

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
(наименование института полностью)

Кафедра _____ Прикладная математика и информатика
(наименование)

09.03.03 Прикладная информатика
(код и наименование направления подготовки, специальности)

Корпоративные информационные системы
(направленность (профиль) / специализация)

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(БАКАЛАВРСКАЯ РАБОТА)**

на тему _____ Разработка приложения по сбору данных для ООО «Статус»

Студент

П. Ю. Лукичев

(И.О. Фамилия)

(личная подпись)

Руководитель

к.т.н., доцент, О. В. Аникина

(ученая степень, звание, И.О. Фамилия)

Консультант

А. В. Москалюк

(ученая степень, звание, И.О. Фамилия)

Тольятти 2021

Аннотация

С. 48, рис. 34, табл. 1, лит. 20 источников

ПРИЛОЖЕНИЕ, СКРИПТ, СБОР ДАННЫХ, UML, BPMN, API, JSON, PYTHON.

Тема: «Разработка приложения по сбору данных для ООО «Статус»».

Выпускную квалификационную работу выполнил студент Тольяттинского государственного университета, института математики, физики и информационных технологий, Лукичев Павел Юрьевич.

Целью данной работы является разработка приложения для автоматизации процесса сбора данных.

Объектом исследования является процесс сбора данных о запчастях из разных каталогов автомобильных запчастей.

Предметом исследования – автоматизация процесса сбора требуемых данных о запчастях из каталогов поставщиков магазина автомобильных запчастей.

Дано описание деятельности организации ООО «Статус», рассмотрен процесс подбора автомобильных запчастей. Построены модели «КАК ЕСТЬ» и «КАК ДОЛЖНО БЫТЬ» с использованием методологии BPMN. Сформулированы основные требования к разрабатываемому приложению при помощи технологии FURPS+.

Дано описание принципа работы с программными средствами по сбору данных, выбрано средство реализации приложения, описан алгоритм работы приложения. В конечном итоге приложение по сбору данных было разработано и протестировано в реальных условиях.

ABSTRACT

The graduation work is devoted to development of an application for data collection for OOO Status.

The graduation work consists of an explanatory note on 48 pages, introduction, including 34 figures, 1 table, the list of 20 references including 6 foreign sources and two appendixes.

The aim of the work is to develop an application for automating the data collection process.

The object of the graduation work is the process of collecting data of spare parts from different catalogs of automobile spare parts.

The subject of the graduation work is an application for automating the process of collecting data.

The graduation work may be divided into three logically connected parts: analyzing the process of collecting data, developing an application for collection data and testing this application.

In the first part we study the activity of organization OOO Status. Then we analyze the process of selecting automobile spare parts by using BPMN methodology with “AS-IS” and “TO-BE” models. Finally we describe the basic requirements for the application we will develop.

The second part of graduation work includes developing an application. We outline the method of working with data collection software. Next we choose the tool for developing an application – Python programming language. Finally we developed an application for collecting data.

Third part of graduation work includes testing the developed application. As a result of testing we came to the conclusion that the developed application works correctly and works without errors.

As a result we have developed an application to automate the data collection process. The developed application can be used by other organizations specialized on automobile spare parts sale.

Оглавление

Введение.....	5
Глава 1 Анализ процесса сбора данных в организации ООО «Статус».....	7
1.1 Описание деятельности организации ООО «Статус»	7
1.2 Описание процесса подбора автомобильных запчастей.....	9
1.3 Разработка и анализ модели процесса подбора автомобильных запчастей «КАК ЕСТЬ».....	13
1.4 Разработка и анализ модели процесса подбора автомобильных запчастей «КАК ДОЛЖНО БЫТЬ»	15
1.5 Требования к разрабатываемому приложению.....	17
Глава 2 Реализация приложения по сбору данных.....	20
2.1 Описание принципа работы со средствами для сбора данных	20
2.2 Выбор технологии для реализации приложения	22
2.3 Описание алгоритма работы приложения по сбору данных	25
2.4 Практическая реализации приложения по сбору данных.....	29
Глава 3 Тестирование разработанного приложения по сбору данных.....	41
Заключение	45
Список используемой литературы и используемых источников.....	46
Приложение А Фрагмент файла «Main.py».....	49
Приложение Б Фрагмент файла «Logic.py»	50

Введение

Активное развитие высоких и информационных технологий влечет за собой необходимость автоматизации бизнес-процессов в различных сферах деятельности. В настоящий момент существует немало подобных процессов, требующих автоматизации. Данная выпускная квалификационная работа (ВКР) посвящена теме автоматизации одного из процессов путем разработки приложения, способного упростить сбор определенных данных, необходимых для принятия управленческих решений.

Актуальность данной темы обусловлена необходимостью обеспечить лицо, принимающее решения в рамках деятельности магазина автомобильных запчастей, приложением, которое позволит в значительной степени упростить и ускорить процесс сбора данных из нескольких каталогов автомобильных запчастей различных поставщиков.

Целью данной ВКР является разработка приложения для автоматизации процесса сбора данных.

Для решения данного вопроса необходимо выполнить следующие поставленные **задачи**:

- изучить и проанализировать процесс сбора данных;
- выполнить проектирование разрабатываемого приложения по сбору данных;
- выбрать методы реализации разрабатываемого приложения по сбору данных;
- разработать приложение по сбору данных;
- провести тестирование разработанного приложения по сбору данных.

Объектом исследования данной ВКР является процесс сбора данных о запчастях из разных каталогов автомобильных запчастей.

В качестве **предмета исследования** выступает автоматизация процесса сбора требуемых данных о запчастях из каталогов поставщиков.

В рамках текущей бакалаврской работы будет рассматриваться автоматизация процесса сбора данных на примере организации, именуемой ООО «Статус». Тема данной бакалаврской работы была выбрана по инициативе автора и предложена для реализации в ООО «Статус», поскольку предложенное автоматизированное решение непосредственным образом повлияет на развитие указанной организации.

В процессе выполнения бакалаврской работы были использованы теоретический (анализ) и эмпирический (моделирование) методы исследования. Анализ объекта исследования позволит изучить процесс сбора данных путём его исследования, а моделирование позволяет изменить процесс сбора данных средствами разрабатываемого приложения.

Таким образом, в результате разработки приложения по сбору данных будет автоматизирован процесс сбора информации о запчастях из каталогов поставщиков организации ООО «Статус». Подобная автоматизация положительным образом скажется на выполнении процесса сбора данных в рамках рассматриваемой организации и поможет лицу, принимающему решения, избавиться от избыточных действий, что скажется на точности принимаемых решений и скорости их принятия.

В первой главе текущей работы будет описана структура организации ООО «Статус» и подробно рассмотрен процесс сбора данных из каталогов поставщиков. В частности, будет дано описание структуры организации, разобран процесс сбора данных, составлены соответствующие диаграммы и разработан список основных требований к разрабатываемому приложению.

Во второй главе данной ВКР будет описан принцип работы с программными средствами для сбора данных, выбрана технология для реализации приложения, описан алгоритм работы разрабатываемого приложения и реализовано приложение по сбору данных.

Третья глава затронет процесс тестирования разработанного приложения на примере выполнения реальной задачи в рамках организации ООО «Статус».

Глава 1 Анализ процесса сбора данных в организации ООО «Статус»

Данная глава посвящена анализу объекта исследования – процесса сбора данных. В частности, необходимо описать процесс сбора данных на примере существующего магазина автомобильных запчастей – основного вида деятельности организации ООО «Статус», и, на основе проведенного анализа, составить список требований к разрабатываемому приложению.

1.1 Описание деятельности организации ООО «Статус»

Основным видом деятельности организации ООО «Статус» является розничная торговля автомобильными запчастями. Торговля ведется в рамках магазина автомобильных запчастей. Помимо торговли запчастями магазин предоставляет услуги по подбору автомобильных запчастей, а также продает различные расходные материалы (ГСМ – масло, смазки, а также воздушные фильтры, уплотнители, присадки и т.п.).

Магазин автомобильных запчастей ведет свою деятельность начиная с 2011 года. На момент написания ВКР данный магазин автомобильных запчастей является действующим. Рассматриваемый магазин автомобильных запчастей не имеет филиалов и территориально располагается в Автозаводском районе города Тольятти.

В рассматриваемом магазине автомобильных запчастей имеется штат сотрудников, каждый из которых выполняет важную и определенную ему роль. Организационная структура данного магазина автомобильных запчастей представлена на рисунке 1.



Рисунок 1 – Организационная структура магазина автомобильных запчастей

Во главе организации стоит директор. Его рабочее место находится непосредственно в магазине, в отдельном помещении. Директор занимается управленческой деятельностью.

В подчинении директора находятся лица следующих должностей:

- заместитель директора,
- бухгалтер,
- системный администратор,
- продавец.

В обязанности заместителя директора входит управление персоналом, а также решение различных вопросов, связанных с деятельностью магазина, в период отсутствия на рабочем месте директора.

Бухгалтер занимается бухгалтерским учетом. Через него проходят документы, оплачиваются счета поставщиков, производятся налоговые отчисления. Работает удаленно.

Системный администратор в данной конкретной организации занимается решением возникающих вопросов, связанных с технической стороной фирмы. К данным вопросам относятся: настройка новых устройств, чистка и поддержание работоспособности оборудования (компьютеров и периферии), замена неисправного оборудования.

Продавец в рассматриваемой организации занимается обслуживанием клиентов. Он отвечает за консультацию клиента, приём оплаты клиентом заказа, выдачу кассового чека. Также в его обязанности входит помощь клиенту с подбором запчастей.

Подбор запчастей – процесс поиска необходимых клиенту запчастей в специальных каталогах поставщиков. Для каждого автомобиля существует большое количество различных деталей. Суть подбора заключается в том, чтобы в специальном справочнике автомобильных деталей найти требуемую клиенту запчасть, скопировать её уникальный номер – артикул, и, имея данный артикул, осуществить с его помощью поиск детали в каталогах имеющихся поставщиков. На данный момент у магазина имеется 6 активно используемых каталогов поставщиков. Продавец в процессе подбора запчастей должен осуществить поиск по каждому из каталогов, собрать необходимую информацию и, на основе полученной информации, обсудить с клиентом возможные варианты заказа.

Исходя из вышеописанного, можно заметить, что в процессе подбора автомобильных запчастей имеется ряд повторяющихся действий, совершаемых лицом, исполняющим данный процесс – поиск запчасти в каждом каталоге имеющихся поставщиков.

1.2 Описание процесса подбора автомобильных запчастей

Рассмотрим процесс подбора автомобильных запчастей более детально.

Процесс начинается с клиента, изъявившего желание сделать заказ. Продавец уточняет у клиента необходимую для подбора информацию: марка

автомобиля, год выпуска, VIN-номер. Полученная информация заносится в специальную форму сайта-справочника. На рисунке 2 изображен фрагмент сайта, используемого магазином автомобильных запчастей в качестве справочника деталей – «карточка» автомобиля одного из клиентов.

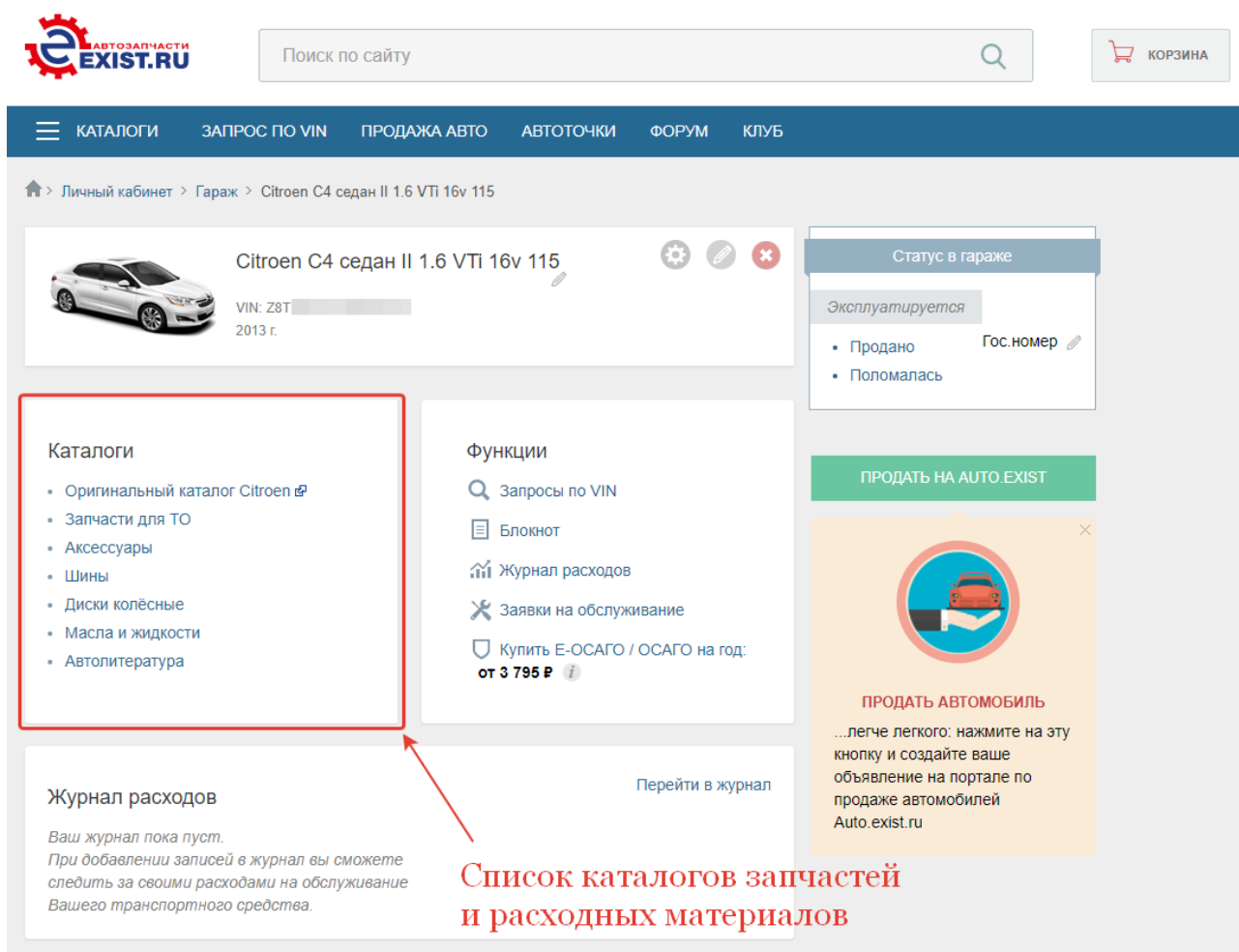


Рисунок 2 – Фрагмент сайта-справочника с «карточкой» автомобиля клиента

На следующем шаге продавец выясняет, какая деталь необходима клиенту. Получив ответ клиента, продавец ищет данную деталь в одном из каталогов, который соответствует типу искомой запчасти. На рисунке 3 изображен пример поиска запчасти, находящейся в каталоге «Запчасти для ТО» - «передний тормозной диск».

Предложения для Citroen/Peugeot 4249.J6

НАИМЕНОВАНИЕ	
ЗАПРОШЕННЫЙ АРТИКУЛ	
★ Citroen/Peugeot 4249.J6 2.5 Диск тормозной передний вентилируемый	
Артикул оригинальной запчасти	
ПРЕДЛОЖЕНИЯ ПО ЗАМЕНИТЕЛЯМ	
★ NK 201927 5.0 Диск тормозной вентилируемый	
Артикул не оригинальной запчасти (аналог)	
★ Iberis IB151007 Диск тормозной передний вентилируемый	
★ Zekkert BS-5024 0.7 Диск тормозной передний вентилируемый	
★ Patron PBD4183 0.6 Диск тормозной вентилируемый	

Рисунок 3 – Пример поиска запчасти в каталоге сайта-справочника

Результат поиска, изображенный на рисунке 3, содержит в себе данные об оригинальных запчастях, а также данные об аналогах – не оригинальных запчастях. Разница между ними, как правило, заключается в цене детали и в качестве производимой брендом продукции. Несмотря на это, поставщики, с которыми работает магазин, являются проверенными и не сотрудничают с некачественными брендами.

На следующем шаге продавец обязан уточнить у клиента возможность заказа запчасти - аналога. Если клиент согласен на аналог, то продавец, исходя из своего личного опыта, подбирает аналог конкретного бренда. Найдя деталь нужного бренда, продавец копирует её артикул – код детали. Следующим шагом является поиск самого выгодного предложения детали данного артикула в каталогах поставщиков магазина.

На текущий момент у магазина имеется 6 поставщиков запчастей. Задача продавца заключается в поочередном просмотре каталогов каждого из поставщиков на предмет нахождения товара с совпадающим артикулом.

Просмотр осуществляется в виде поиска по артикулу детали. Пример поиска в одном из каталогов изображен на рисунке 4.

The screenshot shows the website interface for 'Москворечье' (Moskvorochye). At the top, there is a navigation bar with links for 'Акции', 'Каталог призов', 'Обучение', 'Каталоги производителей', 'Реестр документов', 'Инструкции', and 'FAQ'. Below this, a search bar prompts the user to 'Введите номер детали для поиска или VIN' (Enter the part number for search or VIN). The search results are displayed in a table with columns for 'Фирма' (Company), 'Номер' (Number), 'Наименование' (Name), 'Кол-во' (Quantity), 'Склад' (Warehouse), 'Сред. срок' (Avg. lead time), 'Мин. упак.' (Min. pack), and 'Цена' (Price). The results show several offers for brake discs from NK and MILES. Below the main results, there is a section for 'Проверенные аналоги (заменители) для запрошенного артикула' (Verified analogs (replacements) for the requested part number), listing various brands like Bosch and Brembo.

Фирма	Номер	Наименование	Кол-во	Склад	Сред. срок	Мин. упак.	Цена
NK	201927	Диск тормозной CITROEN BERLINGO +ESP/C5/C4/PEUGEOT 207 05>/307 05> передний	8	Самара на складе	2	2	1 618.61 руб.
NK	201927	Диск тормозной CITROEN BERLINGO +ESP/C5/C4/PEUGEOT 207 05>/307 05> передний	8	Казань 1 день	2	2	1 618.61 руб.
NK	201927	Диск тормозной CITROEN BERLINGO +ESP/C5/C4/PEUGEOT 207 05>/307 05> передний	>100	Центр 2 дня	2	2	1 618.61 руб.
Проверенные аналоги (заменители) для запрошенного артикула							
Miles	K000042	Диск тормозной CITROEN BERLINGO +ESP/C5/C4/PEUGEOT 207 05>/307 05> передний	12	Казань 1 день	2	2	1 515.00 руб.
Miles	K000042	Диск тормозной CITROEN BERLINGO +ESP/C5/C4/PEUGEOT 207 05>/307 05> передний	>100	Центр 2 дня	2	2	1 515.00 руб.
BLUE PRINT	ADP154301	Диск тормозной CITROEN BERLINGO +ESP/C5/C4/PEUGEOT 207 05>/307 05> передний	4	Самара на складе	2	2	1 771.00 руб.
Bosch	0 986 478 979	Диск тормозной CITROEN BERLINGO +ESP/C5/C4/PEUGEOT 207 05>/307 05> передний	4	Самара на складе	2	2	2 074.00 руб.
Bosch	0 986 478 979	Диск тормозной CITROEN BERLINGO +ESP/C5/C4/PEUGEOT 207 05>/307 05> передний	2	Казань 1 день	2	2	2 074.00 руб.
Bosch	0 986 478 979	Диск тормозной CITROEN BERLINGO +ESP/C5/C4/PEUGEOT 207 05>/307 05> передний	18	Центр 2 дня	2	2	2 074.00 руб.
Brembo	09.8760.11	Диск тормозной CITROEN BERLINGO +ESP/C5/C4/PEUGEOT 207 05>/307 05> передний	2	Самара на складе	2	2	2 431.00 руб.
Brembo	09.8760.11	Диск тормозной CITROEN BERLINGO +ESP/C5/C4/PEUGEOT 207 05>/307 05> передний	2	Казань 1 день	2	2	2 431.00 руб.
Brembo	09.8760.75	Диск тормозной CITROEN BERLINGO +ESP/C5/C4/PEUGEOT 207 05>/307 05> передний	2	Центр 2 дня	2	2	3 568.00 руб.
Brembo	09.9619.10	Диск тормозной CITROEN BERLINGO +ESP/C5/C4/PEUGEOT 207 05>/307 05> передний	2	Самара на складе	2	2	2 270.00 руб.

Рисунок 4 – Пример поискового запроса по артикулу в каталоге поставщика «Москворечье»

В примере, изображенном на рисунке 4, показан результат запроса по артикулу, взятому из сайта-справочника. В первых строках находится искомая запчасть требуемого бренда. Ниже располагаются аналоги искомой детали других брендов. Разница между тремя верхними строками заключается в складе, на котором находится данная деталь. От склада нахождения запчасти зависит срок поставки.

Детали, найденные в результате процесса просмотра каталогов поставщиков, записываются, как правило, на лист бумаги – списком.

Таким образом, в процессе подбора продавец проходит по каждому каталогу поставщиков магазина. Его задача заключается в том, чтобы найти

автомобильную деталь с оптимальным сроком поставки и ценой. Приоритетным является именно срок поставки, поскольку зачастую клиенту необходимо в кратчайшие сроки устранить поломку своего автомобиля; вторым по приоритету является такой параметр, как стоимость товара.

По факту просмотра всех каталогов продавец согласует заказ с клиентом, озвучивая сроки поставки и итоговую стоимость, включающую наценку магазина. Если клиент соглашается, продавец оформляет заказ у поставщика, чье предложение является более выгодным для клиента.

Как уже было сказано ранее, основной проблемой является выполнение лицом, исполняющим процесс подбора автомобильных запчастей, череды одинаковых действий – просмотр очередного каталога, выписывание релевантных результатов на лист бумаги. Для лучшего понимания объекта автоматизации необходимо смоделировать данный бизнес-процесс при помощи специальных средств моделирования.

1.3 Разработка и анализ модели процесса подбора автомобильных запчастей «КАК ЕСТЬ»

Для моделирования бизнес-процесса «Подбор автомобильных запчастей» будет использована методология BPMN.

BPMN (от англ. Business Process Management Notation – Нотация управления бизнес-процессами) – это методика создания блок-схем, моделирующая этапы рассматриваемого бизнес-процесса от начала до конца. Создаваемая BPMN-модель позволяет визуально отобразить последовательность операций, выполняемых в бизнес-процессе [6].

Для разработки BPMN-модели будет использовано популярное средство для разработки визуальных моделей - Visio 2019. Данная программа входит в пакет лицензионного программного обеспечения «Microsoft Office профессиональный плюс».

На рисунке 5 изображена диаграмма процесса «Подбор автомобильных запчастей».

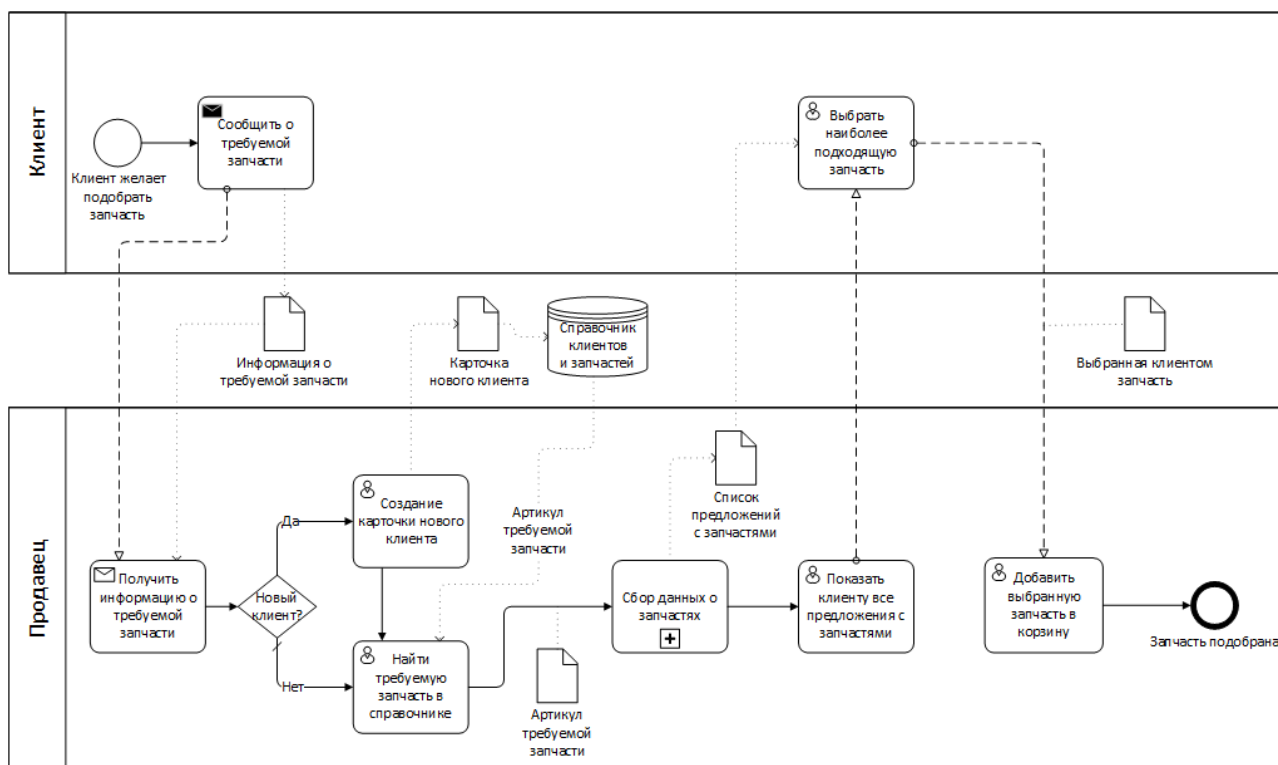


Рисунок 5 – Диаграмма процесса «Подбор автомобильных запчастей»

На диаграмме, представленной на рисунке 5, показан процесс подбора автомобильных запчастей. В нем участвует клиент и лицо, занимающееся подбором – в рамках магазина автомобильных запчастей под руководством ООО «Статус», данным лицом выступает продавец. Данный процесс начинается с обращения клиента и заканчивается оформленным на стороне продавца заказом. На диаграмме видно, что данный процесс содержит в себе процесс «Сбор данных о запчастях».

Процесс «Сбор данных о запчастях» олицетворяет работу продавца, связанную с просмотром каталогов каждого поставщика на предмет нахождения требуемых запчастей. Диаграмма процесса «Сбор данных о запчастях» изображена на рисунке 6.

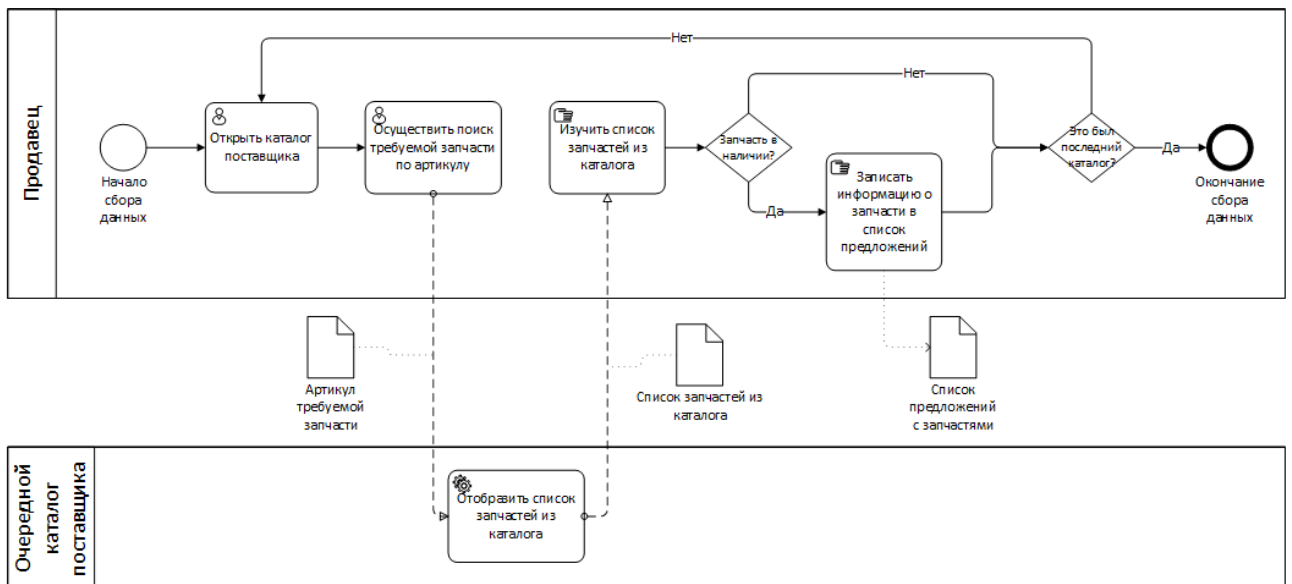


Рисунок 6 – Диаграмма процесса «Сбор данных о запчастях» «КАК ЕСТЬ»

На диаграмме, изображенной на рисунке 6, видно, что процесс «Сбор данных о запчастях» состоит из циклических действий лица, исполняющего процесс подбора запчастей - поиск по артикулу в каждом каталоге поставщиков [9]. Лицо, осуществляющее подбор автомобильных запчастей, вынуждено циклически выполнять одинаковые действия: открыть каталог очередного поставщика, осуществить поиск требуемой запчасти путем ввода в поисковую строку артикула, изучить полученный список запчастей каталога, найти в списке подходящую деталь, записать информацию о данной запчасти, - данный процесс продолжается до тех пор, пока не будут просмотрены все каталоги.

Исходя из всего вышеописанного, можно сделать вывод о том, что процесс «Сбор данных о запчастях» подлежит автоматизации. На основе полученной информации будет разработана модель данного процесса «КАК ДОЛЖНО БЫТЬ».

1.4 Разработка и анализ модели процесса подбора автомобильных запчастей «КАК ДОЛЖНО БЫТЬ»

Следующим шагом является разработка модели «КАК ДОЛЖНО БЫТЬ» [7]. Данная модель должна показать изменения, которые произойдут в процессе в результате автоматизации.

В рамках данной ВКР изменения затронут процесс «Сбор данных о запчастях», поскольку он содержит в себе ряд действий, которые необходимо упростить и автоматизировать. На рисунке 7 изображена диаграмма процесса «Сбор данных о запчастях» «КАК ДОЛЖНО БЫТЬ».

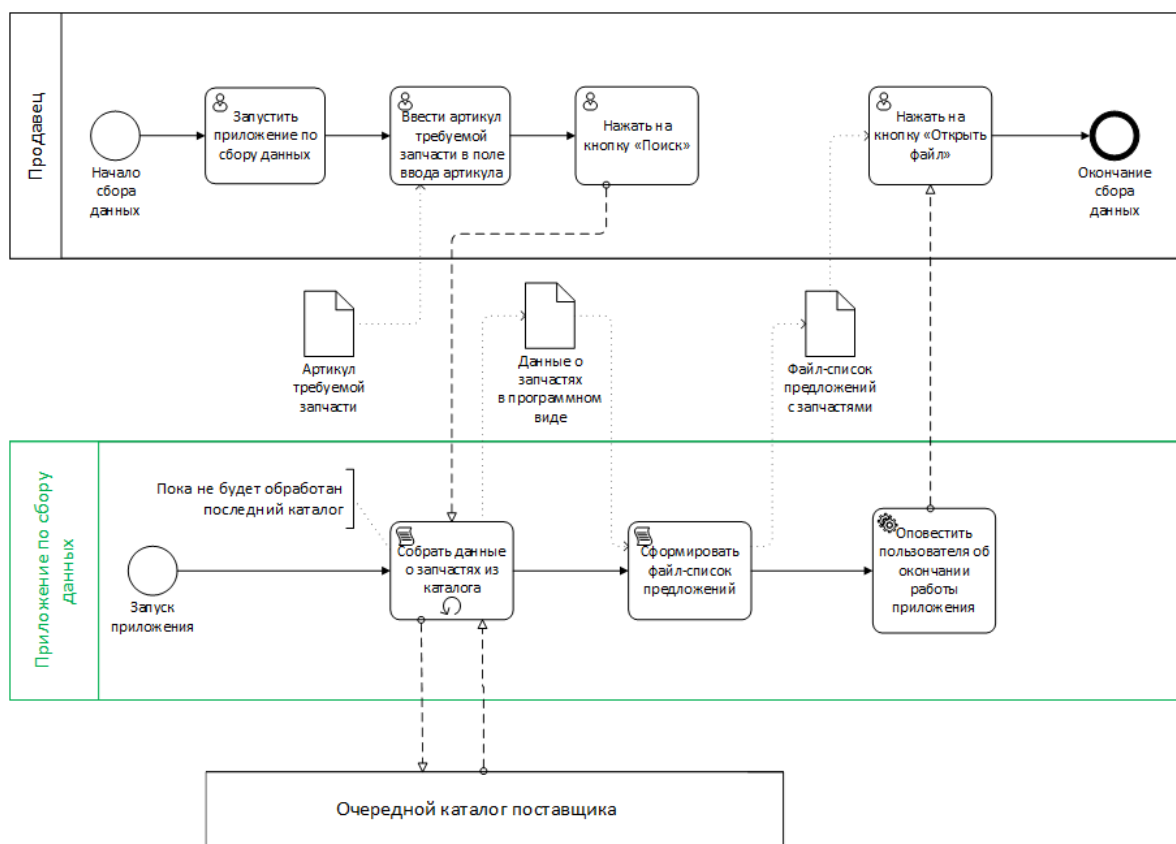


Рисунок 7 - Диаграмма процесса «Сбор данных о запчастях» «КАК ДОЛЖНО БЫТЬ»

Как следует из диаграммы, изображенной на рисунке 7, действия, выполняемые продавцом, изменились: их стало меньше, они упростились. Данная автоматизация предполагает разработку приложения, способного в полуавтоматическом режиме собирать необходимую пользователю информацию. От пользователя потребуется овладеть навыками работы с

приложением. Вся рутинная работа заменится простыми действиями: запустить приложение, вставить ранее скопированный артикул в специальное поле для ввода и нажать на кнопку, тем самым запустив процесс сбора данных. Приложение самостоятельно соберет необходимые пользователю данные и предоставит их в виде файла-таблицы.

Проанализировав текущее положение дел, а также то, в какую сторону необходимо двигаться для успешной автоматизации процесса подбора автомобильных запчастей, следующим шагом будет описание требований к разрабатываемому приложению.

1.5 Требования к разрабатываемому приложению

На основании проведенного анализа процесса подбора автомобильных запчастей, необходимо определить основные требования к разрабатываемому приложению.

Прежде чем приступить к описанию требований, необходимо выявить целевую аудиторию разрабатываемого приложения, его предполагаемые функциональные возможности и минимальные системные требования.

Целевой аудиторией для разрабатываемого приложения является уверенный пользователь персонального компьютера (ПК). Данный пользователь должен быть знаком с основными действиями, совершаемыми на ПК: пользование браузером и интернетом, запуск приложений, копирование и вставка выделенных объектов, манипуляции над файлами и папками.

В плане функциональных возможностей, приложение должно обладать необходимым для работы набором возможностей. В частности, результатом работы приложения по сбору данных должен быть файл с данными, полученными в ходе выполнения данным приложением своей задачи. Разрабатываемое приложение должно иметь простой и лаконичный пользовательский интерфейс, включающий в себя только необходимые для

работы с приложением элементы (кнопки, поля ввода, строку состояния приложения).

Минимальные системные требования для разрабатываемого приложения имеет смысл рассматривать с точки зрения системы, под которую разрабатывается данное приложение. Такой подход обусловлен тем, что разрабатываемое приложение не требует больших вычислительных мощностей и задействует самый минимальный объем ресурсов персонального компьютера. Единственным требованием является операционная система Windows 10 и подключение к сети Интернет.

Исходя из вышеперечисленного, можно сказать, что целевая аудитория разрабатываемого приложения - «уверенный пользователь ПК». Функционал разрабатываемого приложения должен удовлетворять потребности пользователя, работающего с данным приложением. Пользовательский интерфейс должен быть простым, лаконичным, но в то же время содержать в себе все самые необходимые элементы управления приложением. В результате можно сформулировать основные требования к разработке приложения.

Для разработки требований к приложению будет использована технология FURPS+.

FURPS+ - это классификация требований к программным системам. Она позволяет описать классификации атрибутов качества рассматриваемого программного обеспечения. Данная технология довольно часто используется для описания требований к разрабатываемому программному обеспечению.

Средствами технологии FURPS+ можно составить следующие требования к разрабатываемому приложению:

а) Functionality (функциональность):

- 1) сбор необходимой информации в полуавтоматическом режиме;
- 2) размещение собранной информации в файле;
- 3) обеспечение возможностью взаимодействовать с результирующим файлом непосредственно из приложения.

б) Usability (удобство использования):

- 1) простой и понятный пользовательский интерфейс;
 - 2) наличие возможности для получения справочной информации;
 - 3) наличие элементов интерфейса, указывающих на текущее состояние программы.
- в) Reliability (надежность):
- 1) допустимая периодичность сбоев – 1 раз в 200 часов.
- г) Performance (производительность):
- 1) допустимое использование приложением ресурсов персонального компьютера – до 100 Мбайт оперативной памяти.
- д) Supportability (уровень поддержки):
- 1) время для устранения ошибок – до 24 часов;
 - 2) расширяемость возможна по мере предоставления третьими лицами необходимых для расширения данных.
- е) Проектные ограничения:
- 1) операционная система – не ниже Windows 10;
 - 2) подключение к сети Интернет.

На основе сформулированных требований будет разработано приложение, автоматизирующее процесс сбора данных.

Выводы по главе 1

В текущей главе данной ВКР была описана деятельность организации ООО «Статус», выявлен объект автоматизации. Подробно рассмотрен процесс подбора автомобильных запчастей, разработаны модели «КАК ЕСТЬ» и «КАК ДОЛЖНО БЫТЬ». Заключительным шагом стала разработка требований к приложению средствами технологии FURPS+.

Глава 2 Реализация приложения по сбору данных

После описания основных требований для реализации приложения, можно приступать к разработке данного приложения. В данной главе текущей ВКР будет рассмотрен процесс проектирования и реализации приложения по сбору данных.

2.1 Описание принципа работы со средствами для сбора данных

Целью работы приложения является извлечение данных, находящихся в каталогах поставщиков. Каталоги поставщиков располагаются в интернете и доступны только коммерческим организациям, прошедшим специальную проверку. Для работы с web-ресурсами поставщиков, в частности – с каталогами запчастей, существует специальная технология доступа, называемая API.

API (от англ. Application Programming Interface – программный интерфейс приложения) – это интерфейс взаимодействия между целевым ресурсом и сторонними приложениями. Данный программный интерфейс разрабатывается для интеграции функциональных возможностей целевого ресурса в другие приложения или web-сайты [14].

Доступ к программному интерфейсу приложения (ПИП) каталогов автомобильных запчастей различных поставщиков осуществляется по личному запросу директора организации. Данный запрос подается менеджеру, обслуживающему аккаунт организации. При подтверждении доступа к программному интерфейсу организация получает уникальный ключ. Данный ключ позволяет при помощи запросов программно обращаться к ресурсу от лица организации, получившей соответствующий доступ.

В случае разработки приложения для автоматизации процесса сбора данных, данный ключ необходим для получения программного доступа к каталогам поставщика. Имея ключ доступа к программному интерфейсу

приложения, становится возможным извлечение необходимой информации из каталога. Для работы с ПИП необходимо изучить соответствующую документацию, в которой подробно описаны методы, доступные для использования совместно с ключом доступа [10]. Пример запроса к ПИП изображен на рисунке 8.

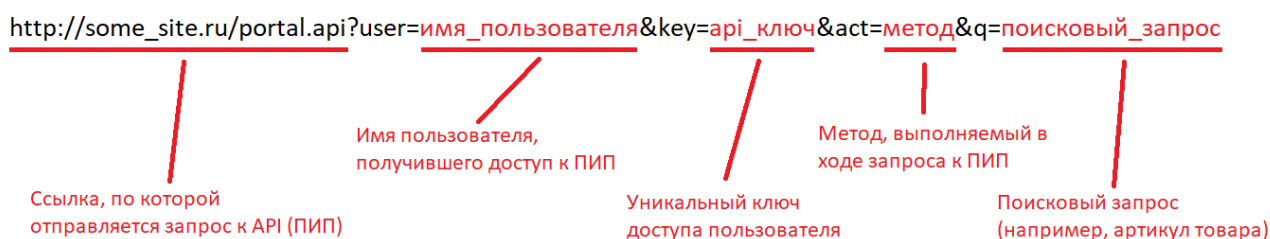


Рисунок 8 – Пример условного запроса к ПИП

В результате выполнения данного запроса возвращается ответ от сервера в виде списка значений формата JSON.

JSON (от англ. JavaScript Object Notation – нотация объекта языка JavaScript) – это текстовый формат обмена данными, основанный на языке программирования JavaScript. Данный формат является независимым текстовым форматом и может быть использован практически любым языком программирования [13]. Пример данного формата изображен на рисунке 9.

```
{
  "name": "Диск тормозной CITROEN BERLINGO +ESP/C5/C4",
  "brand": "NK",
  "stock": 8,
  "delivery": 1,
  "price": "1618,61"
}
```

Рисунок 9 – Пример данных формата JSON

Данные, получаемые в результате запроса к каталогу любого поставщика, приходят в виде наборов ««ключ»: »значение»». Зная названия требуемых ключей, можно извлечь соответствующие им значения. В примере, изображенном на рисунке 9, имеется набор данных одного из товаров существующего каталога. Полю «»price»» соответствует значение ««1618,61»» - стоимость данного товара.

Исходя из вышеперечисленной информации, можно описать принцип работы разрабатываемого приложения: последовательное обращение к API (ПП) каждого сайта поставщика; отправка запроса на получение списка товаров по указанному поисковому запросу (артикулу); обработка полученного списка данных средствами языка программирования, при помощи которого написано приложение; запись обработанных данных в файл-таблицу. По окончании работы приложения должен быть сформирован файл-таблица с информацией, полученной в результате обращения к ПП каждого поставщика.

Следующим шагом необходимо выбрать технологию для реализации приложения.

2.2 Выбор технологии для реализации приложения

В настоящий момент существует множество языков программирования, с помощью которых можно реализовать задуманное приложение. Однако не каждый язык программирования способен дать полноценный спектр инструментов и возможностей для реализации программы.

Для того чтобы определиться с технологией разработки, необходимо провести обзор самых популярных ЯП, подходящих для написания данного рода приложений. Среди них можно выделить два наиболее подходящих языка программирования:

- PHP,
- Python.

PHP – скриптовый язык программирования общего назначения; серверный язык скриптов. Его применяют для разработки web-приложений, в частности – для описания внутренней логики сайтов, а также для работы с базами данных. В настоящее время данный язык программирования поддерживается большинством хостинг-провайдеров и находится в топ-5 самых востребованных ЯП, используемых для web-разработки.

Данный язык обладает обширным спектром инструментов, что позволяет выполнять на нём крупные и серьёзные проекты. В частности, для сбора данных с сайтов используются как встроенные в язык инструменты (DOM или XPath), так и полноценные библиотеки (Simple HTML DOM).

Python – это высокоуровневый язык программирования общего назначения. Он имеет динамическую строгую типизацию данных и умеет автоматически управлять памятью. Python является объектно-ориентированным языком программирования с очень простым синтаксисом, что делает его очень удобным и легким в изучении и использовании [17].

Концепция данного языка программирования заключается в том, что каждый его элемент, будь то переменная, класс, структура, являются объектами с динамической типизацией данных. Это означает, что необязательно явно указывать тип данных того или иного объекта – язык сам понимает, какой тип данных используется при инициализации этого объекта.

Отличительной особенностью данного языка программирования является наличие огромного количества различных инструментов – библиотек. Некоторые из них являются встроенными в язык; другие следует устанавливать самостоятельно. В частности, для сбора данных с сайтов используются такие библиотеки, как BeautifulSoup4, Requests, Selenium, lxml [18].

Для объективного сравнения двух вышеописанных языков программирования, следует описать критерии, по которым они будут сравниваться.

Первым критерием является простота синтаксиса языка программирования. Это довольно важный критерий, поскольку он повлияет как на процесс разработки приложения, так и на его последующее обслуживание.

Вторым по важности критерием является возможность и простота разработки приложения с пользовательским интерфейсом.

В качестве третьего критерия выступит простота работы с данными формата JSON.

Четвертым критерием является быстродействие языка программирования. От скорости выполнения команд зависит скорость сбора данных, а, следовательно, скорость выполнения пользователем задачи.

В качестве пятого критерия выступает перспективность языка программирования. От того как часто будет обновляться язык программирования и насколько востребован он будет в дальнейшем, зависит потенциал улучшения (обновления) разрабатываемой программы в будущем.

Исходя из описанных критериев, можно составить таблицу 1, в которую будут записаны результаты оценки двух языков программирования:

Таблица 1 – Сравнение двух языков программирования

Критерий	Язык программирования	
	PHP	Python
Простой синтаксис	1	1
Работа с пользовательским интерфейсом	0	1
Простая работа с форматом JSON	1	1
Быстродействие	1	1
Перспективность	0	1

По результатам сравнения двух языков программирования, лучшим выбором оказался язык Python. В рамках написания данной ВКР он будет использоваться для разработки приложения по сбору данных. Следующим шагом необходимо описать алгоритм работы разрабатываемого приложения по сбору данных.

2.3 Описание алгоритма работы приложения по сбору данных

Для понимания того, как должно работать приложение по сбору данных, необходимо подробно описать алгоритм его работы.

Разрабатываемое приложение состоит из двух частей: логической и графической.

Логическая часть представляет собой набор функций и алгоритмов, которые непосредственно выполняют задачу программы, а именно – осуществляют запросы к API сайтов-каталогов поставщиков, а также создают итоговый файл-таблицу с данными.

Графическая часть включает в себя пользовательский интерфейс – оболочку взаимодействия пользователя с программой посредством использования устройств ввода-вывода (клавиатура, мышь).

Структуру разрабатываемого приложения можно описать в виде диаграммы компонентов нотации UML [20]. Данная диаграмма изображена на рисунке 10.

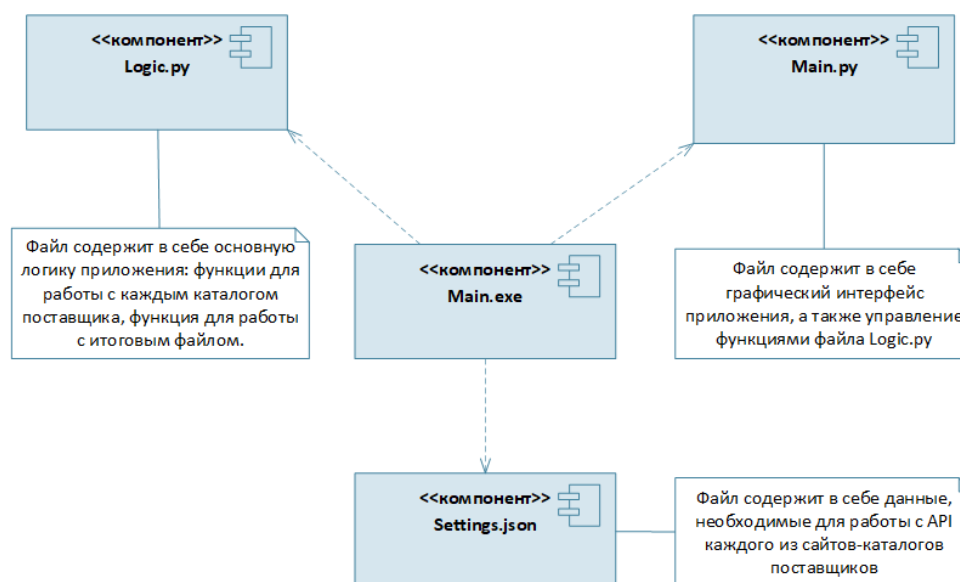


Рисунок 10 – Диаграмма компонентов разрабатываемого приложения

В частности, на данной диаграмме изображен основной компонент – «Main.exe». Он олицетворяет исполняемый файл [12]. Данный файл компилируется из двух файлов – «Logic.py» и «Main.py».

Первый файл – «Logic.py» - является файлом логической части приложения. В нем находятся функции для работы с API сайтов-каталогов поставщиков, а также функция для работы с результирующим файлом [16].

Второй файл – «Main.py» - олицетворяет графическую часть приложения. В нем содержится программный код структуры графического интерфейса приложения, а также обработка событий – взаимодействий пользователя с интерфейсом.

Файл Settings.json служит для хранения упорядоченной информации о каждом сайте-поставщике. Логическая часть обращается к данному файлу для получения следующих персональных данных – ссылка для запроса к API определенного поставщика; имя пользователя аккаунта, работающего с каталогом данного поставщика; уникальный API-ключ, предоставленный определенным поставщиком для организации.

Две данные части – логическая и графическая - тесно связаны между собой. Логическая часть получает информацию от графической – из полей, в которые пользователь ввел какие-либо данные. В результате обработки данной информации логическая часть выполняет свою основную функцию – делает запросы к API сайтов-каталогов поставщиков. По окончании выполнения своей задачи логическая часть обращается к графической, а та, в свою очередь, уведомляет пользователя об окончании работы и ожидает дальнейших указаний.

Взаимодействие между пользователем и двумя частями приложения можно изобразить в виде диаграммы последовательности [11]. На данной диаграмме будет изображено взаимодействие объектов между собой в рамках работы актера – продавца – с приложением. Данная диаграмма изображена на рисунке 11.

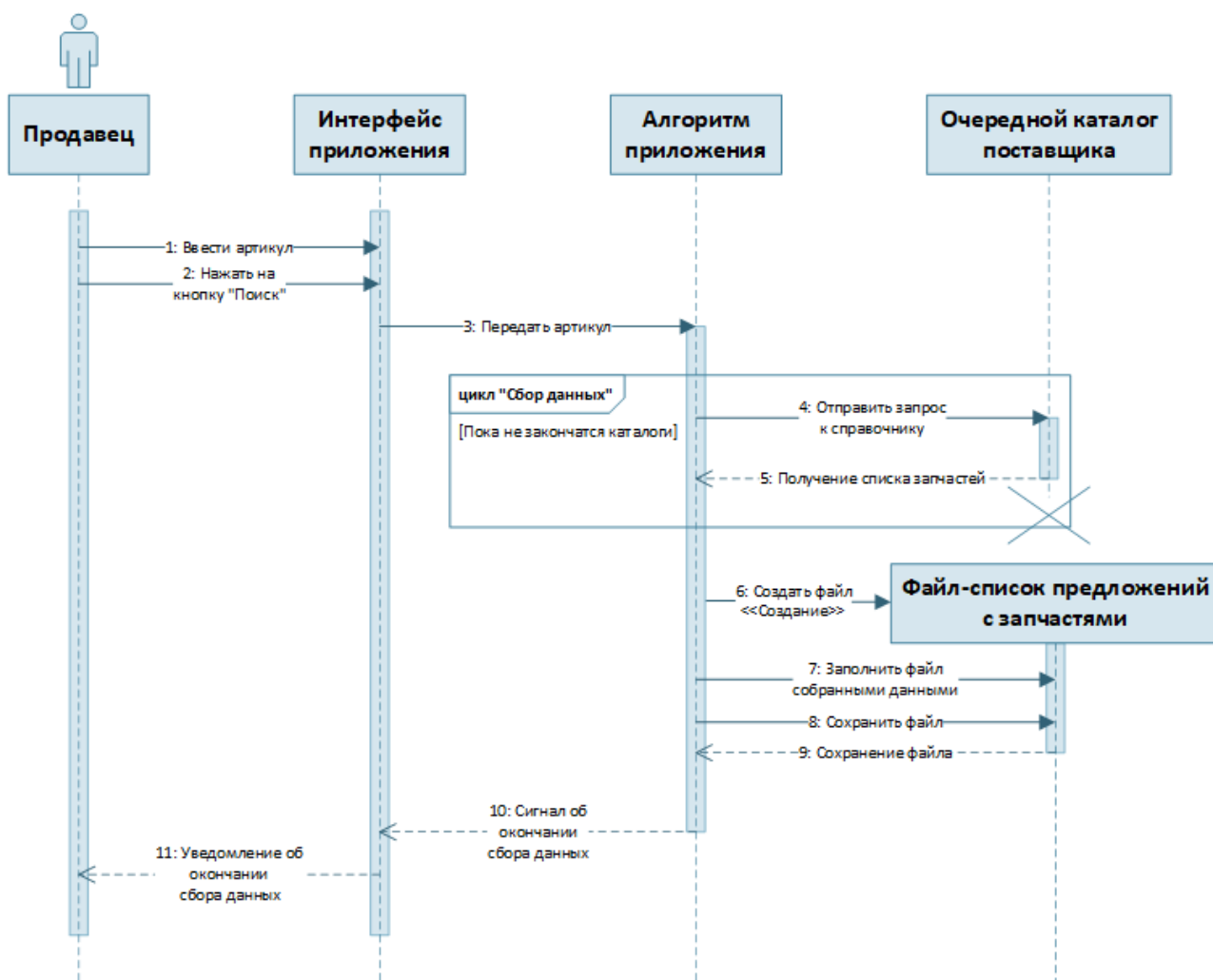


Рисунок 11 – Диаграмма последовательности процесса работы продавца с приложением

В процессе работы с приложением продавец наблюдает интерфейс – графическую часть приложения по сбору данных. Логическая часть подключается непосредственно после совершения продавцом некоторых действий. В результате по окончании работы логической части приложение оповещает продавца о выполненной задаче через графическую часть.

Цикл «Сбор данных», изображенный на рисунке 11, работает по принципу последовательного выполнения прописанных в программном коде функций. Данные функции схожи по алгоритму, но имеют разный подход к каждому конкретному API сайта-каталога поставщика. Общий принцип работы

данных функций можно изобразить в виде диаграммы деятельности [8]. Данная диаграмма изображена на рисунке 12.

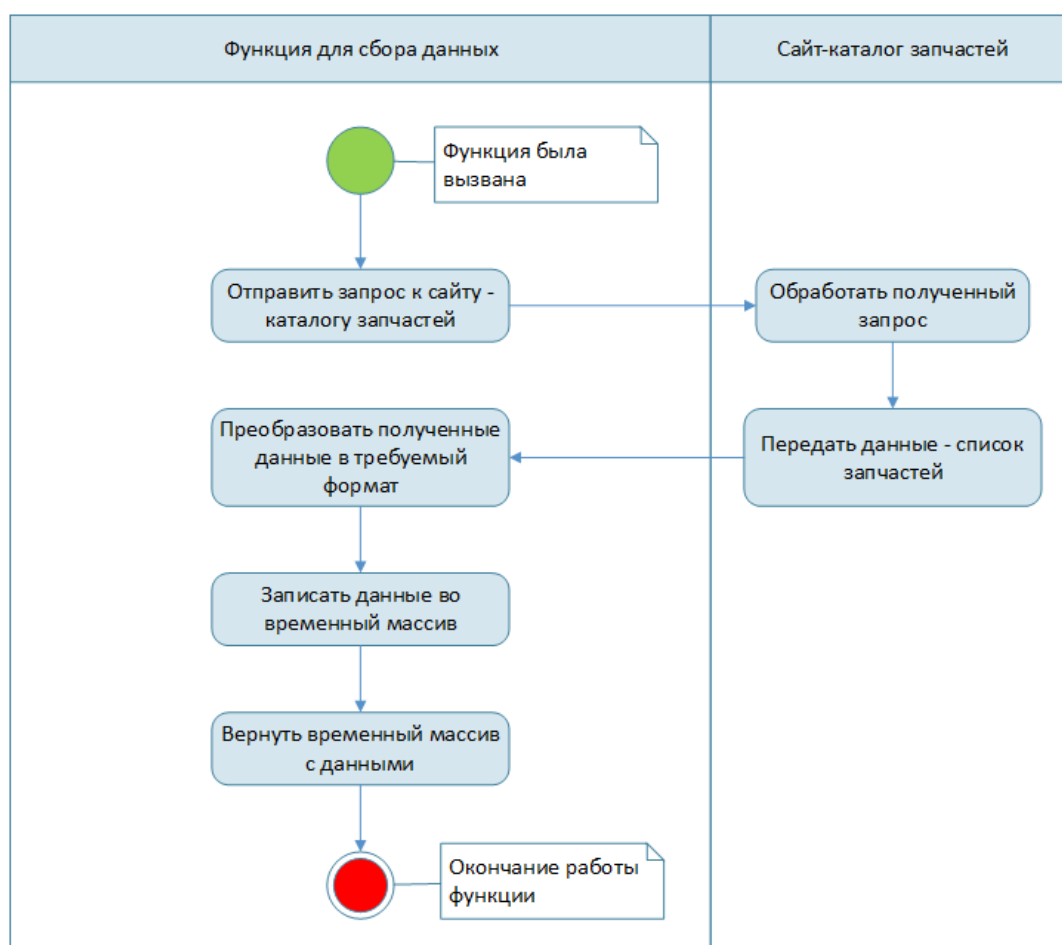


Рисунок 12 – Диаграмма деятельности функций для работы с API

В целом, каждая функция работает по одинаковому принципу. Разница между ними заключается в различиях между запросами к каждому конкретному API сайта-каталога поставщика.

Исходя из всего вышеуказанного, можно сформулировать алгоритм работы пользователя с приложением. В первую очередь пользователю необходимо запустить приложение, вставить заранее скопированный артикул требуемой запчасти в соответствующее поле ввода и нажать на кнопку «Поиск». Также пользователь имеет возможность ввести имя будущего файла – это не обязательное действие и выполняется исключительно по желанию

пользователя. После нажатия на кнопку «Поиск» сработает обработчик события нажатия на данную кнопку и запустит скрипт с логической частью программы. После того, как программа получит все необходимые данные и создаст соответствующий файл, кнопка «Открыть файл» станет активной. Такое поведение кнопки дает пользователю понять, что файл создан и его можно открыть. По нажатию на данную кнопку будет открыт созданный файл с данными – списком найденных запчастей [1].

Результатом работы программы является файл формата «.csv». Данный формат является аналогом формата «.xlsx», но имеет над ним ряд преимуществ. Одним из весомых преимуществ является возможность настройки отображения данных в корректном виде непосредственно при запуске файла.

После описания алгоритма работы приложения можно приступить к описанию его практической реализации.

2.4 Практическая реализации приложения по сбору данных

В результате работы над данным приложением получилось разработать простую и функциональную утилиту с лаконичным и минималистичным дизайном пользовательского интерфейса [15]. Окно разработанной программы изображено на рисунке 13.

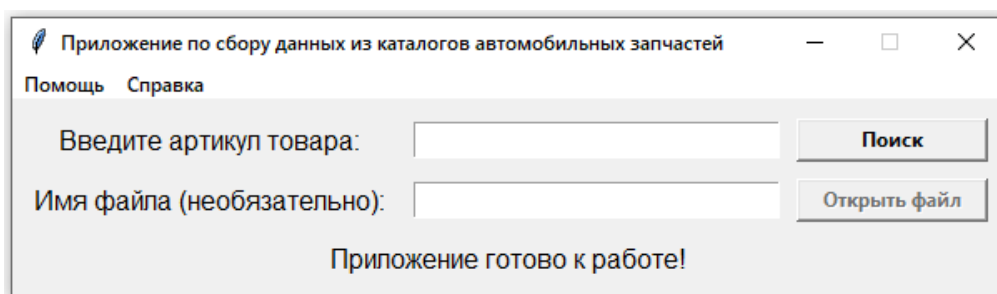


Рисунок 13 – Интерфейс приложения по сбору данных

Как уже было упомянуто в параграфе 2.3 текущей ВКР, программа состоит из двух частей: графической и логической. Каждой части соответствует файл с программным кодом. К разработанному приложению относятся три файла:

- «Main.py»,
- «Logic.py»,
- «Settings.json».

Следует подробнее остановиться на каждом из данных файлов.

В файле «Main.py» прописан графический интерфейс приложения и основной сценарий взаимодействия интерфейса приложения с пользователем [4]. Следует более подробно рассмотреть структуру данного файла.

В первую очередь подключаются необходимые для работы библиотеки и модули. Список подключаемых инструментов изображен на рисунке 14.

```
1 import tkinter as tk # Библиотека для работы с графическим интерфейсом
2 from tkinter.messagebox import showinfo # Дополнение графического интерфейса - всплывающие окна
3 import os # Библиотека для работы с операционной системой
4 from time import sleep # Библиотека для искусственного ограничения отправки запросов
5 import threading # Библиотека для корректной работы кнопок
6 import logic # Файл с основными функциями для работы программы
```

Рисунок 14 – Подключаемые библиотеки и модули файла «Main.py»

В частности, для создания графического интерфейса программы служит библиотека «tkinter», подключаемая в самом начале данного файла. Затем следует импорт метода showinfo() из модуля «messagebox» библиотеки «tkinter» – данный метод позволяет программе генерировать всплывающие окна. Для работы с операционной системой компьютера импортируется библиотека «os». Метод sleep() библиотеки «time» требуется для установки временного ограничения отправки запросов к API. Библиотека threading служит для корректной работы быстрых команд (Ctrl+C – копирование, Ctrl+V – вставка и

т.д.). В самую последнюю очередь импортируется файл «Logic.py», содержащий основную логику программы [19].

На очереди идет функция `product_search()`. Она является связующим звеном между логической и графической частями приложения и вызывается по нажатию на кнопку «Поиск». Код данной функции изображен на рисунке А.1 приложения А.

Данная функция первым делом получает артикул, введенный в соответствующее ему поле ввода. Если пользователь не ввел артикул в поле для ввода и нажал на кнопку «Поиск», поле ввода артикула окрасится в красный цвет и строка состояния укажет на ошибку. Интерфейс приложения при возникновении данной ошибки изображен на рисунке 15.

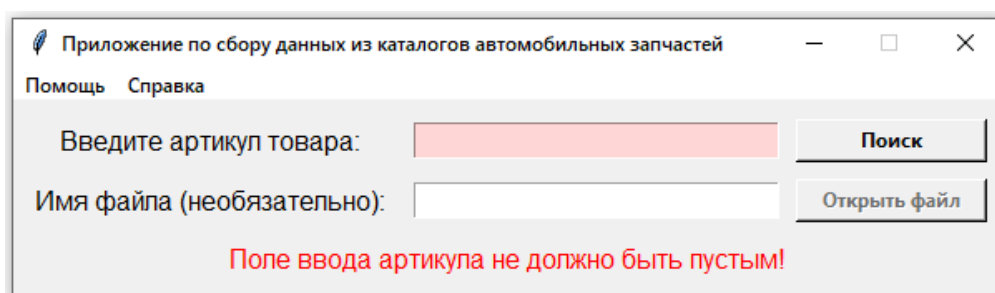


Рисунок 15 – Интерфейс приложения при попытке «пустого поиска»

Далее следует еще одна валидация – на этот раз проверка на корректность введенного пользователем артикула. Данный тип проверки требуется во избежание ввода пользователем неверных данных. Если пользователь ввел некорректный артикул и нажал на кнопку «Поиск», поле ввода артикула окрасится в красный цвет и строка состояния укажет на данную ошибку. Интерфейс приложения при возникновении данной ошибки изображен на рисунке 16.

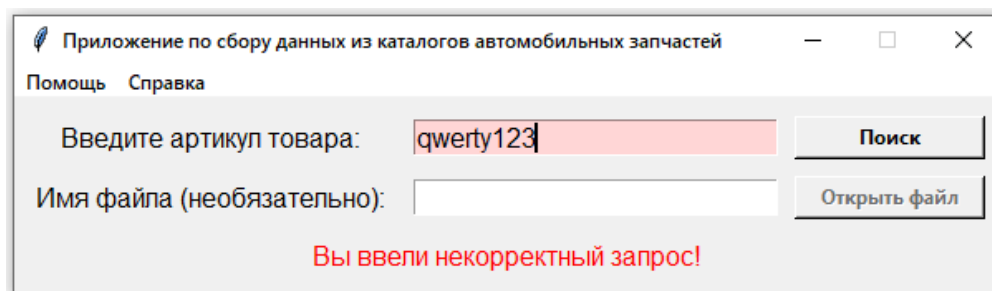


Рисунок 16 – Интерфейс приложения при вводе пользователем некорректного запроса

В том случае, когда введенные пользователем данные успешно прошли проверку, следующим шагом будет начало процесса сбора данных. Из файла «Logic.py» поочередно вызываются функции, соответствующие разным сайтам-каталогам поставщиков. В качестве аргумента передается введенный пользователем артикул детали [3]. Пример одной из подобных функций по обращению к API сайта-каталога поставщика представлен на рисунке Б.1 приложения Б.

Результатом работы каждой функции по сбору данных является список с позициями каталога запчастей каждого поставщика. Данные списки соединяются в один большой список и сортируются по требуемым параметрам: сроку поставки и цене [5].

Следующим действием в программном коде является проверка на введенное пользователем имя файла: если поле не пустое, то создается файл с именем содержимого данного поля; в противном случае именем файла будет являться артикул, введенный пользователем в соответствующее поле ввода. В обоих случаях вызывается метод `create_file()` из файла «Logic.py», в качестве аргументов в него передается итоговый список с позициями всех каталогов и имя файла. Данный метод создает файл, наполняет его полученными данными и сохраняет по пути расположения самой программы [2]. Программный код функции `create_file()` из файла «Logic.py» изображен на рисунке 17.


```

91 # Функция для создания и наполнения итогового файла
92 # Возвращает абсолютный путь к созданному файлу (нужен для обработчика кнопки "Открыть файл")
93 def create_file(data, name):
94     file_name = f'{name}.csv' # Название будущего файла
95     base_dir = os.path.abspath(os.path.dirname(__file__)) # Нахождение абсолютного пути расположения файла
96     full_path_to_file = os.path.join(base_dir, file_name) # Конкатенация абсолютного пути с именем файла
97     # Проверка на существование файла с таким же названием
98     i = 1
99     while os.path.exists(file_name):
100         file_name = f'{name}({i}).csv'
101         i += 1
102     # Создание файла для записи
103     with open(file_name, 'w', newline='', encoding='utf8') as file:
104         record = csv.writer(file)
105         # Запись в первую строку файла названий колонок
106         record.writerow(('Поставщик', 'Наименование', 'Бренд', 'На складе, шт.', 'Склад',
107                         'Срок поставки, дн.', 'Цена', 'Ссылка'))
108         # Построчная запись всех данных в файл
109         for ad in data:
110             record.writerow((ad['provider'], ad['name'], ad['brand'], ad['stock'], ad['storage'],
111                             ad['delivery'], ad['price'], ad['href']))
112     return full_path_to_file

```

Рисунок 17 – Функция для работы с итоговым файлом

После того, как приложение соберет все данные и создаст итоговый файл, графический интерфейс сообщит об этом пользователю при помощи строки состояния. В частности, при успешно выполненном сборе данных в строке состояния приложения появится сообщение «Готово» зеленого цвета. Также станет доступна для нажатия кнопка «Открыть файл». Интерфейс приложения при успешном выполнении запроса изображен на рисунке 18.

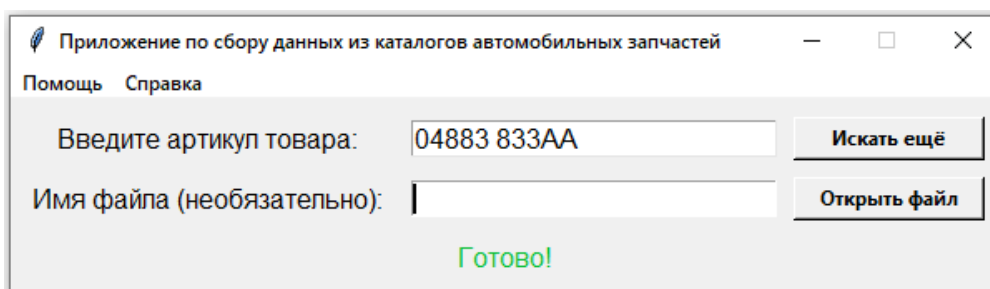


Рисунок 18 – Интерфейс приложения при успешном окончании сбора данных

По нажатию на ставшую активной кнопку «Открыть файл», откроется созданный файл с данными – результатом сбора данных. Нажатие на данную кнопку вызывает функцию `open_result_file()`. Программный код данной функции изображен на рисунке 19.

```
73 # Функция открытия файла, вызывается при нажатии на кнопку "Открыть файл"
74 # Ничего не возвращает
75 def open_result_file():
76     os.startfile(full_path_to_file)
```

Рисунок 19 – Функция для открытия результирующего файла

В качестве дополнительных функциональных возможностей приложения были добавлены окно помощи и справочное окно с информацией о приложении – функции `help_window()` и `about_program()`. Данные функции являются информирующими и используются по нажатию на соответствующие им кнопки в меню приложения. Программный код функции `help_window()` изображен на рисунке 20.

```
48 # Функция вывода окна помощи
49 # Ничего не возвращает
50 def help_window():
51     showinfo(title='Помощь', message='Порядок работы с приложением:\n'
52                                     '1) На сайте Exist.ru найти искомый код запчасти и скопировать его\n'
53                                     '2) Вставить код в поле ввода артикула\n'
54                                     '3) По желанию, можно ввести в нижнее поле название будущего файла\n'
55                                     '4) Нажать кнопку "Поиск"\n'
56                                     '5) По окончании работы программы станет активной кнопка "Открыть файл".\n'
57                                     'Нажав на нее, откроется созданный файл формата ".csv". Данный формат\n'
58                                     'открывается при помощи программы Libre Office Calc.')
```

Рисунок 20 – Функция для открытия окна помощи приложения

При срабатывании данной функции открывается окно помощи. Данное окно изображено на рисунке 21.

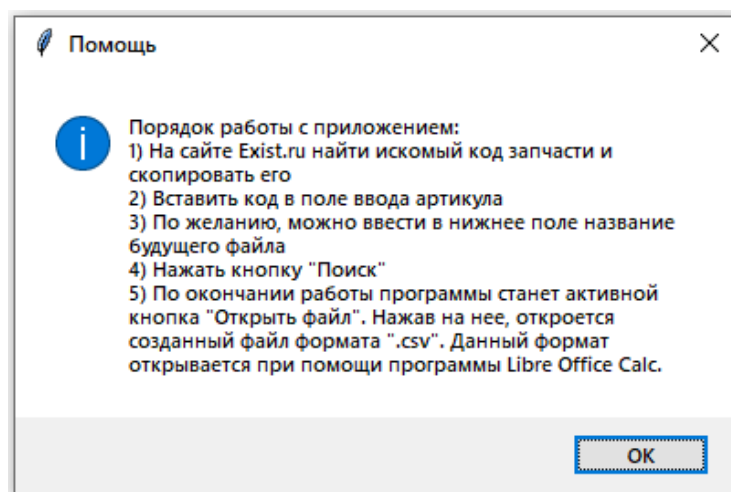


Рисунок 21 – Окно помощи в работе с приложением

Программный код функции, вызывающей справочное окно, представлен на рисунке 22.

```

61  # Функция вывода окна с информацией о программе
62  # Ничего не возвращает
63  def about_program():
64      showinfo(title='Справка', message='Приложение разработано в рамках бакалаврской работы.\n\n'
65      'Язык программирования: Python (версия 3.9)\n\n'
66      'Используемые технологии:\n'
67      ' - библиотека "Requests" для отправки запросов\n'
68      ' - библиотека "json" для работы с данными формата JSON\n'
69      ' - библиотека "csv" для работы с файлами-таблицами\n'
70      ' - библиотека "Tkinter" для создания графического интерфейса утилиты\n'
71      ' - API-ключи сайтов-поставщиков организации\n\n'
72      'Среда разработки: PyCharm 2021.1.1\n\n'
73      'Разработчик: Лукичев Павел, студент группы ПИп-1703а\n'
74      'Для связи: e1lonorix@gmail.com')

```

Рисунок 22 – Функция для открытия справочного окна приложения

По аналогии с предыдущей функцией, открывающей окно помощи, данная функция открывает справочное окно, в котором написана основная информация о приложении, разработчике, а также использованных при разработке технологиях. Данное окно изображено на рисунке 23.

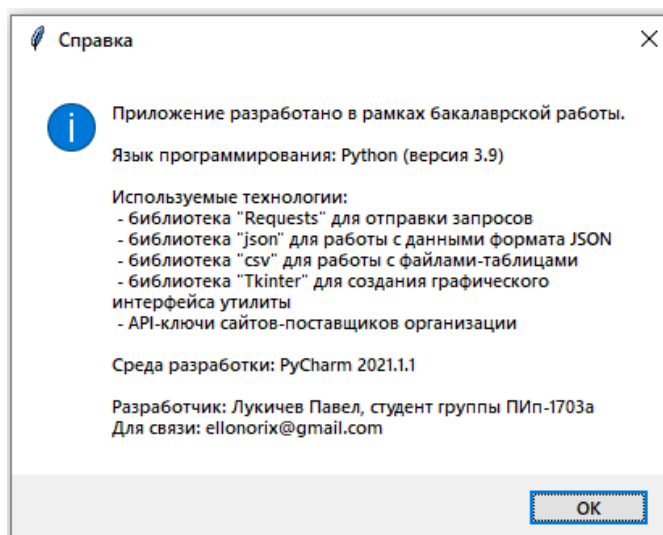


Рисунок 23 – Окно справочной информации о приложении

Следующая функция – `on_key_release()`. Её основная задача заключается в исправлении ошибки, при которой было невозможно взаимодействовать с полями ввода приложения при помощи горячих клавиш клавиатуры – программа реагировала только на английскую раскладку клавиатуры, что является достаточно неудобным в использовании. Программный код данной функции изображен на рисунке 24.

```
79  # Функция для работы "копирования", "вставки" и "вырезать" на русской раскладке
80  # Ничего не возвращает
81  def on_key_release(event):
82      ctrl = (event.state & 0x4) != 0
83      if event.keycode == 88 and ctrl and event.keysym.lower() != "x":
84          event.widget.event_generate("<<Cut>>")
85
86      if event.keycode == 86 and ctrl and event.keysym.lower() != "v":
87          event.widget.event_generate("<<Paste>>")
88
89      if event.keycode == 67 and ctrl and event.keysym.lower() != "c":
90          event.widget.event_generate("<<Copy>>")
```

Рисунок 24 – Функция для корректной работы горячих клавиш русской раскладки клавиатуры

Далее в программном коде описывается стартовая точка работы приложения – графический интерфейс. На рисунке 25 показаны настройки окна программы.

```
97 ▶ if __name__ == '__main__':
98     window = tk.Tk() # Создание объекта графического интерфейса
99     window.title('Приложение по сбору данных из каталогов автомобильных запчастей') # Заголовок (название) окна
100    # Расположение окна в пространстве (на экране)
101    x = (window.winfo_screenwidth() - window.winfo_reqwidth()-400) / 2 # Коорд. по horiz. со сдвигом на размер окна
102    y = (window.winfo_screenheight() - window.winfo_reqheight()) / 2 # Коорд. по вертикали
103    window.wm_geometry("%d+%d" % (x, y)) # Установка координат
104    window.resizable(width=tk.FALSE, height=tk.FALSE) # Запрет на изменение размеров окна
105    window.bind_all("<Key>", on_key_release, "+") # Строка для работы функций "копир/вставка/вырезки" на ру раскладке
```

Рисунок 25 – Основные настройки окна программы

В данные настройки входят: инициализация объекта графического интерфейса, название окна программы, размеры окна, его расположение на экране (по центру), а также настройка горячих клавиш на русской раскладке клавиатуры.

Следом располагаются строки программного кода, отвечающие за размещение элементов пользовательского интерфейса – кнопок, меню, полей ввода. На рисунке 26 изображен программный код первой строки элементов – текстовой надписи «Введите артикул товара:», поля ввода артикула, кнопки «Поиск».

```
103    # Текстовая надпись "Введите артикул товара:"
104    Label_Article_var = tk.StringVar()
105    Label_Article = tk.Label(text='Введите артикул товара: ', font=12)
106    Label_Article.grid(row=0, column=0, padx=(10, 5), pady=(10, 5), sticky="we")
107
108    # Поле ввода артикула
109    Entry_Article = tk.Entry(textvariable=Label_Article_var, width=24, font=12)
110    Entry_Article.grid(row=0, column=1, padx=5, pady=(10, 5))
111
112    # Кнопка "Поиск"
113    Button_Search = tk.Button(text='Поиск', width=15,
114                             |
115                             |
116                             |
117                             |
118                             |
119                             |
120                             |
121                             |
122                             |
123                             |
124                             |
125                             |
126                             |
127                             |
128                             |
129                             |
130                             |
131                             |
132                             |
133                             |
134                             |
135                             |
136                             |
137                             |
138                             |
139                             |
140                             |
141                             |
142                             |
143                             |
144                             |
145                             |
146                             |
147                             |
148                             |
149                             |
150                             |
151                             |
152                             |
153                             |
154                             |
155                             |
156                             |
157                             |
158                             |
159                             |
160                             |
161                             |
162                             |
163                             |
164                             |
165                             |
166                             |
167                             |
168                             |
169                             |
170                             |
171                             |
172                             |
173                             |
174                             |
175                             |
176                             |
177                             |
178                             |
179                             |
180                             |
181                             |
182                             |
183                             |
184                             |
185                             |
186                             |
187                             |
188                             |
189                             |
190                             |
191                             |
192                             |
193                             |
194                             |
195                             |
196                             |
197                             |
198                             |
199                             |
200                             |
201                             |
202                             |
203                             |
204                             |
205                             |
206                             |
207                             |
208                             |
209                             |
210                             |
211                             |
212                             |
213                             |
214                             |
215                             |
216                             |
217                             |
218                             |
219                             |
220                             |
221                             |
222                             |
223                             |
224                             |
225                             |
226                             |
227                             |
228                             |
229                             |
230                             |
231                             |
232                             |
233                             |
234                             |
235                             |
236                             |
237                             |
238                             |
239                             |
240                             |
241                             |
242                             |
243                             |
244                             |
245                             |
246                             |
247                             |
248                             |
249                             |
250                             |
251                             |
252                             |
253                             |
254                             |
255                             |
256                             |
257                             |
258                             |
259                             |
260                             |
261                             |
262                             |
263                             |
264                             |
265                             |
266                             |
267                             |
268                             |
269                             |
270                             |
271                             |
272                             |
273                             |
274                             |
275                             |
276                             |
277                             |
278                             |
279                             |
280                             |
281                             |
282                             |
283                             |
284                             |
285                             |
286                             |
287                             |
288                             |
289                             |
290                             |
291                             |
292                             |
293                             |
294                             |
295                             |
296                             |
297                             |
298                             |
299                             |
300                             |
301                             |
302                             |
303                             |
304                             |
305                             |
306                             |
307                             |
308                             |
309                             |
310                             |
311                             |
312                             |
313                             |
314                             |
315                             |
316                             |
317                             |
318                             |
319                             |
320                             |
321                             |
322                             |
323                             |
324                             |
325                             |
326                             |
327                             |
328                             |
329                             |
330                             |
331                             |
332                             |
333                             |
334                             |
335                             |
336                             |
337                             |
338                             |
339                             |
340                             |
341                             |
342                             |
343                             |
344                             |
345                             |
346                             |
347                             |
348                             |
349                             |
350                             |
351                             |
352                             |
353                             |
354                             |
355                             |
356                             |
357                             |
358                             |
359                             |
360                             |
361                             |
362                             |
363                             |
364                             |
365                             |
366                             |
367                             |
368                             |
369                             |
370                             |
371                             |
372                             |
373                             |
374                             |
375                             |
376                             |
377                             |
378                             |
379                             |
380                             |
381                             |
382                             |
383                             |
384                             |
385                             |
386                             |
387                             |
388                             |
389                             |
390                             |
391                             |
392                             |
393                             |
394                             |
395                             |
396                             |
397                             |
398                             |
399                             |
400                             |
401                             |
402                             |
403                             |
404                             |
405                             |
406                             |
407                             |
408                             |
409                             |
410                             |
411                             |
412                             |
413                             |
414                             |
415                             |
416                             |
417                             |
418                             |
419                             |
420                             |
421                             |
422                             |
423                             |
424                             |
425                             |
426                             |
427                             |
428                             |
429                             |
430                             |
431                             |
432                             |
433                             |
434                             |
435                             |
436                             |
437                             |
438                             |
439                             |
440                             |
441                             |
442                             |
443                             |
444                             |
445                             |
446                             |
447                             |
448                             |
449                             |
450                             |
451                             |
452                             |
453                             |
454                             |
455                             |
456                             |
457                             |
458                             |
459                             |
460                             |
461                             |
462                             |
463                             |
464                             |
465                             |
466                             |
467                             |
468                             |
469                             |
470                             |
471                             |
472                             |
473                             |
474                             |
475                             |
476                             |
477                             |
478                             |
479                             |
480                             |
481                             |
482                             |
483                             |
484                             |
485                             |
486                             |
487                             |
488                             |
489                             |
490                             |
491                             |
492                             |
493                             |
494                             |
495                             |
496                             |
497                             |
498                             |
499                             |
500                             |
501                             |
502                             |
503                             |
504                             |
505                             |
506                             |
507                             |
508                             |
509                             |
510                             |
511                             |
512                             |
513                             |
514                             |
515                             |
516                             |
517                             |
518                             |
519                             |
520                             |
521                             |
522                             |
523                             |
524                             |
525                             |
526                             |
527                             |
528                             |
529                             |
530                             |
531                             |
532                             |
533                             |
534                             |
535                             |
536                             |
537                             |
538                             |
539                             |
540                             |
541                             |
542                             |
543                             |
544                             |
545                             |
546                             |
547                             |
548                             |
549                             |
550                             |
551                             |
552                             |
553                             |
554                             |
555                             |
556                             |
557                             |
558                             |
559                             |
560                             |
561                             |
562                             |
563                             |
564                             |
565                             |
566                             |
567                             |
568                             |
569                             |
570                             |
571                             |
572                             |
573                             |
574                             |
575                             |
576                             |
577                             |
578                             |
579                             |
580                             |
581                             |
582                             |
583                             |
584                             |
585                             |
586                             |
587                             |
588                             |
589                             |
590                             |
591                             |
592                             |
593                             |
594                             |
595                             |
596                             |
597                             |
598                             |
599                             |
600                             |
601                             |
602                             |
603                             |
604                             |
605                             |
606                             |
607                             |
608                             |
609                             |
610                             |
611                             |
612                             |
613                             |
614                             |
615                             |
616                             |
617                             |
618                             |
619                             |
620                             |
621                             |
622                             |
623                             |
624                             |
625                             |
626                             |
627                             |
628                             |
629                             |
630                             |
631                             |
632                             |
633                             |
634                             |
635                             |
636                             |
637                             |
638                             |
639                             |
640                             |
641                             |
642                             |
643                             |
644                             |
645                             |
646                             |
647                             |
648                             |
649                             |
650                             |
651                             |
652                             |
653                             |
654                             |
655                             |
656                             |
657                             |
658                             |
659                             |
660                             |
661                             |
662                             |
663                             |
664                             |
665                             |
666                             |
667                             |
668                             |
669                             |
670                             |
671                             |
672                             |
673                             |
674                             |
675                             |
676                             |
677                             |
678                             |
679                             |
680                             |
681                             |
682                             |
683                             |
684                             |
685                             |
686                             |
687                             |
688                             |
689                             |
690                             |
691                             |
692                             |
693                             |
694                             |
695                             |
696                             |
697                             |
698                             |
699                             |
700                             |
701                             |
702                             |
703                             |
704                             |
705                             |
706                             |
707                             |
708                             |
709                             |
710                             |
711                             |
712                             |
713                             |
714                             |
715                             |
716                             |
717                             |
718                             |
719                             |
720                             |
721                             |
722                             |
723                             |
724                             |
725                             |
726                             |
727                             |
728                             |
729                             |
730                             |
731                             |
732                             |
733                             |
734                             |
735                             |
736                             |
737                             |
738                             |
739                             |
740                             |
741                             |
742                             |
743                             |
744                             |
745                             |
746                             |
747                             |
748                             |
749                             |
750                             |
751                             |
752                             |
753                             |
754                             |
755                             |
756                             |
757                             |
758                             |
759                             |
760                             |
761                             |
762                             |
763                             |
764                             |
765                             |
766                             |
767                             |
768                             |
769                             |
770                             |
771                             |
772                             |
773                             |
774                             |
775                             |
776                             |
777                             |
778                             |
779                             |
780                             |
781                             |
782                             |
783                             |
784                             |
785                             |
786                             |
787                             |
788                             |
789                             |
790                             |
791                             |
792                             |
793                             |
794                             |
795                             |
796                             |
797                             |
798                             |
799                             |
800                             |
801                             |
802                             |
803                             |
804                             |
805                             |
806                             |
807                             |
808                             |
809                             |
810                             |
811                             |
812                             |
813                             |
814                             |
815                             |
816                             |
817                             |
818                             |
819                             |
820                             |
821                             |
822                             |
823                             |
824                             |
825                             |
826                             |
827                             |
828                             |
829                             |
830                             |
831                             |
832                             |
833                             |
834                             |
835                             |
836                             |
837                             |
838                             |
839                             |
840                             |
841                             |
842                             |
843                             |
844                             |
845                             |
846                             |
847                             |
848                             |
849                             |
850                             |
851                             |
852                             |
853                             |
854                             |
855                             |
856                             |
857                             |
858                             |
859                             |
860                             |
861                             |
862                             |
863                             |
864                             |
865                             |
866                             |
867                             |
868                             |
869                             |
870                             |
871                             |
872                             |
873                             |
874                             |
875                             |
876                             |
877                             |
878                             |
879                             |
880                             |
881                             |
882                             |
883                             |
884                             |
885                             |
886                             |
887                             |
888                             |
889                             |
890                             |
891                             |
892                             |
893                             |
894                             |
895                             |
896                             |
897                             |
898                             |
899                             |
900                             |
901                             |
902                             |
903                             |
904                             |
905                             |
906                             |
907                             |
908                             |
909                             |
910                             |
911                             |
912                             |
913                             |
914                             |
915                             |
916                             |
917                             |
918                             |
919                             |
920                             |
921                             |
922                             |
923                             |
924                             |
925                             |
926                             |
927                             |
928                             |
929                             |
930                             |
931                             |
932                             |
933                             |
934                             |
935                             |
936                             |
937                             |
938                             |
939                             |
940                             |
941                             |
942                             |
943                             |
944                             |
945                             |
946                             |
947                             |
948                             |
949                             |
950                             |
951                             |
952                             |
953                             |
954                             |
955                             |
956                             |
957                             |
958                             |
959                             |
960                             |
961                             |
962                             |
963                             |
964                             |
965                             |
966                             |
967                             |
968                             |
969                             |
970                             |
971                             |
972                             |
973                             |
974                             |
975                             |
976                             |
977                             |
978                             |
979                             |
980                             |
981                             |
982                             |
983                             |
984                             |
985                             |
986                             |
987                             |
988                             |
989                             |
990                             |
991                             |
992                             |
993                             |
994                             |
995                             |
996                             |
997                             |
998                             |
999                             |
1000                            |
```

Рисунок 26 – Программный код первой строки элементов графического интерфейса

На рисунке 27 изображен программный код второй строки элементов – текстовой надписи «Имя файла (необязательно):», поля ввода имени файла, кнопки «Открыть файл».

```
117 # Надпись "Название файла" и его поле ввода
118 Label_Filename_var = tk.StringVar()
119 Label_Filename = tk.Label(text='Имя файла (необязательно): ', font=12)
120 Label_Filename.grid(row=1, column=0, padx=(10, 5), pady=(5, 10), sticky="w")
121
122 # Поле ввода имени файла
123 Entry_Filename = tk.Entry(textvariable=Label_Filename_var, width=24, font=12)
124 Entry_Filename.grid(row=1, column=1, padx=5, pady=(5, 10))
125
126 # Кнопка "Открыть файл"
127 Button_ToFile = tk.Button(text='Открыть файл', width=15, state=tk.DISABLED, command=open_result_file)
128 Button_ToFile.grid(row=1, column=2, padx=(5, 10), pady=(5, 10))
```

Рисунок 27 – Программный код второй строки элементов графического интерфейса

Закрывающими строками программного кода файла «Main.py» являются команды, отвечающие за отображение строки состояния приложения, а также отображение кнопок меню приложения - «Помощь» и «Справка». Данный программный код изображен на рисунке 28.

```
134 # Строка отслеживания состояния программы
135 Label_Status = tk.Label(text='Приложение готово к работе!', font=12)
136 Label_Status.grid(row=2, column=0, columnspan=3, pady=(0, 10))
137
138 # Верхняя панель
139 main_menu = tk.Menu()
140 main_menu.add_command(label='Помощь', command=help_window)
141 main_menu.add_command(label='Справка', command=about_program)
142 window.config(menu=main_menu)
143
144 window.mainloop()
```

Рисунок 28 – Программный код для отображения строки состояния приложения и меню приложения

Последним файлом, принимающим участие в работе приложения, является файл «Settings.json». Данный файл содержит информацию о каждом сайте поставщика. В частности, речь идет об информации, связанной с доступом к API сайта-каталога поставщика: родительская ссылка для генерации API-запроса, имя пользователя и API-ключ, предоставляемый третьим лицом – технической поддержкой каждого из поставщиков. Вся информация хранится в формате «“ключ”: ”значение”». Пример информации от конкретного поставщика изображен на рисунке 29.

```
"Moskvorechie":  
{  
  "site": "http://portal.moskvorechie.ru/portal.api?",  
  "username": "stat[REDACTED]",  
  "API_KEY": "P0Hm[REDACTED]0d"  
},
```

Рисунок 29 – Пример хранимой информации с данными о сайте поставщика «Москворечье»

Информация, хранимая в файле «Settings.json», используется непосредственно в функциях файла «Logic.py» при сборе данных с конкретного сайта-каталога поставщика.

В данном параграфе была описана реализация приложения по сбору данных. В дальнейшем будет проведено тестирование разработанного приложения в реальных условиях.

Выводы по главе 2

Во второй главе текущей ВКР была рассмотрена непосредственная реализация приложения по сбору данных.

Первым делом был описан принцип работы со средствами для сбора данных. В частности, было детально описано понятие «API» и способы

взаимодействия с ним; был описан формат данных «JSON»; описан общий принцип работы приложения.

На следующем шаге была выбрана технология для реализации приложения по сбору данных. Выбранной технологией оказался язык программирования Python.

Следующим шагом было проведено описание алгоритма работы разрабатываемого приложения, в ходе которого были составлены соответствующие UML-диаграммы для визуализации приложения: диаграмма компонентов, диаграмма последовательности и диаграмма деятельности.

Заключительным шагом стала практическая реализация разрабатываемого приложения. В ходе данного параграфа был детально разобран программный код приложения, показаны различные варианты его работы, а также была частично затронута тема тестирования приложения. Более подробно тема тестирования разработанного приложения будет разобрана в следующей главе настоящей ВКР.

Глава 3 Тестирование разработанного приложения по сбору данных

В данной главе текущей ВКР будет рассмотрен процесс тестирования разработанного приложения. В частности, тестирование будет проводиться непосредственно в рамках рабочей деятельности магазина автомобильных запчастей.

Для тестирования разработанного приложения сотрудником магазина автомобильных запчастей – продавцом - было произведено несколько тестовых запросов по сбору данных.

В качестве демонстрационного примера работы приложения был взят пример поиска запчасти – передний тормозной диск для автомобиля «Chevrolet Lacetti седан 1.6». На сайте-справочнике был найден и скопирован артикул требуемой запчасти. Далее было запущено приложение по сбору данных. В соответствующие поля ввода была помещена необходимая информация – артикул в поле ввода артикула, название файла указано как «Диск торм. Lacetti седан». Окно приложения изображено на рисунке 30.

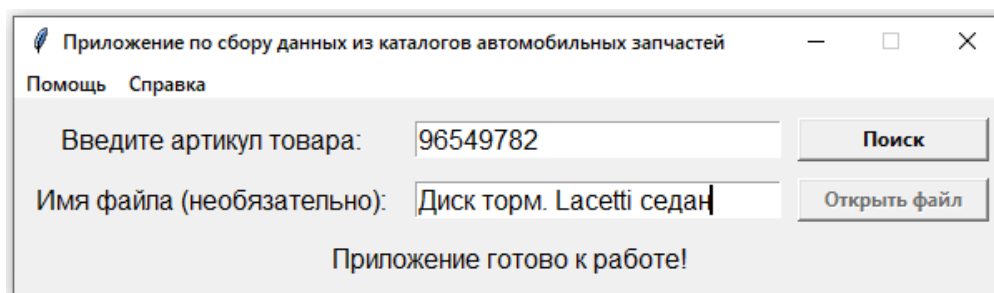


Рисунок 30 – Окно приложения во время тестирования

По нажатии на кнопку «Поиск» приложение было запущено в работу. На рисунке 31 изображено окно приложения через 10 секунд после запуска приложения в работу.

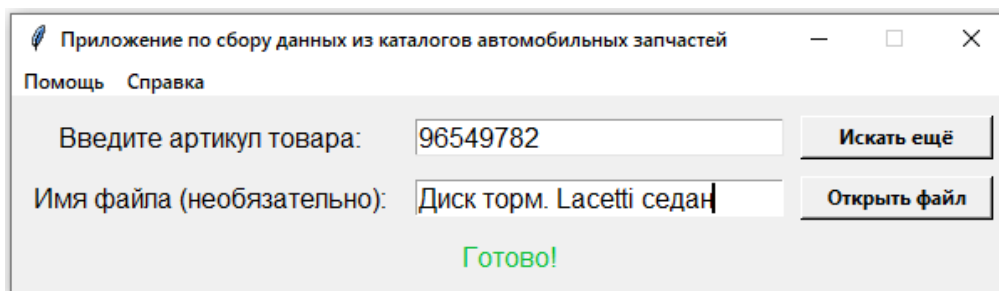


Рисунок 31 – Окно приложения после выполненного сбора данных во время тестирования

Исходя из сообщения в строке состояния приложения, а также ставшей активной кнопки «Открыть файл», можно сделать вывод о том, что приложение успешно справилось со своей задачей – сбором данных.

Нажав на кнопку «Открыть файл», открылось диалоговое окно программы Libre Office Calc. Оно служит для первичной настройки отображения данных в таблице. Данное диалоговое окно изображено на рисунке 32.

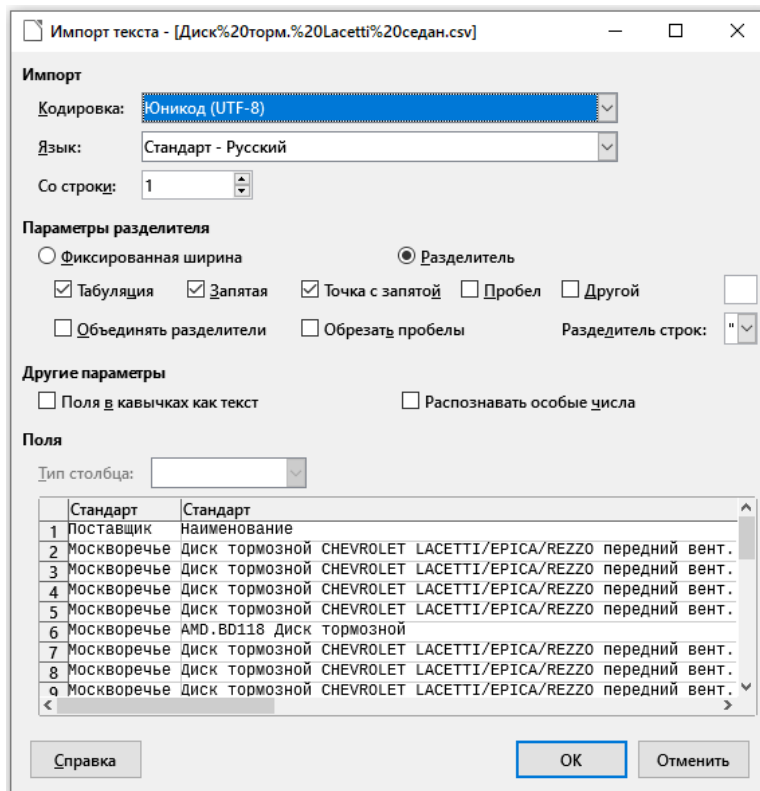


Рисунок 32 – Диалоговое окно программы Libre Office Calc

С помощью диалогового окна, изображенного на рисунке 32, можно выбрать вид разделителей, имеющих в данных таблицы. Также данное диалоговое окно позволяет выбрать кодировку данных, находящихся в таблице. Для корректной работы следует выбрать кодировку формата Юникод (UTF-8). Выбрав данный параметр один раз, он будет установлен по умолчанию в последующих открываемых данным методом файлах.

После выбора необходимых параметров, необходимо нажать на кнопку «ОК».

В результате открывается файл-таблица, наполненная собранными приложением данными. Часть данной таблицы изображена на рисунке 33.

1	Поставщик	Наименование	Бренд	На складе, шт.	Склад	Срок поставки, дн.	Цена	Ссылка
2	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	Miles	40	Казань	1	1173.0	http://portal.moskvorechie.ru/search.lmz#sK000459
3	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	Fenox	2	Казань	1	1173.0	http://portal.moskvorechie.ru/search.lmz#sTB219050
4	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	TRIALLI	2	Казань	1	1187.0	http://portal.moskvorechie.ru/search.lmz#sDF_062202
5	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	Nipparts	2	Казань	1	1192.65	http://portal.moskvorechie.ru/search.lmz#sJ3300913
6	Москворечье	AMD.BD118 Диск тормозной	AMD	4	Казань	1	1204.35	http://portal.moskvorechie.ru/search.lmz#sAMD_BD118
7	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	Japan parts	2	Казань	1	1204.79	http://portal.moskvorechie.ru/search.lmz#sDI-W09
8	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	Hi-Q	8	Казань	1	1261.21	http://portal.moskvorechie.ru/search.lmz#sSD3009
9	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	FIT	2	Казань	1	1344.0	http://portal.moskvorechie.ru/search.lmz#sFR0470V
10	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	NK	8	Казань	1	1348.81	http://portal.moskvorechie.ru/search.lmz#s205010
11	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	Hi-Q	16	Казань	1	1360.61	http://portal.moskvorechie.ru/search.lmz#sSD3017
12	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	Bosch	4	Казань	1	1380.0	http://portal.moskvorechie.ru/search.lmz#s0_986_479_R70
13	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	Girling	6	Казань	1	1384.2	http://portal.moskvorechie.ru/search.lmz#s6072181
14	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	LPR	2	Казань	1	1410.23	http://portal.moskvorechie.ru/search.lmz#sO100718
15	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	TRW	4	Казань	1	1538.0	http://portal.moskvorechie.ru/search.lmz#sDF7218
16	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	NK	2	Казань	1	1555.84	http://portal.moskvorechie.ru/search.lmz#s205009
17	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	NK	2	Казань	1	1605.47	http://portal.moskvorechie.ru/search.lmz#s315010
18	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	Bosch	2	Казань	1	1637.0	http://portal.moskvorechie.ru/search.lmz#s0_986_479_328
19	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	ACDelco	2	Казань	1	1866.0	http://portal.moskvorechie.ru/search.lmz#s19372233
20	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний вен.	Brembo	2	Казань	1	1993.0	http://portal.moskvorechie.ru/search.lmz#s09_9483_11

Рисунок 33 – Часть таблицы с собранными приложением данными

Взглянув на данную часть таблицы, можно увидеть результат работы приложения – собранные данные об автомобильных запчастях по введенному пользователем артикулу. Все данные имеют указание на источник (каталог поставщика) и отсортированы по колонкам «Срок поставки, дн.» и «Цена».

Всего данная таблица насчитывает 627 предложений из двух каталогов поставщиков. Самые выгодные предложения располагаются вверху таблицы. В данном случае, самыми выгодными являются предложения поставщика «Москворечье». Предложения другого поставщика – «АвтоЕвро» - по итогу

сортировки были помещены на 194 строку таблицы и до конца. Результаты полученных данных из каталога поставщика «АвтоЕвро» изображены на рисунке 34.

189	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний венг.	NiBK	2	Стр 70	4	2497.44	http://portal.moskvorechie.ru/search.lmz#sRN1324
190	Москворечье	Диск тормозной CHEVROLET LACETTI/EPICA/REZZO передний венг.	NiBK	8	Стр 70	4	2565.02	http://portal.moskvorechie.ru/search.lmz#sRN1360
191	Москворечье	Диск тормозной передний с покрытием daewoo nubira, chevrolet lacetti 1.4-1.6	Brembo	4	Стр 96	4	2598.77	http://portal.moskvorechie.ru/search.lmz#s09.9483.11
192	Москворечье	Диск тормозной передний daewoo nubira 1.4/1.6/1.8 03>	Ate	9	Стр 97	4	3099.49	http://portal.moskvorechie.ru/search.lmz#s24.0124-0199.1
193	Москворечье	Диски тормозные передние (перфорированные с проточкой) комплект 2 ш	AMD	1	Стр 70	4	3159.61	http://portal.moskvorechie.ru/search.lmz#sAMD.SPORT.BD5118
194	АвтоЕвро	Диск торм. CHEVROLET LACETTI 2005 =>	TRIALI	4	АвтоЕвро	5	1187.0	https://shop.autoeuro.ru/main/search?DF062202
195	АвтоЕвро	Диск пер. CHEVROLET EPICA, EVANDA, NUBIRA, LACETTI, REZZO 05=>	BLITZ	6	АвтоЕвро	5	1236.0	https://shop.autoeuro.ru/main/search?BS0245
196	АвтоЕвро	Тормозной диск	JAPANPARTS	14	партнер	5	1276.52	https://shop.autoeuro.ru/main/search?DIW09
197	АвтоЕвро	Диск торм. CHEVROLET (Lacetti J200)	SPEEDMATE	23	АвтоЕвро	5	1300.0	https://shop.autoeuro.ru/main/search?SM-BDU013
198	АвтоЕвро	Диск тормозной передний вентилируемый	BAPCO	2	партнер	5	1306.74	https://shop.autoeuro.ru/main/search?BD0156
199	АвтоЕвро	Диск тормозной передний вентилируемый (Лш.)	BM	4	партнер	5	1324.14	https://shop.autoeuro.ru/main/search?BDV110
200	АвтоЕвро	Диск тормозной передний / Chevrolet Lacetti 03~	UX CLENT	28	партнер	5	1351.03	https://shop.autoeuro.ru/main/search?201-615-SD11
201	АвтоЕвро	Диск тормозной FR DF7218 (256x24) 4 отв	JD	50	партнер	5	1377.18	https://shop.autoeuro.ru/main/search?JBD0012
202	АвтоЕвро	Диск торм. CHEVROLET LACETTI 04=>	BOSCH	8	АвтоЕвро	5	1380.0	https://shop.autoeuro.ru/main/search?0.986.479.R70
203	АвтоЕвро	Диск торм. CHEVROLET LACETTI 04=>	BOSCH	2	АвтоЕвро	5	1380.0	https://shop.autoeuro.ru/main/search?0.986.479.R70

Рисунок 34 – Часть таблицы с результатами собранных данных из каталога поставщика «АвтоЕвро»

По итогу проведенного тестирования можно сказать, что приложение работает исправно и без ошибок. Оно корректно собирает и обрабатывает данные, производит сортировку по установленным данным. Тот факт, что предложения поставщика «АвтоЕвро» располагаются с 194 строки, говорит о том, что их предложение по данному конкретному запросу является не самым выгодным для выбора.

Исходя из полученной в результате тестирования информации, можно прийти к выводу о том, что приложение работает исправно и отлично справляется со своей задачей.

Выводы по главе 3

В текущей главе данной ВКР было проведено тестирование разработанного приложения на примере выполнения реальной задачи.

В результате проведенного тестирования разработанного приложения можно сделать вывод о том, что приложение работает исправно, быстро и без ошибок.

Заключение

Данная выпускная квалификационная работа посвящена теме разработки приложения по сбору данных для автоматизации одного из процессов магазина автомобильных запчастей.

В результате выполнения данной ВКР были выполнены следующие поставленные задачи:

- изучен и проанализирован процесс сбора данных на примере организации, именуемой ООО «Статус». Анализ показал, что для вышеуказанного процесса требуется автоматизация, достигаемая путем разработки приложения по сбору данных. Для разрабатываемого приложения были прописаны основные требования;
- выполнено проектирование разрабатываемого приложения средствами языка UML. Разработаны диаграмма компонентов, диаграмма последовательности и диаграмма деятельности;
- выбрана технология реализации разрабатываемого приложения – язык программирования Python;
- разработано приложение по сбору данных;
- проведено тестирование разработанного приложения по сбору данных на примере выполнения реальной задачи по сбору данных в магазине автомобильных запчастей.

Разработанное приложение потенциально может быть использовано сторонними организациями, имеющими схожую тематику ведения бизнеса – торговлю автомобильными запчастями.

Разработанное приложение имеет возможность масштабирования и улучшения путем добавления дополнительного функционала и подключения большего количества сайтов-каталогов поставщиков.

Список используемой литературы и используемых источников

Научная и методическая литература

1. Бхаргава, А. Грокаем алгоритмы. Иллюстрированное пособие для программистов и любопытствующих [Текст] / А. Бхаргава ; [пер. с англ. Е. Матвеев]. – Санкт-Петербург : Издательство «Питер», 2017. – 288 с.
2. Бэрри, П. Изучаем программирование на Python [Текст] / П. Бэрри ; [пер. с англ. М. Райтман]. – Москва : Издательство «Э», 2017. – 624 с.
3. Васильев, А. Н. Python на примерах. Практический курс по программированию [Текст] / А. Н. Васильев. – Санкт-Петербург : Издательство «Наука и Техника», 2019. – 432 с.
4. Лутц, М. Изучаем Python, том 1, 5-е изд. [Текст] / М. Лутц ; [пер. с англ. Ю. Артеменко]. – Санкт-Петербург : Издательство «Диалектика», 2019. – 832 с.
5. Лутц, М. Изучаем Python, том 2, 5-е изд. [Текст] / М. Лутц ; [пер. с англ. Ю. Артеменко]. – Санкт-Петербург : Издательство «Диалектика», 2020. – 720 с.

Электронные ресурсы

6. Анисимов, В.В. Лекция 8.8 «Методология BPMN» [Электронный ресурс] / В.В. Анисимов // Собрание учебно-методических и научных материалов, 2020. URL: https://sites.google.com/site/anisimovkhv/learning/pris/lecture/tema8/tema8_4 (дата обращения: 05.03.2021)
7. Борознов, О. Руководство по моделированию бизнес-процессов и алгоритмом в нотации BPMN 2.0 [Электронный ресурс] / О. Борознов // руководство, 2018. URL: https://www.optimacons.info/kb/course.php?LESSON_ID=63 (дата обращения: 16.03.2021)
8. Вставская, Е. Программирование. UML-диаграммы классов: сущности, связи, интерфейсы [Электронный ресурс] / Е. Вставская // статья для

форума по программированию, 2020. URL: <https://prog-cpp.ru/uml-classes> (дата обращения: 12.04.2021)

9. Дашкевич, Н. В. Моделирование экономических и бизнес-процессов: [Электронный ресурс] / Н. В. Дашкевич, Т. А. Петровская // методическое пособие по выполнению лабораторных работ для студентов специальности «Управление инновационными проектами промышленных предприятий», 2018. URL:

https://rep.bntu.by/bitstream/handle/data/48124/Modelirovanie_ekonomicheskikh_i_biznes-processov_removed.pdf?sequence=1&isAllowed=y (дата обращения: 22.03.2021)

10. Матюшкин, Л. Python и API: превосходное комбо для автоматизации работы с публичными данными [Электронный ресурс] / Л. Матюшкин // статья, 2021. URL: <https://proglib.io/p/python-i-api-prevoshodnoe-kombo-dlya-avtomatizacii-raboty-s-publichnyimi-dannymi-2021-02-26> (дата обращения: 05.04.2021)

11. Простое руководство по UML-диаграммам и моделированию баз данных [Электронный ресурс] / Microsoft 365 Team // статья, 2019. URL: <https://www.microsoft.com/ru-ru/microsoft-365/business-insights-ideas/resources/guide-to-uml-diagramming-and-database-modeling> (дата обращения: 11.04.2021)

12. Теория и практика UML. Диаграмма деятельности [Электронный ресурс] / IT-GOST.RU // статья для электронной библиотеки стандартов оформления проектной документации, 2019. URL: http://www.it-gost.ru/articles/view_articles/96 (дата обращения: 10.04.2021)

13. Формат данных JSON в Python [Электронный ресурс] / © Pythonist, 2020. URL: <https://pythonist.ru/format-dannyh-json-v-python> (дата обращения: 06.04.2021)

14. Шамаев, И. Python API Tutorial: Примеры GET POST AUTH requests (HTTP запросов) [Электронный ресурс] / И. Шамаев // Статья, 2020. URL:

<https://python.ivan-shamaev.ru/python-api-tutorial-get-post-auth-http-requests-examples> (дата обращения: 05.04.2021)

Литература на иностранном языке

15. Learn Python 3 the Hard Way: A Very Simple Introduction to the Terrifyingly Beautiful World of Computers and Code, First Edition / Zed. A. Shaw, 2017. – 320 pages.

16. Algorithmic Problem Solving with Python [Электронный ресурс] / J. B. Shneider, S. L. Broschat, J. Dahmen, 2017. URL: <https://t.me/pythonbookarchive/695> (дата обращения: 11.04.2021)

17. Python 3.9.5 documentation [Электронный ресурс] / Software Foundation, last update on May 18, 2021. URL: <https://docs.python.org/3.9> (дата обращения: 07.04.2021)

18. Python Web Scraping [Электронный ресурс] / Katharina Jarmul, Richard Lawson, 2017. URL: <https://t.me/pythonbookarchive/18> (дата обращения: 07.04.2021)

19. Requests: HTTP for Humans [Электронный ресурс] / © MMXVIX. A Kenneth Reitz Project, 2011-2021. URL: <https://docs.python-requests.org/en/master> (дата обращения: 21.04.2021)

20. UML Diagram Types Guide: Learn About All Types of UML Diagrams with Examples [Электронный ресурс] / © Cinergix Pty. Ltd, 2008-2021. URL: <https://creately.com/blog/diagrams/uml-diagram-types-examples> (дата обращения: 10.04.2021)

Приложение А

Фрагмент файла «Main.py»

```
9 # Функция инициализирует сбор данных и создание файла
10 # Ничего не возвращает
11 def product_search():
12     global full_path_to_file # Глобальная переменная - абсолютный путь к создаваемому файлу
13     article = Label_Article_var.get() # Получение артикула
14     # Проверка на пустое поле артикула
15     if article != '':
16         Button_Search.config(text='Идёт поиск...', bg='#f0f0f0') # Изменение кнопки "Поиск"
17         Button_ToFile.config(state=tk.DISABLED, bg='#f0f0f0') # Изменение кнопки "Открыть файл"
18         Entry_Article.config(bg='ffffff') # Отбеливание поля ввода ссылки
19         Label_Status.config(text=f'Выполняется поиск по запросу {article}...', fg='black') # Изменение статуса
20         msk = logic.get_mskrechie_list(article) # Процесс сбора данных с сайта "Москворечье"
21         sleep(2) # Создание задержки
22         aeuro = logic.get_autoeuro_list(article) # Процесс сбора данных с сайта "АвтоЕвро"
23         # Проверка на неверный ввод артикула
24         if len(msk) == 0 and len(aeuro) == 0:
25             Button_Search.config(text='Поиск', bg='#f0f0f0') # Изменение кнопки "Поиск"
26             Button_ToFile.config(state=tk.DISABLED, bg='#f0f0f0') # Изменение кнопки "Открыть файл"
27             Entry_Article.config(bg='#ffd6d6') # Окрашивает поле ввода ссылки в красный цвет
28             Label_Status.config(text='Вы ввели некорректный запрос!', fg='red') # Окрашивает статус в красный цвет
29         else:
30             total_list = msk + aeuro # Соединение списков с данными в один список
31             total_list = sorted(total_list, key=lambda x: (x['delivery'], x['price'])) # Сортировка итогового списка
32             # Проверка на ввод пользовательского названия файла
33             if Label_FileName_var.get() == '':
34                 full_path_to_file = logic.create_file(total_list, article) # Создание файла с названием по умолч.
35             else:
36                 full_path_to_file = logic.create_file(total_list, Label_FileName_var.get()) # Пользов. название
37             # Изменение стилей кнопок и статуса при удачном поиске
38             Label_Status.config(text='Готово!', fg='#0abe38')
39             Button_Search.config(text='Искать ещё')
40             Button_ToFile.config(state=tk.NORMAL)
41     else:
42         Button_Search.config(text='Поиск', bg='#f0f0f0') # Изменение кнопки "Поиск"
43         Button_ToFile.config(state=tk.DISABLED, bg='#f0f0f0') # Изменение кнопки "Открыть файл"
44         Entry_Article.config(bg='#ffd6d6') # Окрашивает поле ввода ссылки в красный цвет
45         Label_Status.config(text='Поле ввода артикула не должно быть пустым!', fg='red')
```

Рисунок А.1 - Программный код функции product_search()

Приложение Б Фрагмент файла «Logic.py»

```
12 # Функция собирает данные с сайта "Москворечье"
13 # Возвращает список с данными о позициях
14 def get_mskrechie_list(article):
15     mskrechie = settings['Moskvorechie'] # Получение данных о сайте из файла settings.json
16     response = requests.get(mskrechie['site'],
17                             params={
18                                 'l': mskrechie['username'],
19                                 'p': mskrechie['API_KEY'],
20                                 'act': "price_by_nr_firm",
21                                 'nr': str(article),
22                                 'cs': 'utf8',
23                                 'oe': 'oe',
24                                 'avail': 'avail',
25                                 'v': 1
26                             }) # Отправка запроса к сайту по API
27     search_list = json.loads(response.text) # Перевод запроса из json в py-object
28     result_list = [] # Создание списка для товаров
29     # Цикл прохождения по каждому товару и сбору данных по ключам
30     for product in search_list['result']:
31         # Фильтр по кол-ву товаров на складе
32         if int(product['ddays']) > 0:
33             result_list.append({
34                 'provider': 'Москворечье',
35                 'name': product['name'],
36                 'brand': product['brand'],
37                 'stock': product['stock'],
38                 'storage': product['sname'],
39                 'delivery': int(product['ddays']),
40                 'price': float(product['price']),
41                 'href': 'http://portal.moskvorechie.ru/search.lmz#s' + product['nr']
42             }) # Все данные записываются в конец списка
43     return result_list
```

Рисунок Б.1 – Программный код функции по сбору данных с каталога поставщика «Москворечье»