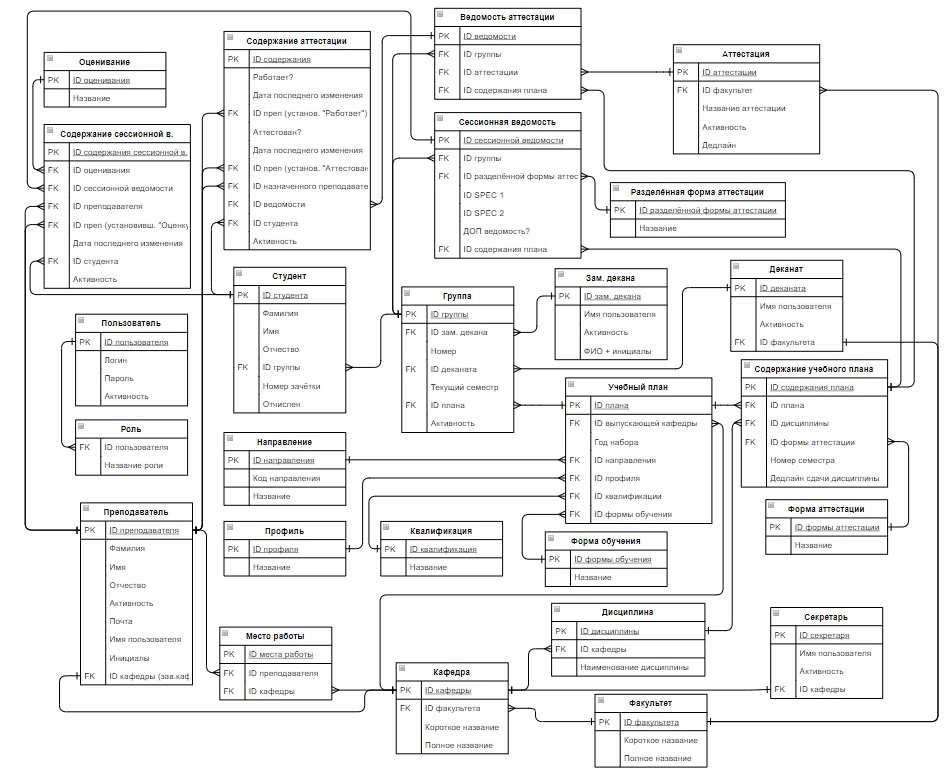


Рисунок 11 – Схема БД

ПО импортирует данные в следующие таблицы:

* учебный план,
* содержание учебного плана,
* направление,
* профиль,
* дисциплина,
* сессионная ведомость,
* содержание сессионной ведомости.

В таблице 1 приведена спецификация полей таблиц базы данных.

Таблица 1 – Требования к данным БД

|  |  |  |
| --- | --- | --- |
| Название таблицы | Поля таблицы | Требования к импортируемым данным |
| Учебный план | id выпускающей кафедры | Существующий id из таблицы «Кафедра» |
| Год набора | Целое 4-значное число |
| id направления | Существующий id из таблицы «Направление» |
| id профиля | Существующий id из таблицы «Профиль» |
| id квалификации | Существующий id из таблицы «Квалификация» |
| id формы обучения | Существующий id из таблицы «Форма обучения» |
| Содержание учебного плана | id учебного плана | Существующий id из таблицы «Учебный план» |
| id дисциплины | Существующий id из таблицы «Дисциплина» |
| id формы аттестации | Существующий id из таблицы «Форма аттестации» |
| Номер семестра | Целое число, обозначающее номер семестра |
| Направление | Код направления | Строка, состоящая из цифр и точек |
| Название | Строка, содержащая название направления |
| Дисциплина | id кафедры | Существующий id из таблицы «Кафедра» |
| Наименование дисциплины | Строка, содержащая название дисциплины |

Продолжение таблицы 1

|  |  |  |
| --- | --- | --- |
| Название таблицы | Поля таблицы | Требования к импортируемым данным |
| Направление | Код направления | Строка, состоящая из цифр и точек |
| Название | Строка, содержащая название направления |
| Дисциплина | id кафедры | Существующий id из таблицы «Кафедра» |
| Наименование дисциплины | Строка, содержащая название дисциплины |
| Сессионная ведомость | id группы | Существующий id из таблицы «Группа» |
| id формы аттестации | Существующий id из таблицы «Форма аттестации» |
| Специальный id ведомости №1 | Целое число |
| Специальный id ведомости №2 | Целое число |
| id содержания учебного плана | Существующий id из таблицы «Содержание учебного плана» |
| Содержание сессионной ведомости | id сессионной ведомости | Существующий id из таблицы «Сессионная ведомость» |
| id студента | Существующий id из таблицы «Студент» |

## Соотнесение данных из документов с полями БД

Сущность — это любой однозначно идентифицируемый конкретный или абстрактный объект, включая события и связи между объектами, информация о котором хранится и обрабатывается в базе данных [7].

Для корректной обработки данных сопоставим данные из документов с полями БД ИС «Личный кабинет преподавателя».

### 2.4.1 Направление, профиль, квалификация, форма обучения

На рисунке 12 указаны соответствующие сущностям «Направление», «Профиль», «Квалификация», «Форма обучения» данные из документов типа «Учебная ведомость».

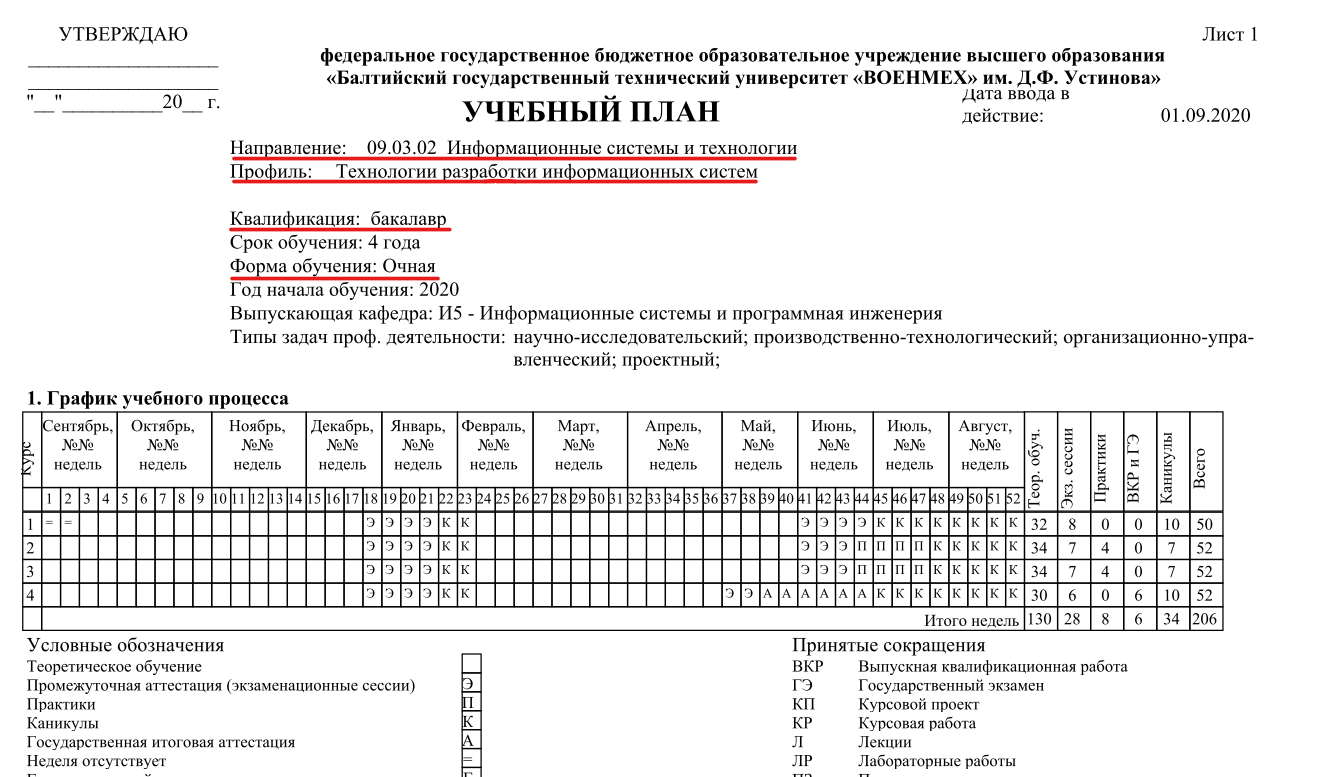


Рисунок 12 – Информация о направлении в учебном плане

### 2.4.2 Дисциплина, форма аттестации, номер семестра

На рисунке 13 указаны соответствующие сущностям «Дисциплина», «Форма аттестации», «номер семестра» данные из документов типа «Учебная ведомость».

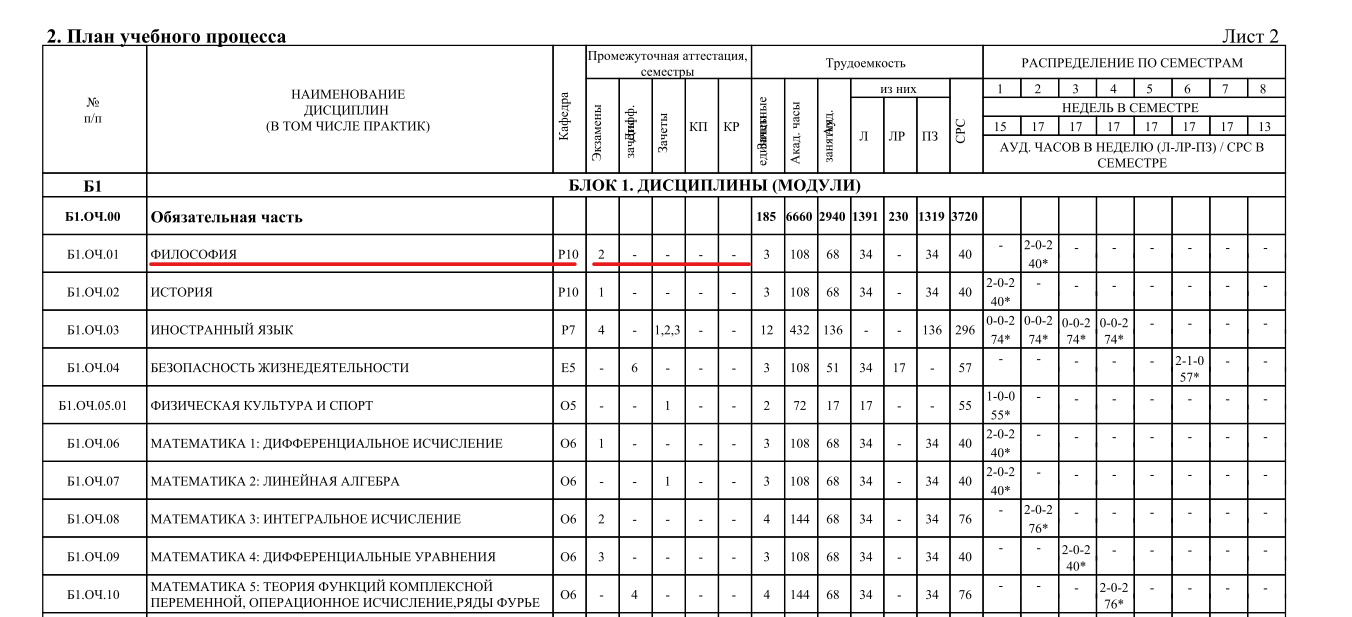


Рисунок 13 – Информация о дисциплине в учебном плане

### 2.4.3 Группа, форма аттестации

На рисунке 14 указаны соответствующие сущностям «Группа», «Форма аттестации» данные из документов типа «Сессионная ведомость».

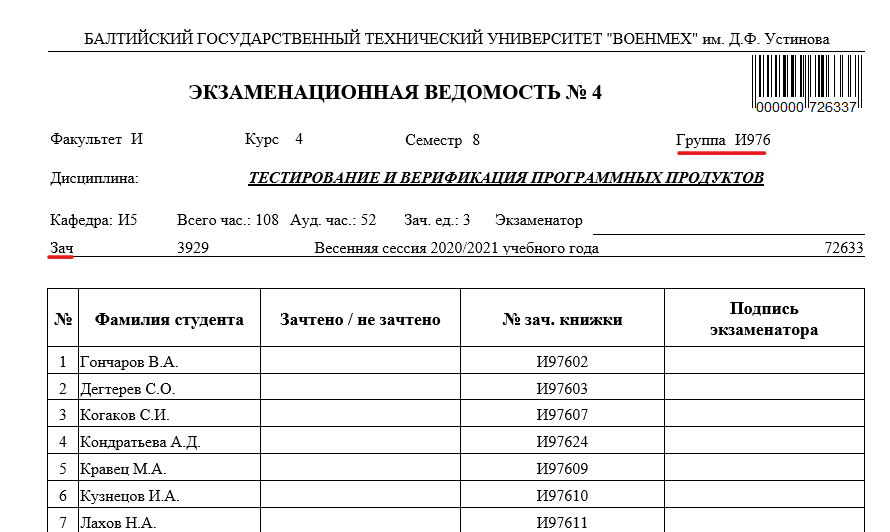


Рисунок 14 – Информация о группе в сессионной ведомости

### 2.4.4 Специальный идентификатор ведомости №1

На рисунке 15 указаны соответствующие сущности «Специальный идентификатор ведомости №1» данные из документов типа «Сессионная ведомость».

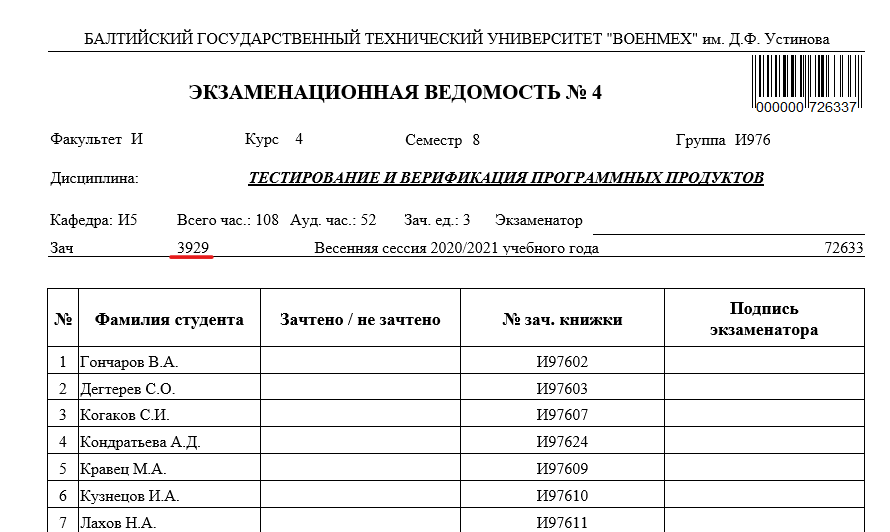


Рисунок 15 – Информация о специальных идентификаторах ведомости в сессионной ведомости

### 2.4.5 Специальный идентификатор ведомости №1

На рисунке 16 указаны соответствующие сущности «Специальный идентификатор ведомости №1» данные из документов типа «Сессионная ведомость».

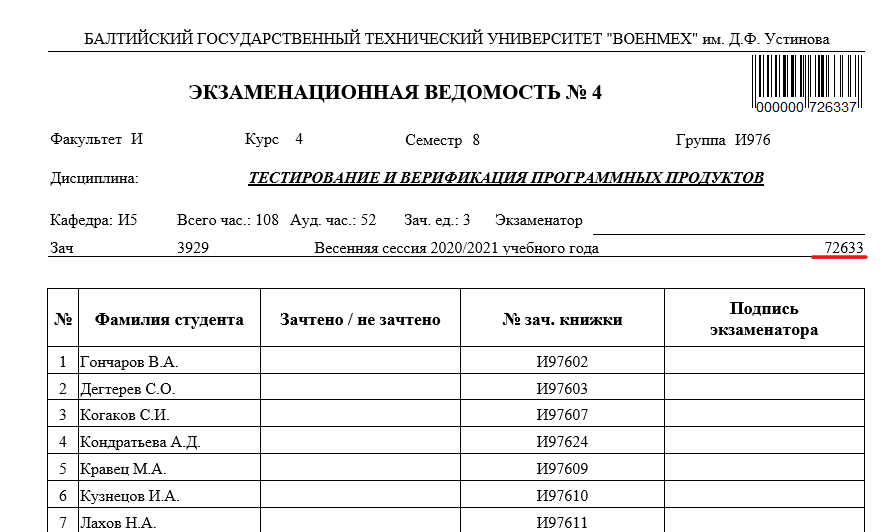


Рисунок 16 – Информация о специальных идентификаторах ведомости в сессионной ведомости

### 2.4.6 Содержание учебного плана

Сущности «Содержание учебного плана» соответствует связка дисциплины и учебного плана, которые извлекаются из документов типа «Учебный план». Из документов типа «Сессионная ведомость» можно извлечь название дисциплины и найти соответствующую запись в таблице «Содержание учебного плана». На рисунке 17 указаны данные соответствующие сущности «Дисциплина».

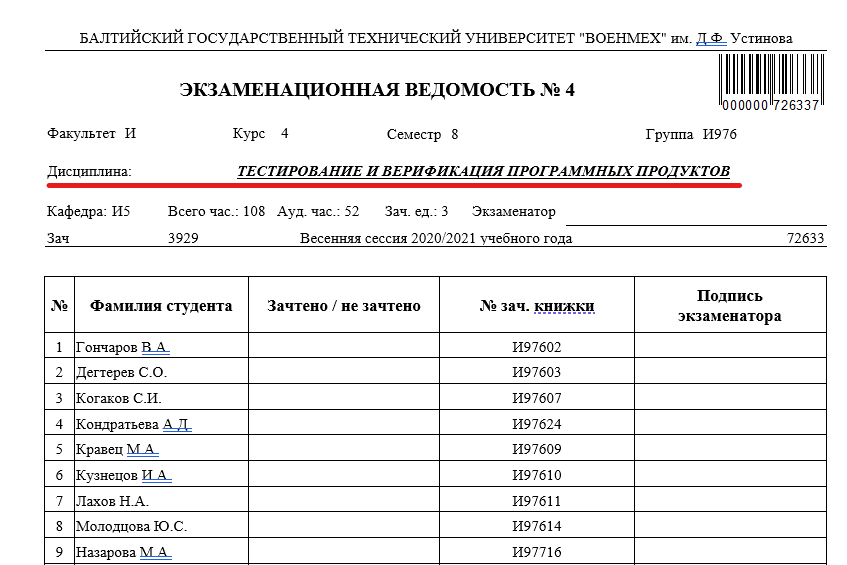


Рисунок 17 – Информация о дисциплине в сессионной ведомости

### 2.4.7 Студент

Данные из таблицы «Студент» (id студента) для таблицы «Содержание сессионной ведомости» находятся в соответствии с данными из документов типа «Сессионная ведомость», которые указаны на рисунке 18.

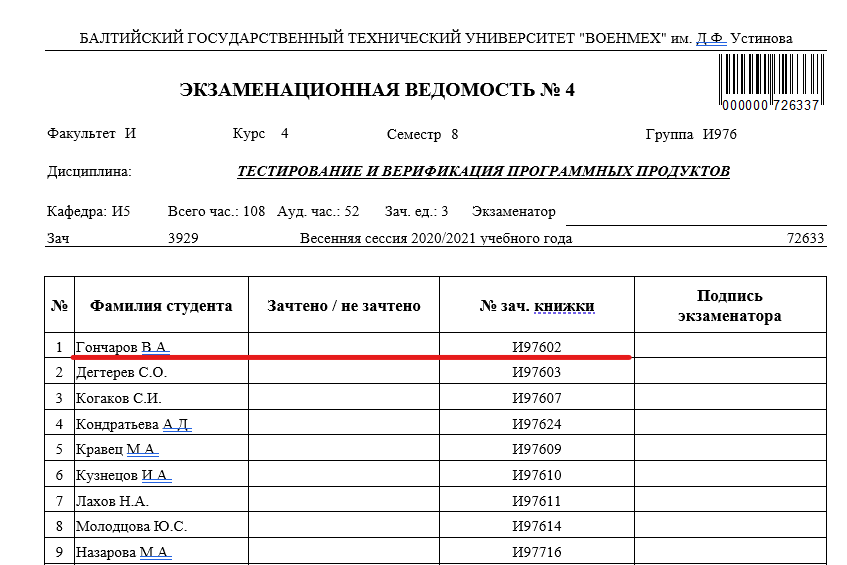


Рисунок 18 – Информация о студентах в сессионной ведомости

## Выбор средств разработки

Для реализации ПО необходимо выбрать средство разработки. Сравним и проанализируем три языка программирования: Python, C++ и С#.

### Язык программирования С++

Язык программирования С++ представляет собой высокоуровневый компилируемый язык программирования общего назначения со статической типизацией, который подходит для создания самых различных приложений [8]. На сегодняшний день С++ является одним из самых популярных и распространенных языков программирования. Для языка написано большое количество учебников и пособий, облегчающих освоение языка. Так же С++ имеет подробную официальную документацию, к которой можно обратиться при возникновении вопросов. Несмотря на это, язык имеет достаточно высокий порог вхождения из-за своей сложности и объёмности. С++ нечасто используется для работы с PDF документами из-за небольшого количества библиотек языка, предназначенных для этого, и их сложности. Таким образом, язык имеет ограниченные и усложненные методы работы с PDF и RTF файлами [9].

### Язык C#

C# – объектно-ориентированный и типобезопасный язык программирования. C# позволяет разработчикам создавать множество типов безопасных и надежных приложений, работающих в экосистеме .NET. C# относится к широко известному семейству языков C, и покажется хорошо знакомым любому, кто работал с C, C++, Java или JavaScript [10]. Язык имеет подробную официальную документацию и множество учебников и пособий в свободном доступе. С# несколько проще в освоении, чем С++. Для языка написано небольшое количество библиотек для работы с PDF и RTF документами, из-за чего возможности C# в данной сфере ограничены [11].

### Язык Python

Python – высокоуровневый объектно-ориентированный язык программирования общего назначения с динамической типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ. Недостатками языка являются зачастую более низкая скорость работы и более высокое потребление памяти написанных на нём программ по сравнению с аналогичным кодом, написанным на Си или C++. В то же время стандартная библиотека включает большой набор полезных функций [12]. Python имеет подробную официальную документацию. Для изучения языка написано большое количество учебников и статей, большую часть которых можно найти в свободном доступе. В отличии от С++ и С# Python достаточно прост в освоении. Язык имеет обширное количество библиотек для работы с документами в формате PDF и RTF [13].

### Выбор языка программирования

По результатам сравнения трёх языков программирования (Таблица №2), можно сделать вывод, что для реализации поставленных задач наиболее подходящим выбором будет Python. Язык прост в освоении и использовании и, что самое важное, имеет большой выбор средств для работы с PDF и RTF.

Таблица 2 – Сравнение языков программирования

|  |  |  |  |
| --- | --- | --- | --- |
| Язык программирования | Готовые библиотеки для работы с PDF | Готовые библиотеки для работы с RTF | Готовые библиотеки для работы с БД |
| C++ | PoDoFo;  Hummus PDF; MuPDF | librtf | libpq |
| C# | iTextSharp;  PdfPig;  PDF Clown | RichTextBox (встроенный класс);  .NET RTF Writer Library | Npgsql |
| Python | PyPDF2;  PDFMiner;  PDFQuery;  tabula-py;  pdflib;  PyMuPDF;  pdfrw;  Camelot | PyRTF;  docx;  rtfwriter | Psycopg2;  pg8000;  py-postgreql;  PyGreSQL;  ocpgdb;  bpsql;  SQLAlchemy |

# Программная реализация

Алгоритм — конечный набор предписаний для получения решения задачи посредством конечного количества операций [14].

## Модуль обработки PDF

Алгоритм работы модуля обработки PDF представлен на рисунках 19, 20.

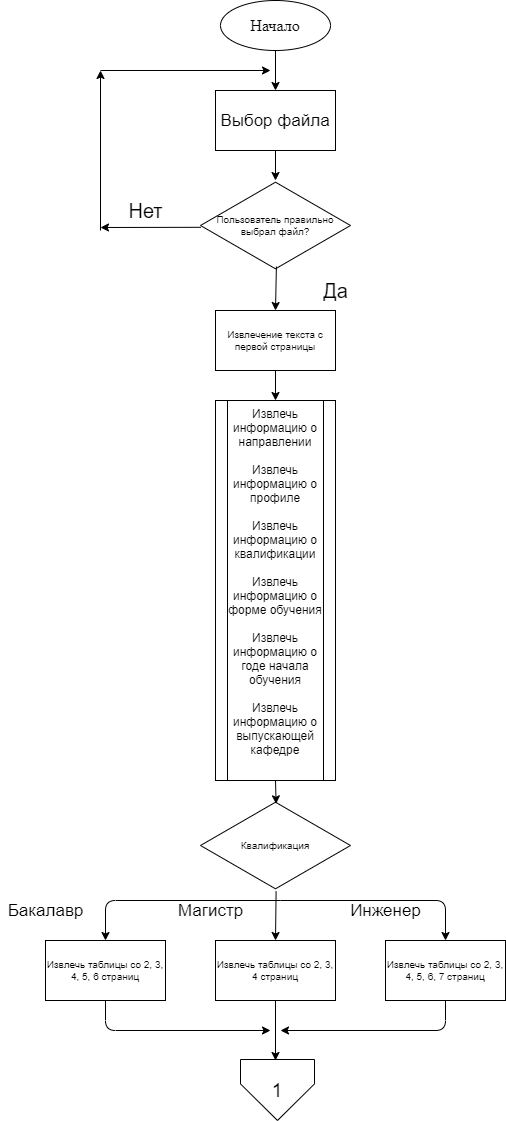


Рисунок 19 – Алгоритм работы модуля обработки PDF

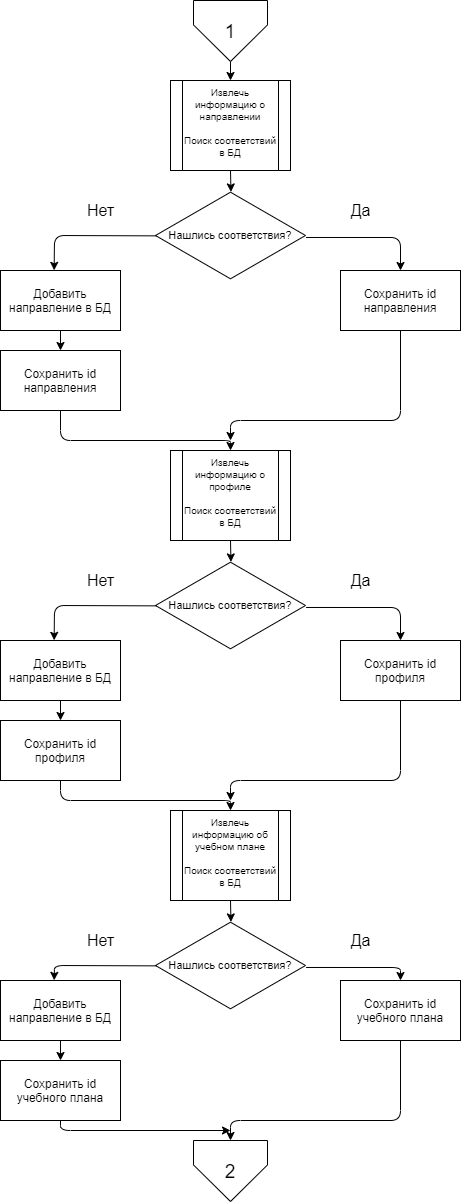


Рисунок 20 – Алгоритм работы модуля обработки PDF

На рисунке 21 изображено продолжение алгоритма работы модуля.

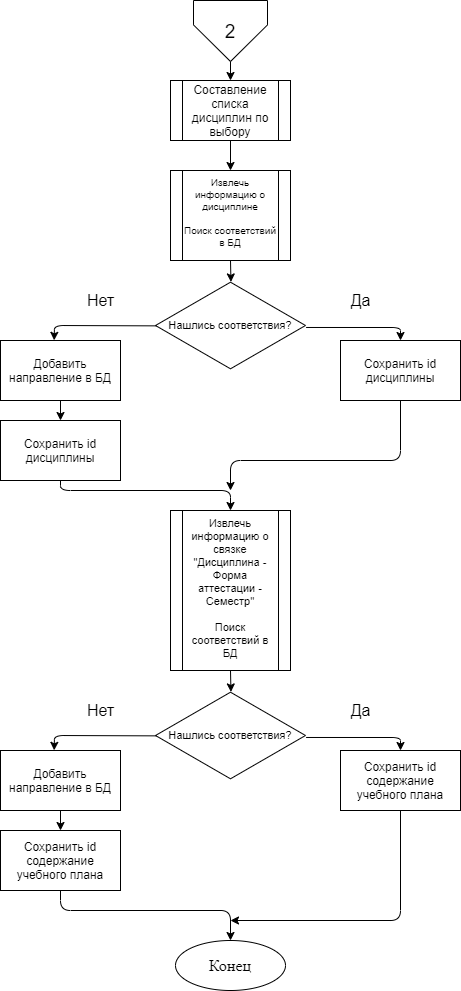


Рисунок 21 – Алгоритм работы модуля обработки PDF

Фрагмент кода, отвечающий за получение данных:

page = doc.loadPage(0)

page\_text = page.getText("text")

a = page\_text.find("Направление")

b = page\_text.find("организационно")

if a == -1:

a = page\_text.find("Квалификация:")

b = page\_text.find("Виды проф. деятельности:")

print(page\_text)

print("a = ", a, " b = ", b)

data = page\_text[a:b]

book = openpyxl.Workbook()

sheet1 = book.create\_sheet('list 1', 0)

i = data.find('0')

direction = data[i:i+8]

sheet1['B1'] = direction

m = data.find('Магистерская')

if (m==-1):

b = data.find('Профиль')

j = b

sheet1['A2'] = 'Профиль'

else:

sheet1['A2'] = "Магистерская программа"

name\_direction = data[i+9:j]

a = data.find('Квалификация')

b = data.find('Форма')

qualification = data[a+13:b]

sheet1['B3'] = qualification

## Модуль обработки RTF

Алгоритм работы модуля обработки RTF представлен на рисунках 22, 23.

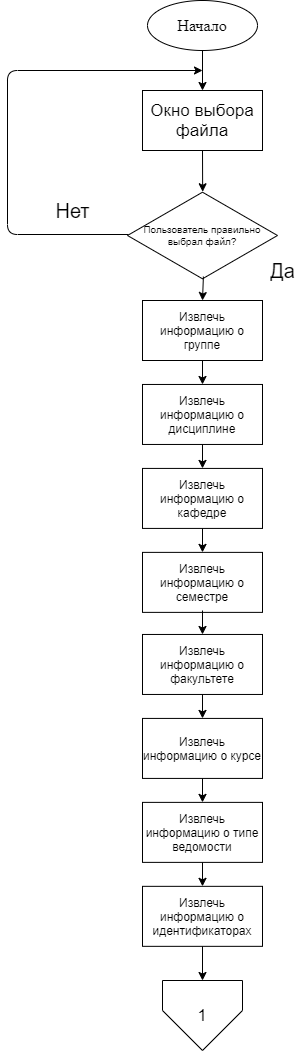


Рисунок 22 – Алгоритм работы модуля обработки RTF

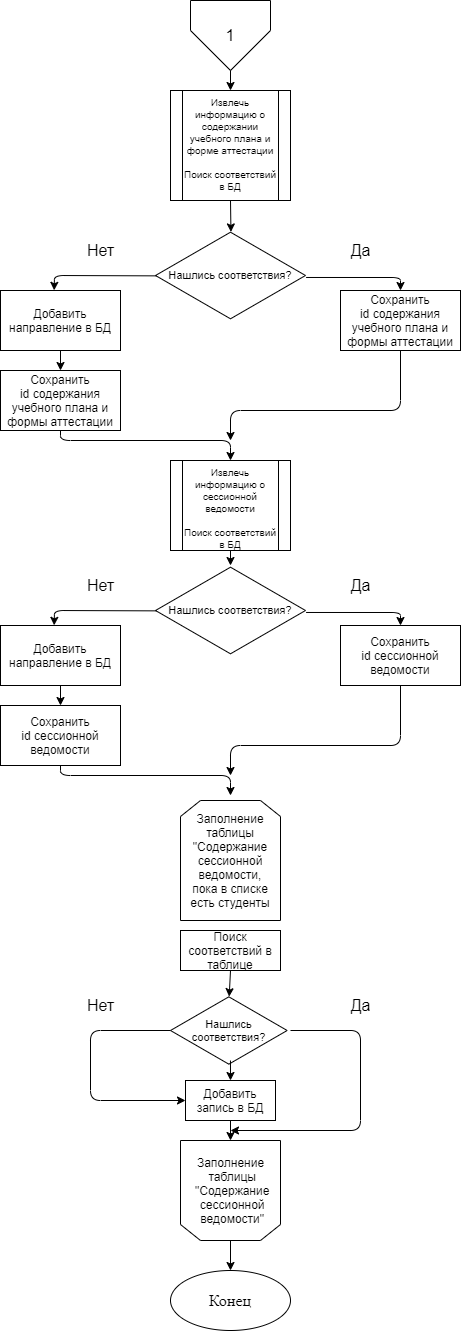


Рисунок 23 – Алгоритм работы модуля обработки RTF

Фрагмент кода, отвечающий за получение данных:

table = document.tables[0]

discipline = table.rows[5].cells[4].text.strip()

semester = table.rows[4].cells[17].text.strip()

kaf = table.rows[6].cells[4].text.strip()

grname = str(table.rows[4].cells[27].text.strip())

course = table.rows[4].cells[12].text.strip()

faculty = table.rows[4].cells[7].text.strip()

vedid = table.rows[7].cells[28].text.strip()

vedid2 = table.rows[7].cells[6].text.strip()

vedtype = table.rows[7].cells[0].text.strip()

period = table.rows[7].cells[14].text.strip()

dateexam = table.rows[len(table.rows) - 9].cells[7].text.strip()

student = table.rows[10].cells[15].text.strip()

## Работа с SQL

Во время работы ПО производит SQL-запросы в базу данных для поиска и добавления записей [15]. В программе используются запросы типа SELECT и INSERT.

Примеры использующихся запросов:

"SELECT syllabus\_id FROM syllabus WHERE year = %s AND department\_id = %s AND direction\_id = %s AND profile\_id = %s AND form\_of\_training\_id = %s AND qualification\_id = %s ", (year, department\_id, direction\_id, profile\_id, form\_of\_training\_id, qualification\_id )

"INSERT INTO syllabus (syllabus\_id, year, department\_id, direction\_id, profile\_id, qualification\_id, form\_of\_training\_id ) VALUES (%s, %s, %s, %s, %s, %s, %s)", (syllabus\_id, year,department\_id, direction\_id, profile\_id, qualification\_id, form\_of\_training\_id )

## Результат работы ПО

На рисунках 24, 25 приведены результаты работы ПО с таблицами «Учебный план» и «Содержание учебного плана».

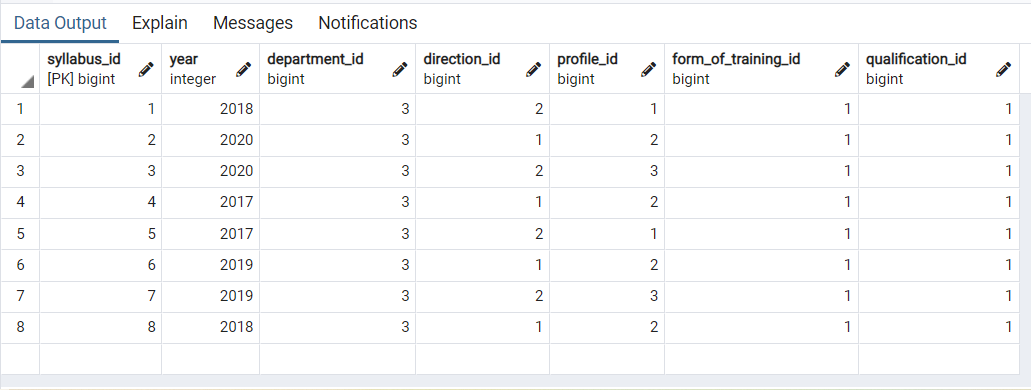


Рисунок 24 – Содержание таблицы «Учебный план»

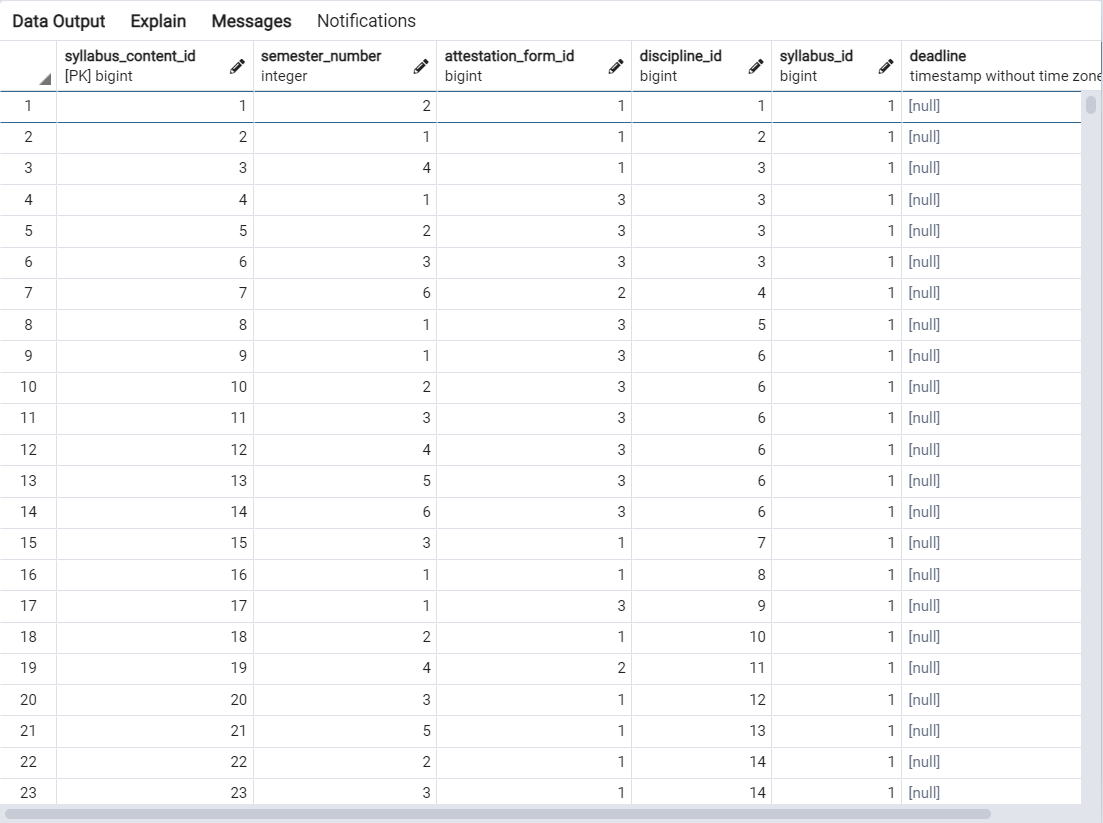


Рисунок 25 – Содержание таблицы «Содержание учебного плана»

На рисунках 26, 27 изображены результаты работы ПО с таблицами «Дисциплина» и «Профиль».



Рисунок 26 – Содержание таблицы «Дисциплина»

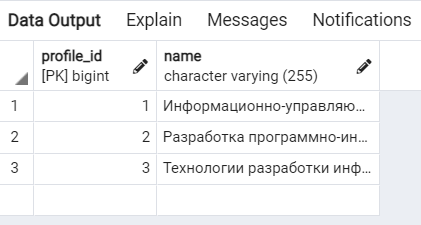


Рисунок 27 – Содержание таблицы «Профиль»

На рисунках 28, 29, 30 изображены результаты работы ПО с таблицами «Направление», «Сессионная ведомость» и «Содержание сессионной ведомости».

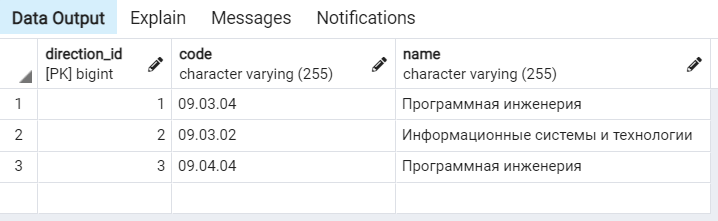


Рисунок 28 – Содержание таблицы «Направление»



Рисунок 29 – Содержание таблицы «Сессионная ведомость»

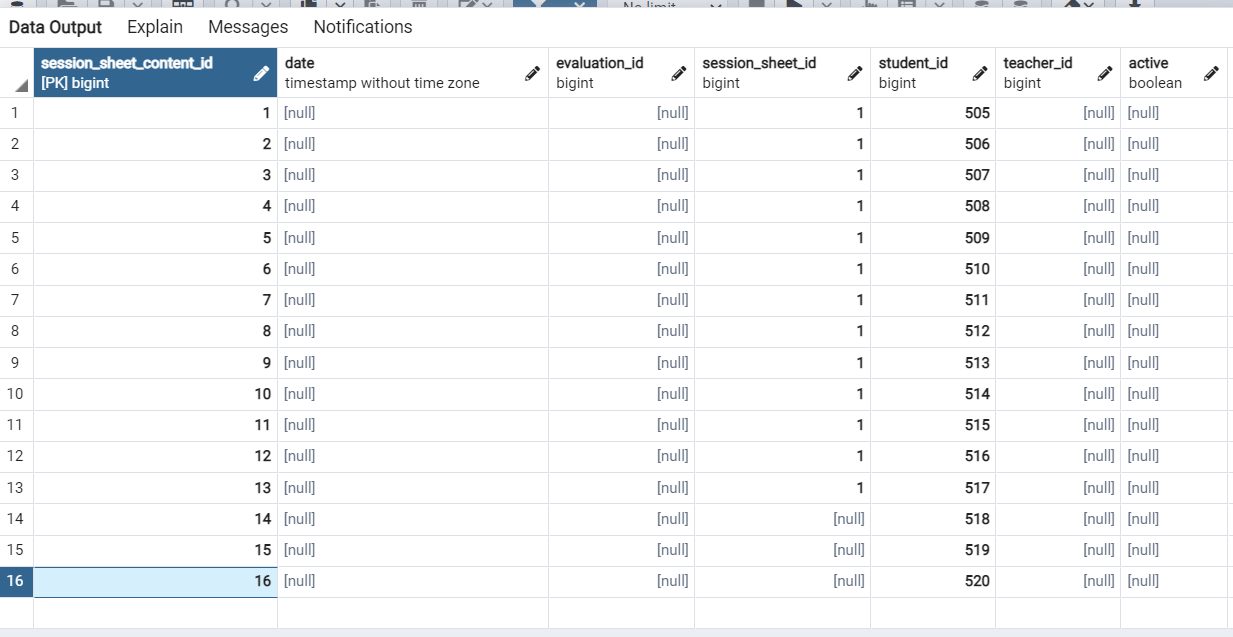


Рисунок 30 – Содержание таблицы «Содержание сессионной ведомости»

ЗАКЛЮЧЕНИЕ

В рамках данной выпускной квалификационной работы было разработано программное обеспечение для импорта данных в базу данных информационной системы.

В первом разделе был проведён анализ предметной области. Были рассмотрены процессы электронного документооборота и документооборота вуза. Были проанализированы способы синхронизации с ИС.

Во втором разделе были рассмотрены общая структура файлов формата PDF и RTF и шаблоны обрабатываемых документов. Были обозначены требования к данным БД. Описаны требования к соответствию данных. Был обоснован выбор средств разработки.

В третьем разделе представлены скриншоты результата работы ПО.

Итогом данной выпускной квалификационной работы является готовый программный продукт.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ Р 53898-2013 Системы электронного документооборота [Электронный ресурс]. – URL: <https://docs.cntd.ru/document/1200108037> (дата обращения: 25.05.2021)
2. Положение о текущем рубежном контроле успеваемости и промежуточной аттестации обучающихся [Электронный ресурс]. – URL: <https://voenmeh.ru/images/docs/uch_upr/polozh_o_tekushem_kontrole_uspevaemosti.pdf> (дата обращения: 25.05.2021)
3. Приказ о внесении изменений в Положение о текущем рубежном контроле успеваемости и промежуточной аттестации обучающихся [Электронный ресурс]. – URL: <https://voenmeh.ru/images/docs/otdel-kachestvaobraz/SMK/prikaz_80-O_05032021_izm_v_polozh.pdf> (дата обращения: 25.05.2021)
4. Подробная информация о внутренней структуре PDF [Электронный ресурс]. – URL: https://opensource.adobe.com/dc-acrobat-sdk-docs/ (дата обращения: 25.05.2021)
5. Спецификация формата PDF 1.7 [Электронный ресурс]. – URL: <https://wwwimages2.adobe.com/content/dam/acom/en/devnet/pdf/pdfs/pdf_reference_1-7.pdf> (дата обращения: 25.05.2021)
6. Спецификация формата RTF 1.9.1 [Электронный ресурс]. – URL: <https://web.archive.org/web/20190708132914/http://www.kleinlercher.at/tools/Windows_Protocols/Word2007RTFSpec9.pdf> (дата обращения: 25.05.2021)
7. Конноллн, Томас, Бегг, Карелии. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. 3-е издание: Пер. с англ. – М. : Издательский дом "Вильяме", 2003. — 1440 с.
8. Официальный сайт языка программирования С++ [Электронный ресурс]. – URL: <https://isocpp.org/> (дата обращения: 25.05.2021)
9. Стауструп, Бьярне. Программирование: принципы и практика использования С++: Пер. с англ. – М.: ООО «И. Д. Вильямс», 2011. – 1248 с.
10. Документация языка программирования C# [Электронный ресурс]. – URL: https://docs.microsoft.com/ru-ru/dotnet/csharp/ (дата обращения: 25.05.2021)
11. Албахари, Джозеф, Албахари, Бен. С# 7.0 Справочник. Полное описание языка. – Спб. : ООО «Альфа-книга», 2017. – 1507 с.
12. Официальный сайт языка программирования Python [Электронный ресурс]. – URL: <https://www.python.org/> (дата обращения: 25.05.2021)
13. Стандартная библиотека Python 3: справочник с примерами, 2-е изд. Пер. с англ. – СПб. : ООО “Диалектика”, 2019. – 1376 с.
14. Дж. Макконнелл. Основы современных алгоритмов. 2-е дополненное издание Москва: Техносфера, 2004. – 368с.
15. Документация к PostgreSQL 13.3 [Электронный ресурс]. – URL: <https://postgrespro.ru/docs/postgresql/13/index> (дата обращения: 25.05.2021)

ПРИЛОЖЕНИЕ А

### Акт о внедрении