

Разработка программного комплекса для автоматизации формирования документов основных профессиональных образовательных программ высшего образования факультета компьютерных наук Воронежского государственного университета

А. И. Претков, email: pretkovartyom@gmail.com¹

Е. А. Копытина, email: zhemkaterina@yandex.ru¹

¹ Воронежский государственный университет

***Аннотация.** В данной статье предложена реализация программного комплекса для генерации документов, описывающих основные профессиональные образовательные программы высшего образования факультета компьютерных наук Воронежского государственного университета, на основе информации импортируемой из учебных планов, реализуемых на факультете.*

***Ключевые слова:** электронные документы, основная профессиональная образовательная программа высшего образования, учебные планы, автоматизация рутинных задач, прикладная задача.*

Введение

Большое количество электронных документов необходимо периодически актуализировать, однако, несмотря на большое количество функций для поиска и замены текста в документе эта задача является сложной и рутинной, особенно когда редактируемый документ связан с другим с документом и необходимо найти отличия между ними и только после этого внести необходимые изменения.

Данные действия требуют внимательности, большого количества времени и чреваты ошибками.

Данная проблема крайне актуальна в настоящее время и её решению посвящены работы [1-8]. Особенно вышеуказанная проблема характерна для высших учебных заведений: периодически появляются новые учебные стандарты и, соответственно, обновляется учебный план. В дальнейшем на основе измененного учебного плана нужно вносить изменения в документ с информацией об образовательной программе. Данный документ сокращенно называется «ОПОП».

Для устранения вышеописанных сложностей был разработан программный комплекс, формирующий Word-документ с информацией об образовательной программе на основе Excel-файла с учебным

планом, предыдущей версии Word-документа образовательной программы и текстового файла со списком компетенций и их названием.

1. Постановка задачи

К описываемому в статье программному комплексу для автоматизации формирования документов основных профессиональных образовательных программ высшего образования были выдвинуты следующие функциональные требования:

- Чтение информации об учебном плане из Excel-таблиц;
- Генерация Word-документа ОПОП на основе данных из учебного плана.

2. Средства реализации программного комплекса

Проект реализуется посредством чтения Excel-таблиц, формирования на их основе ОПОП документа, подстановкой редко изменяемой информации из предыдущей версии ранее созданного документа ОПОП, и копированием его форматирования в генерируемый.

Проект реализован на нескольких языках, поскольку их библиотеки для работы с офисными документами в некоторых моментах серьезно различаются, особенно в доступных настройках для форматирования документа.

В процессе работы были использованы следующие технологии:

- Java 8. Данный объектно-ориентированный язык программирования обладает готовыми библиотеками для решения многих прикладных задач и является строго типизированным, что облегчает написание кода и поиск ошибок. Использовался: для парсинга Excel-таблиц; для создания нового документа на основе данных полученных из Excel таблиц;
- Python 3.10. Мощный объектно-ориентированный язык программирования, на котором написано множество библиотек. На этом языке были написаны утилиты для копирования редко изменяемой информации вместе с форматированием из уже созданного документа ОПОП в новый документ;
- Idea Professional. Среда разработки для эффективного программирования Java-приложений с мощным отладчиком и подсказками при написании кода.
- PyCharm Professional. Среда разработки языка Python, функционально практически аналогична Idea за исключением ориентации на язык Python.

3. Реализация логики программного комплекса

Часть программного комплекса написана на языке Java. Для её реализации была использована библиотека Apache poi.

Часть программного комплекса считывает Excel-файл и формирует на его основе ОПОП документ.

Программный комплекс состоит из классов.

Для классов по формированию содержания документа выделен отдельный интерфейс ParagraphWriter, содержащий метод writeParagraph, принимающий на вход формируемый Word-документ и Excel-документ с информацией об учебном плане. Для случая, когда недостаточно Excel-файла и нежелательно добавлять логику по получению недостающих данных в класс формирующий итоговый документ присутствует метод для отправки дополнительных данных setData. Все реализации интерфейса ParagraphWriter вынесены в отдельный пакет docWriters.

Опишем реализации этого интерфейса подробнее:

- ThirdParagraphWriter отвечает за формирование общей характеристики основной профессиональной образовательной программы. На основе данных из учебного плана записывается профиль образовательной программы, квалификация, присваиваемая выпускникам, объем программы, срок получения образования и минимальный объем контактной работы по образовательной программе.
- FourthParagraphWriter формирует планируемые результаты освоения ОПОП. На основе файла со списком категорий универсальных компетенций и учебного плана добавляет таблицу с информацией о каждой компетенции и индикаторах ее достижения, также формируется таблица общепрофессиональных компетенций: из учебного плана берутся коды компетенций, их формулировки и индикаторы достижения компетенции и записываются в таблицу, аналогично формируется таблица с профессиональными компетенциями.
- FifthParagraphWriter добавляет таблицу со структурой программы и объемом в зачетных единицах каждого пункта учебной программы и добавляет подробное описание каждого блока учебной программы.
- SixthParagraphWriter добавляет матрицу соответствия компетенций, индикаторы их достижения и элемент ОПОП, приложение с календарным учебным графиком из учебного

плана и приложения с учебными планами за каждый курс обучения.

- EighthDocApplicationWriter записывает аннотации рабочих программ дисциплин и практик на основе данных из учебного плана: общая трудоемкость дисциплины, формируемые компетенции и форма промежуточной аттестации.

Классы для вспомогательной логики вынесены в пакет Utils:

- ParagraphUtil содержит методы для выполнения часто повторяющихся в ходе работе программного комплекса действий, например создание маркированного или нумерованного списка, добавление подпункта в списке, объединение ячеек таблицы, создание таблицы с заданными заголовками, строками и размером.
- ExcelReader содержит методы для работы с Excel-таблицей учебного плана и инкапсулирует в себе логику по получению данных в нужном формате из таблиц.
- FontConstants содержит константы с настройками шрифта, например названия шрифта, стандартный размер шрифта в тексте.
- FontReplacer содержит метод для установки заданного шрифта в параграфах.
- DocumentMarginsSetter содержит метод для установки отступов в документе.

Остальная работа по формированию документа связанная с функциональными возможностями, отсутствующим в общедоступных библиотеках Java, производится Python-скриптами, запускаемыми Java-приложением.

Данные скрипты при вызове могут получать входные параметры. Например, название Excel-листа с учебным планом.

Первый скрипт конвертирует Excel-таблицу с учебным планом в картинку для последующей вставки в Word-документ.

Второй скрипт считывает исходный Word-документ для переноса в формируемый ОПОП документ редко изменяемых данных вместе с их форматированием, например условия осуществления образовательной деятельности, цель и задачи программы. Второй скрипт в силу более сложной логики содержит вспомогательные методы: copy_paragraph – копирует параграф вместе с его форматированием; copy_font – копирует настройки шрифта; copy_style – копирует настройки форматирования; функция iter_block_items позволяет в цикле обойти все элементы документа в правильном порядке; move_table_after возвращает таблицу, добавленную после заданного параграфа; table_insert_paragraph_after и

`insert_paragraph_before` возвращают новый параграф, вставленный перед заданным параграфом; `insert_table_before` возвращает таблицу вставленную перед заданным элементом; `copy_paragraphs` копирует содержимое параграфов из одного документа в другой.

Заключение

В данной статье была рассмотрена реализация программного комплекса для автоматизации формирования документа основной профессиональной образовательной программы высшего образования. На данный момент формируемый ОПОП документ содержит практически всю информацию, за исключением информации о материально-техническом обеспечении. Планируется добавить логику для вставки недостающей информации в итоговый документ.

Список литературы

1. Реализация проекта по созданию реляционной базы данных для Контрольно-счётной палаты Воронежской области / А. Г. Потапов, А. В. Порядин, Т. Д. Братышев [и др.] // Труды молодых ученых факультета компьютерных наук ВГУ. Том Выпуск 2. – Воронеж : ООО «ВЭЛБОРН», 2022. – С. 132-137.

2. Разработка парсера для автоматизации деятельности Контрольно-счётной палаты Воронежской области / А. В. Порядин, А. Г. Потапов, Т. Д. Братышев [и др.] // Труды молодых ученых факультета компьютерных наук ВГУ. Том Выпуск 2. – Воронеж : ООО «ВЭЛБОРН», 2022. – С. 127-131.

3. Вариант решения задачи оптимизации толщины теплоизоляционного слоя в теплоснабжении / Н. А. Петрикеева, Д. М. Чудинов, Л. П. Мышовская // Научный журнал строительства и архитектуры. – 2018. – № 2(50). – С. 21-28.

4. Kopytina, E. A. Smart Service for Medical Care of the Region's Population / E. A. Kopytina, A. V. Kopytin, F. A. Desyatirikov // Proceedings of the 2021 IEEE International Conference "Quality Management, Transport and Information Security, Information Technologies", T and QM and IS 2021, Yaroslavl, 06–10 сентября 2021 года. – Yaroslavl, 2021. – P. 206-209. – DOI 10.1109/ITQMIS53292.2021.9642885.

5. Кропачев, А. Н. Формирование отчета деятельности наркоконтроля по Воронежской области средствами офисных приложений / А. Н. Кропачев, Е. А. Копытина // Труды молодых учёных факультета компьютерных наук ВГУ : Сборник статей / Под редакцией Д.Н. Борисова. Том Выпуск 1. – Воронеж : Общество с ограниченной ответственностью "Вэлборн", 2021. – С. 80-84.

6. Копытин, А. В. О распределении штрафов и вознаграждений в проектах / А. В. Копытин, Д. И. Соломатин // Вестник Воронежского государственного университета. Серия: Системный анализ и информационные технологии. – 2016. – № 4. – С. 119-126.

7. Васянкин, А. А. Модификация понятия «электронный документ» в системе электронного документооборота / А. А. Васянкин, Э. В. Кузьмина // Кайгородовские чтения : Материалы региональной научно-практической конференции, Краснодар, 29 апреля 2010 года / Краснодарский государственный университет культуры и искусств. – Краснодар: Краснодарский государственный университет культуры и искусств, 2010. – С. 75-79.

8. Application of game theory when taking decisions and choosing the best variant / N. Petrikeevea, Y. Kopytina, G. Martynenko, V. Kozlova // E3S Web of Conferences : Topical Problems of Green Architecture, Civil and Environmental Engineering, TPACEE 2019, Moscow, 20–22 ноября 2019 года. Vol. 164. – Moscow: EDP Sciences, 2020. – P. 08027. – DOI 10.1051/e3sconf/202016408027.