Электронный научный журнал "Математическое моделирование, компьютерный и натурный эксперимент в естественных науках" http://mathmod.esrae.ru/
URL статьи: **mathmod.esrae.ru/20-81**
Ссылка для цитирования этой статьи:
Блинков Ю.А., Панкратов И.А. Документо-ориентированное хранение и обработка научных публикаций // Математическое моделирование, компьютерный и натурный эксперимент в естественных науках. 2018. №4

УДК 004.65

ДОКУМЕНТО-ОРИЕНТИРОВАННОЕ ХРАНЕНИЕ И ОБРАБОТКА НАУЧНЫХ ПУБЛИКАЦИЙ

Блинков Ю.А.[1], Панкратов И.А.[2]
[1]Саратовский национальный исследовательский государственный университет имени Н.Г. Чернышевского, Россия, Саратов, BlinkovYuA@info.sgu.ru
[2]Саратовский национальный исследовательский государственный университет имени Н.Г. Чернышевского, Россия, Саратов, PankratovIA@info.sgu.ru

DOCUMENT-ORIENTED STORAGE AND PROCESSING OF SCIENTIFIC PUBLICATIONS

Blinkov Yu.A.[1], Pankratov I.A.[2]
[1]Saratov State University, Russia, Saratov, BlinkovYuA@info.sgu.ru
[2]Saratov State University, Russia, Saratov, PankratovIA@info.sgu.ru

Аннотация. В статье представлена информационная система для хранения и обработки информации о научных публикациях. Информация о научных публикациях хранится в нереляционной базе данных MongoDB. Показаны преимущества использования технологии NoSQL. Работа является развитием [1-4].

Ключевые слова: нереляционная база данных, «паук», NoSQL.

Abstract. In this paper we present the information system for storing and processing information about scientific publications. Information about scientific publications is stored in the non-relational database MongoDB. The advantages of using NoSQL technology are shown. The investigation is a development of [1-4].

Keywords: non-relational database, web-crawler, NoSQL.

В настоящей работе рассмотрено хранение библиографической информации с помощью технологии NoSQL [5], набирающей в последнее время популярность. При использовании NoSQL не требуется создавать несколько связанных друг с другом таблиц. Данные о таких разных публикациях, как статьи, книги, учебные пособия, тезисы и материалы конференций хранятся в одной коллекции (collection, аналог таблицы в реляционных базах данных (БД)). В общем случае у записей, которые соответствуют публикациям разных типов, поля не совпадают. Например, для

статьи требуется хранить наименование журнала, в котором она была опубликована, а для учебного пособия – наименование издательства и т.д. При этом некоторые поля у записей одинаковы (авторы, год издания и т. д.). Традиционный подход (использование реляционных БД) привёл бы к созданию отдельных таблиц для публикаций каждого вида. Напротив, применение технологии NoSQL позволило отказаться от жёсткой структуры БД.

Для наполнения вышеуказанной БД был написан «паук» (web crawler) на языке Python. Информация о публикациях извлекается с сайта национальной библиографической базы данных научного цитирования (РИНЦ, http://elibrary.ru). Разработанное программное обеспечение позволяет получить список публикаций любого автора, зарегистрированного в РИНЦ. При этом в качестве формата выходных данных был выбран широко известный JavaScript Object Notation (JSON), удобный для чтения человеком и компьютером [6]. JSON-файл, содержащий библиографическую информацию, легко импортируется в документо-ориентированную БД MongoDB. MongoDB обладает гибкостью, масштабируемостью и очень быстро работает даже с большими объёмами данных. Работа с MongoDB осуществлялась посредством модуля pymongo языка программирования Python, который позволяет, как загружать информацию в БД, так и писать к ней запросы.

NoSQL – это не одна конкретная технология. NoSQL – это множество подходов к хранению и обработке данных, которые объединены общей идеей. Самые известные подходы реализации NoSQL представлены в табл. 1, взятой из [7].

Таблица 1

Подходы к реализации NoSQL

Тип	Суть подхода	Область использования	Примеры СУБД
Документо-ориентированные	Отсутствует какая-либо схема данных. Сущности – это документы в форматах XML, JSON и т.п.	Хранение независимых документов без поддержания ссылочной целостности между ними (данные форумов, каталогов товаров, систем логирования и сбора статистики)	MongoDB, CouchDB
Хранилища типа «ключ-значение»	Данные хранятся в виде пар ключ-значение(хэш-таблица). Значениями могут быть как простые типы данных, так и массивы, списки, множества и т.п.	Промежуточное (или основное) звено для систем логирования и сбора статистики	DynamoDB, Redis, Riak

Продолжение таблицы 1

Колоночные	Данные хранятся в виде последовательности столбцов, а не строк. Таблица представляет собой совокупность колонок, каждая из которых представляет собой таблицу из одного поля. Такие системы гораздо медленнее работают на запись, чем на чтение.	Аналитические системы.	Hadoop, HBase
Граф-ориентированные	Данные хранятся в виде графа с вершинами (узлы) и ребрами (связи между ними)	Моделирование социальных графов (социальные сети)	Neo4j, Infinite Graph

Существует огромное множество NoSQL БД. Они отличаются по типу используемых моделей данных и языкам запросов. Также в них применены различные внутренние системы хранения данных.

NoSQL – это не замена традиционного реляционного подхода к хранению и обработке данных. Данный подход применяется тогда, когда решаемые задачи связаны или с большими и постоянно возрастающими объемами данных (которые требуют высокой масштабируемости), или связанных с хранением таких данных, которые сильно отличаются от реляционной формы представления (документно-ориентированные системы, объектно-ориентированные системы). Часто создают СУБД с поддержкой как традиционно-реляционного подхода, так и альтернативных NoSQL-решений. Реляционные модели лучше подходят для относительно небольших объемов данных высокой ценности (таких как данные о пользователях некоторой информационной системы, билинговая информация), а NoSQL решения – для больших объемов данных низкой ценности (ведение логов и сбор статистики, хранение документов).

Организация работы с библиографической информацией является неотъемлемой частью научно-исследовательской работы преподавателей и студентов. Традиционные способы, такие как создание картотек, уходят в прошлое. Развитие информационных технологий привело к созданию специализированного программного обеспечения – систем управления библиографической информацией (СУБИ). В своей работе [8] Караваев Н.Л. указывает, что архитектура современных СУБИ включает следующие компоненты: БД для хранения библиографической информации (автор, название, издательство, журнал, год и т. д.); интерфейс с библиотечными каталогами, онлайн-журналами и другими базами данных; интерфейс с текстовыми процессорами, который позволяет автоматически вставлять библиографические ссылки и списки; систему генерации ссылок и списков, которая позволяет сразу оформлять их в требуемом стандарте. Автор

сравнивает между собой наиболее популярные СУБИ: EndNote, Zotero, Mendeley и Citavi.

Анализу существующих программных средств обработки библиографической информации посвящены также работы [9, 10]. Здесь указаны следующие проблемы, имеющиеся в настоящее время в области управления библиографической информацией: малое количество СУБИ с интерфейсом на русском языке; неудобство интерфейса СУБИ для пользователя; трудно или невозможно создать/импортировать много новых записей в БД (и/или новые стандарты библиографического описания); нет автоматической нумерации списка процитированных источников, полученного в результате работы СУБИ; нет импорта библиографической информации из документов MS Word и других текстовых редакторов для ее повторного использования; в большинстве случаев бесплатной является лишь пробная версия СУБИ, имеющая ограниченный период действия и/или функционал.

Для устранения описанных выше проблем авторами была разработана информационная система для документо-ориентированного хранения данных о научных публикациях. Исходная информация о публикациях была импортирована с сайта крупнейшей в России электронной библиотеки научных публикаций eLIBRARY.RU, обладающей богатыми возможностями поиска и анализа научной информации. Национальная библиографическая БД научного цитирования аккумулирует более 26 миллионов публикаций российских ученых, а также информацию о цитировании этих публикаций из более 5000 российских журналов.

Для извлечения библиографической информации был написан «паук» (web-crawler) с использованием фреймворка Scrapy [11]. Модель данных, собираемых «пауком», представляет собой отдельный класс, содержащий перечень полей искомых данных. Объектом парсинга в нашем случае будут являться публикации, а набором атрибутов – их характеристики (если у каждого объекта перечень доступных атрибутов различный, то итоговым набором будет являться объединение множеств атрибутов всех объектов). Необходимо хранить такие атрибуты, как идентификатор, гиперссылка, название публикации, авторы, номер журнала, аннотация, ключевые слова, издательство или название журнала, язык публикации, год издания, номера страниц, индекс УДК, том журнала, в какие БД входит публикация: РИНЦ, Scopus, WoS; DOI – Digital Object Identificator, тип публикации (статья, книга и т.д.).

«Паук» начинает работу с персональной страницы автора. Из неё извлекаются гиперссылки на публикации. При этом необходимо учесть, что для неавторизованного пользователя отображается по двадцать публикаций на странице, авторизованному пользователю доступны уже сто публикаций на каждой странице. Также неавторизованному пользователю недоступна некоторая информация о публикации, например список использованных

источников, ссылка на полный текст публикации и т.д. Авторизация пользователя осуществляется с помощью связки Selenium + PhantomJS [12, 13].

Обработка публикаций конкретного автора происходит следующим образом. Из html-кода веб-страницы автора извлекается таблица с id = restab. Строки этой таблицы содержат информацию об опубликованных автором работах. Из каждой строки указанной таблицы «паук» получает название и список авторов очередной публикации. Если в строке есть гиперссылка (@href) на публикацию, то собранная информация передаётся функции parse_paper, которая возвращает дополнительные сведения о публикации (соавторы, название, том, номер, номера страниц, аннотация, ключевые слова и т.д.). Если публикация не входит в РИНЦ (например, она извлечена из списка цитируемой литературы некоторой статьи), то «паук» переходит к следующей строке таблицы, а собранная на этот момент краткая информация о публикации записывается в JSON-файл.

Собранная информация помещается в запись словаря с ключом 'ru' или 'en' в зависимости от языка публикации. Ниже приведён пример словаря, который возвращает функция parse_paper:

{
"ru": {"url": "http://elibrary.ru/item.asp?id=32434529",
"pages": "57-62", "id": "arw32434529", "idb": ["РИНЦ"],
"title": "ИССЛЕДОВАНИЕ ДИНАМИКИ ГИДРОУПРУГИХ КОЛЕБАНИЙ КОЛЬЦЕВОГО КАНАЛА С ГЕОМЕТРИЧЕСКИ НЕРЕГУЛЯРНОЙ ВНЕШНЕЙ СТЕНКОЙ ПРИ НАЛИЧИИ ВИБРАЦИИ",
"entry": "journal", "number": "5 (25)",
"author": ["Калинина А.В.", "Кондратов Д.В.", "Могилевич Л.И."],
"abstract": "Построена математическая модель механической системы, представляющая собой трубу кольцевого профиля, образованную двумя поверхностями соосных цилиндрических оболочек, взаимодействующих с вязкой несжимаемой жидкостью, внешняя из которых является геометрически нерегулярной, а внутренняя - абсолютно жесткий цилиндр, при воздействии вибрации. Представленная математическая модель состоит из уравнений динамики вязкой несжимаемой жидкости, уравнений динамики геометрически нерегулярной и геометрически регулярной оболочек и соответствующих граничных условий. Приведено решение задачи гидродинамики и найдена амплитудная частотная характеристика внешней геометрически нерегулярной оболочки.",
"keywords": ["СООСНЫЕ ОБОЛОЧКИ", "ГИДРОДИНАМИКА", "ВИБРАЦИЯ", "ГЕОМЕТРИЧЕСКИ НЕРЕГУЛЯРНАЯ ОБОЛОЧКА", "СВОБОДНОЕ ОПИРАНИЕ", "COAXIALLY SHELLS", "HYDRODYNAMICS", "VIBRATION", "GEOMETRICALLY IRREGULAR SHELL", "FREE ATTACHED"],
"publisher": "ТЕХНИЧЕСКОЕ РЕГУЛИРОВАНИЕ В ТРАНСПОРТНОМ СТРОИТЕЛЬСТВЕ", "udc": "51-74", "year": "2017"}}

После обработки всех публикаций автора, расположенных на текущей странице, в функции parse_author из html-кода извлекается гиперссылка на следующую страницу с публикациями автора. Поиск этой гиперссылки происходит по тексту «Следующая страница».

Графический интерфейс пользователя описанной информационной системы был создан на PySide [14]. Вид главного окна разработанного приложения показан на рис. 1 (БД заполнена информацией о публикациях сотрудников кафедры математического и компьютерного моделирования ФГБОУ ВО «Саратовский национальный исследовательский государственный университет имени Н.Г. Чернышевского»).

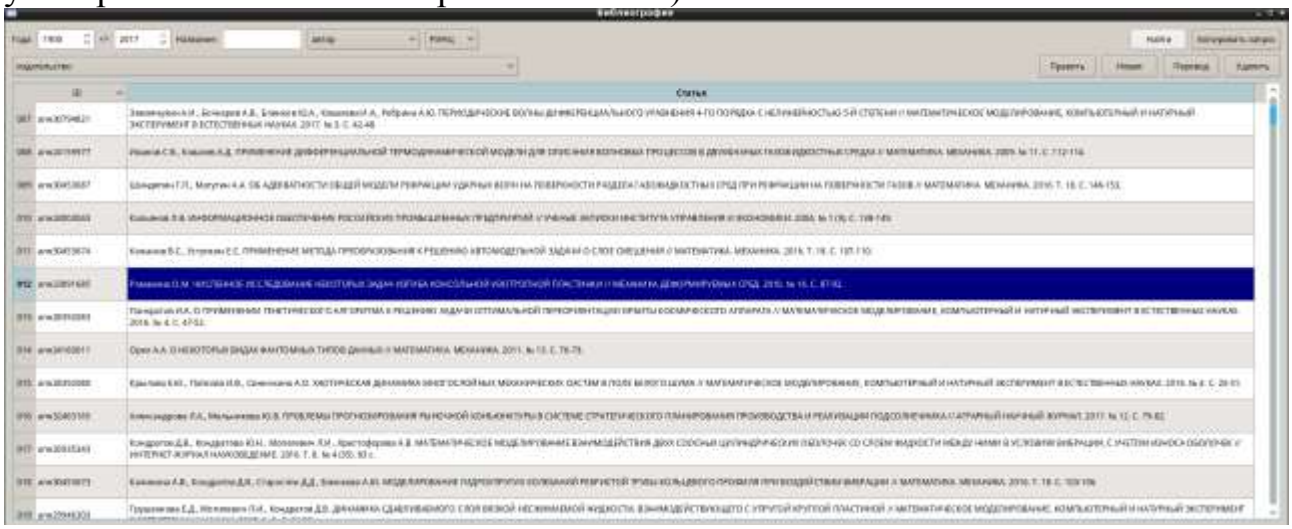

Рис.1. Главное окно приложения

Заметим, что описание публикаций, полученных из БД MongoDB, автоматически формируется по ГОСТ Р 7.0.5-2008.

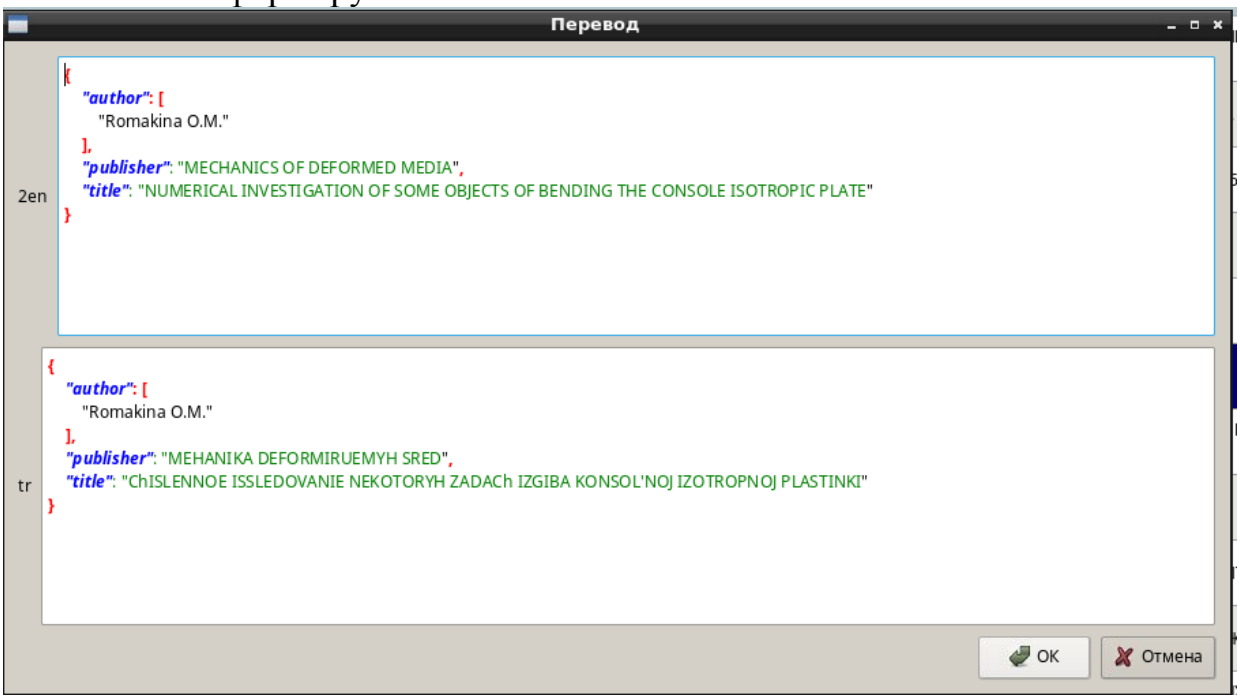

Рис.2. Пример перевода и транслитерации

Пример автоматического перевода на английский язык и транслитерации данных о публикации, выделенной на рис. 1, показан на рис. 2. Перевод был осуществлён с помощью Google Translate API [15, 16].

Заметим, что в приложении предусмотрена возможность фильтрации статей по году издания, издательству, автору, тексту в названии работы, вхождению в базы данных РИНЦ, Web of Science, Scopus. Также можно вручную добавить новую публикацию или поправить ранее внесенные данные. Пример правки внесённой библиографической информации показан на рис. 3.

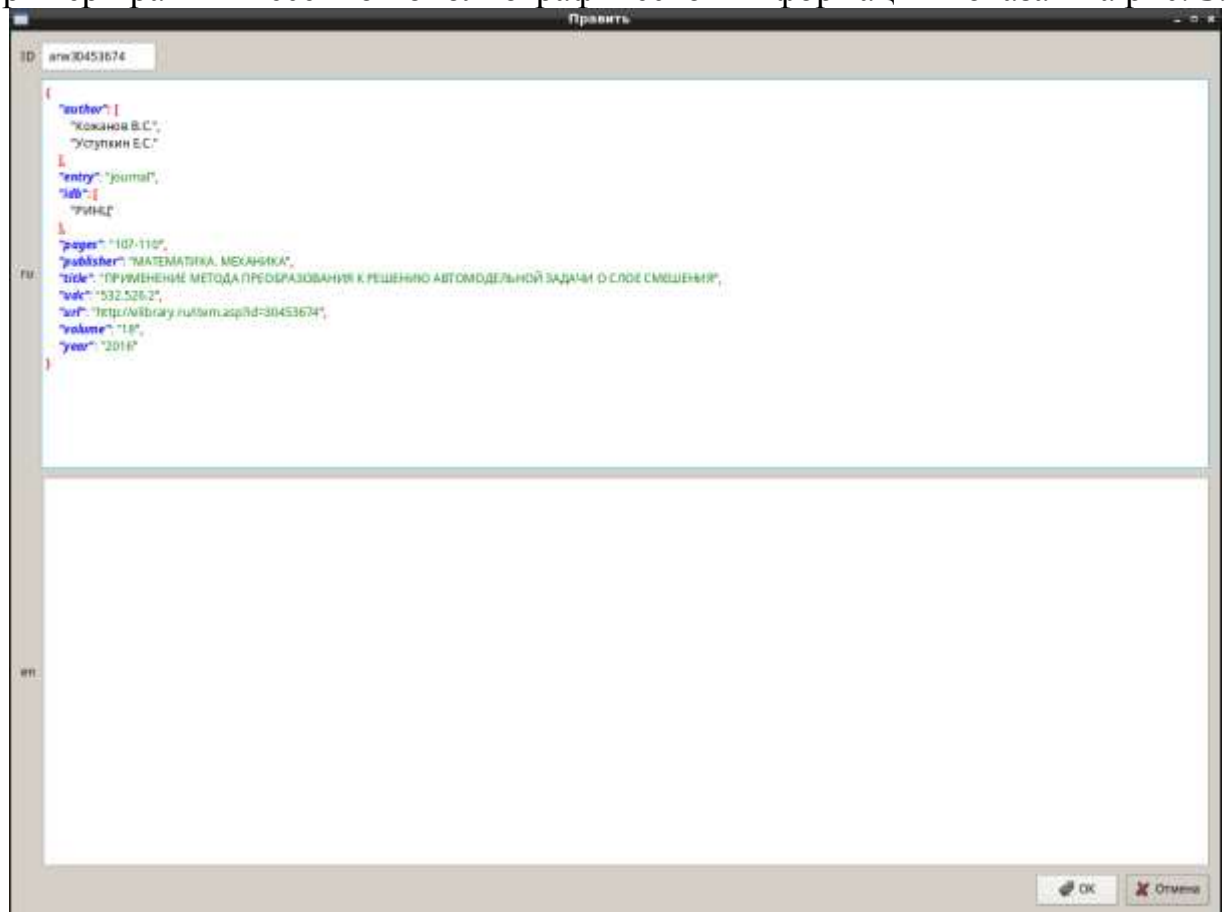

Рис.3. Правка библиографической информации

В работе построена информационная система, позволяющая автоматизировать обработку научных публикаций различного рода. Для наполнения документо-ориентированной БД MongoDB с помощью фреймворка Scrapy был написан «паук», извлекающий с сайта РИНЦ публикации того или иного автора. БД была заполнена публикациями сотрудников кафедры математического и компьютерного моделирования. Разработано графическое приложение, позволяющее структурировать и анализировать накопленную информацию.

Литература

1. Блинков Ю.А., Панкратов И.А. Обработка библиографической информации с помощью нереляционных баз данных // XIX Всероссийская студенческая научно-практическая конференция Нижневартовского государственного

университета: сборник статей (г. Нижневартовск, 4–5 апреля 2017 года) / отв. ред. А.В. Коричко. Ч. 2. Информационные технологии. Математика. Физика. 2017. С. 425-426.
2. Блинков Ю.А., Панкратов И.А. Автоматизация обработки библиографической информации // Молодежная наука в развитии регионов: материалы Всерос. (с междунар. участием) науч.-практ. конф. студентов и молодых ученых (Березники, 26 апреля 2017). Пермь: Изд-во Перм. нац. исслед. политех. ун-та. 2017. С. 39-40.
3. Блинков Ю.А., Панкратов И.А. Построение информационной системы для обработки библиографической информации // Решение: материалы Шестой всерос. науч.-практ. конф., г. Березники, 20 октября 2017. Пермь: Изд-во Перм. нац. исслед. политех. ун-та. 2017. С. 77-78.
4. Блинков Ю.А., Панкратов И.А. Хранение и обработка библиографической информации с помощью NoSQL // Актуальные направления научных исследований XXI века: теория и практика. 2017. № 7, ч. 1 (33-1). С. 247-249.
5. Редмонд, Э., Уилсон, Д.Р. Семь баз данных за семь недель. Введение в современные базы данных и идеологию NoSQL. М.: ДМК Пресс, 2013. 384 с.
6. Официальный сайт JSON. URL: http://json.org/json-ru.html (дата обращения: 15.03.2018).
7. Мухина Ю.Р. Обзор NoSQL решений управления данными // Управление в современных системах. 2013. № 1. С. 68-73.
8. Караваев Н.Л. Автоматизация работы с библиографической информацией как часть научно-исследовательской деятельности студентов // Вестник гуманитарного образования. 2017. № 2. С. 17-20.
9. Садковская С.Ю., Тараник М.А. Разработка системы управления библиографической информацией // Молодёжь и современные информационные технологии. Сборник трудов XII Международной научно-практической конференции студентов, аспирантов и молодых ученых. Томск: Национальный исследовательский Томский политехнический университет. 2014. Т. 2. С. 26-27.
10. Логунова О.С., Ильина Е.А., Попов, С. Н., Кочержинская Ю.В., Сибилева Н.С. Структура программного модуля для обработки библиографической информации // Омский научный вестник. 2016. № 6 (150). С. 158-164.
11. Scrapy documentation. URL: https://doc.scrapy.org/en/latest/ (дата обращения: 10.03.2018).
12. Selenium documentation. URL: https://seleniumhq.github.io/selenium/docs/api/py/api.html (дата обращения: 20.03.2018).
13. Официальный сайт PhantomJS URL: http://phantomjs.org (дата обращения: 15.03.2018).
14. PySide Documentation URL: https://pyside.github.io/docs/pyside/index.html (дата обращения: 25.01.2018).

15. Free Google Translate API for Python URL: https://pypi.python.org/pypi/googletrans (дата обращения: 25.01.2018).
16. Bi-directional transliterator for Python URL: https://pypi.python.org/pypi/transliterate (дата обращения: 25.01.2018).