

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»**
(Н И У « Б е л Г У »)

ИНСТИТУТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ И ЕСТЕСТВЕННЫХ НАУК

КАФЕДРА ИНФОРМАЦИОННЫХ СИСТЕМ И РОБОТОТЕХНИЧЕСКИХ СИСТЕМ

АВТОМАТИЗАЦИЯ ПРОЦЕССА КРАУЛИНГА

Выпускная квалификационная работа студента
очной формы обучения
направления подготовки 09.03.02 Информационные системы и технологии
4 курса группы 07001407
Пичугина Никиты Игоревича

Научный руководитель
к.т.н., доцент Щербинина Н.В.

БЕЛГОРОД 2018

РЕФЕРАТ

Автоматизация процесса краулинга. –
Пичугин Никита Игоревич, выпускная квалификационная работа бакалавра. –
Белгород, Белгородский государственный национальный исследовательский университет (НИУ «БелГУ»), количество страниц 58, включая приложение 76, количество рисунков 22, количество таблиц 7, количество использованных источников 30.

КЛЮЧЕВЫЕ СЛОВА: информационная подсистема, процесс краулера, техническая оптимизация сайта.

ОБЪЕКТ ИССЛЕДОВАНИЯ: процессы технической оптимизации сайтов.

ПРЕДМЕТ ИССЛЕДОВАНИЯ: средства краулинга сайта.

ЦЕЛЬ РАБОТЫ: совершенствование процессов автоматизации краулинга сайтов.

ЗАДАЧИ ИССЛЕДОВАНИЯ: анализ методов и средств извлечения информации со сторонних web-сайтов; выбор метода и средств извлечения информации; программная реализация метода извлечения информации; сравнение и тестирование разработанного приложения.

МЕТОДЫ ИССЛЕДОВАНИЯ: методы моделирования бизнес-процессов, методы проектирования информационных систем, оценка окупаемости подсистемы, методы технической оптимизации сайтов.

ПОЛУЧЕННЫЕ РЕЗУЛЬТАТЫ: в результате работы была спроектирована и реализована информационная подсистема автоматизации процесса краулинга для деятельности организации ООО «Продвижение».

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
1 Аналитическая часть	7
1.1 Технико- экономическая характеристика предметной области предприятия.....	7
1.2 Экономическая сущность задачи.....	10
1.3 Обоснование необходимости и цели использования вычислительной т ехники для решения задачи.....	10
1.4 Постановка задачи	17
1.5 Анализ существующих разработок и обоснование выбора технологии проектирования	20
2 Обоснование проектных решений	25
2.1 Обоснование проектных решений по техническому обеспечению.....	25
2.2 Обоснование проектных решений по информационному обеспечению	26
2.3 Обоснование проектных решений по программному обеспечению....	27
2.4 Обоснование проектных решений по технологическому обеспечению	29
2.5 Обоснование выбора программных средств.....	31
3 Проектная часть	33
3.1 Информационное обеспечение задачи информационной подсистемы ав томатизации процесса краулинга.....	33
3.2 Программное обеспечение задачи информационной подсистемы.....	38
3.3 Технологическое обеспечение задачи	41
3.4 Описание контрольного примера реализации подсистемы.....	42
3.5 Расчет экономической эффективности.....	51
ЗАКЛЮЧЕНИЕ	60
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	62
ПРИЛОЖЕНИЕ А	65
ПРИЛОЖЕНИЯ Б.....	68

ВВЕДЕНИЕ

Современный сайт на просторах сети Интернет – это, прежде всего, источник уникальной и полезной информации. Люди в сети ищут ответы на свои повседневные вопросы. Если ваш ресурс в состоянии дать исчерпывающие ответы, то его популярность не заставит себя долго ждать.

Однако в основные задачи контента на сайте входит не только донесение информации до посетителя и повышение его общей грамотности и уровня образованности, но и для привлечения клиентов или с целью извлечения прибыли. Это условие должно обязательно учитываться в процессе подготовки контента для сайта.

Источники получения контента подразделяются на заимствованные и оригинальные. Но, несмотря на то, что сторонний используемый материал имеет много недостатков, с точки зрения поискового продвижения, большинство web-сайтов используют данный подход к своему содержанию.

Самые грубые ошибки, которые происходят при создании web-сайтов –

это технические. Такие ошибки могут быть выявлены только определёнными методами или краулерами сайтов. Не все ИТ специалисты имеют опыт в выявлении такого вида ошибок, поэтому для упрощения выявления их разработали краулер способный выявлять и перебирать страницы сайтов. Однако не один из разработанных на данный момент краулеров не может производить анализ сайта и выводить отчет, а также они являются платными. Поэтому на данный момент актуальность разработки данной подсистемы является очень актуальным.

Целью выпускной квалификационной работы является оптимизация web-сайтов за счет автоматизации процесса краулинга.

Объектом исследования является техническая оптимизация сайта. В качестве предмета исследования рассматриваются средства краулеров сайтов.

Для достижения поставленной цели были определены следующие задачи:

- анализ методов и средств извлечения информации со сторонних web-сайтов;
- выбор метода и средств извлечения информации;
- программная реализация метода извлечения информации;
- сравнение и тестирование разработанного приложения.

Выпускная квалификационная работа состоит из трех частей, реализующие поставленные задачи.

В первой главе описаны теоретические обоснования усовершенствования деятельности специалиста по поисковому продвижению, подробно описана характеристика деятельности и организационная структура управления ООО «Продвижения».

Вторая глава содержит обоснование проектных решений по техническому, информационному, программному и технологическому обеспечению задачи по автоматизации процесса краулинга.

В третьей главе описан процесс проектирования и разработки информационной подсистемы автоматизации процесса краулинга для деятельности организации, представлены результаты работы информационной подсистемы и рассчитана экономическая эффективность.

В результате выполнения выпускной квалификационной работы разработана информационная подсистема автоматизации процесса краулинга для деятельности организации ООО «Продвижение».

Данная работа состоит из страницы, 42 рисунка, 7 таблиц, 33 литературных источников и 2 приложений.

1 Аналитическая часть

1.1 Техно-экономическая характеристика предметной области предприятия

1.1.1 Характеристика предприятия

Компания «Продвижение» занимает первое место среди компаний по разработке, продвижение сайтов и настройки контекстной рекламы в Белгороде.

Организационно-правовая форма предприятия – общество с ограниченной ответственностью.

Во главе предприятия стоит генеральный директор, руководящий всеми отделами и финансово-экономической деятельностью. Управление предприятием осуществляется в соответствии с законодательством РФ и уставом предприятия. Численность сотрудников ООО «Продвижения» составляет 15 человек.

К основному виду текущей деятельности организации настройка контекстной рекламы в Яндекс Директе и в Google Adwords, разработка продающих и оптимизированных сайтов для максимальной эффективности интернет-маркетинга. К ряду основных деятельности большую часть занимает SEO продвижение.

ООО «Продвижение» располагается в городе Белгород.

1.1.2 Характеристика организационной структуры управления ООО «Продвижение»

Под организационной структурой управления предприятием понимается состав (перечень) отделов, служб, подразделений в аппарате управления предприятием, характер соподчиненности, взаимодействия, координационные и информационные связи, порядок распределения функций управления по различным уровням и подразделениям [2].

ООО «Продвижение» имеет линейно-функциональную организационную структуру управления предприятием.

Линейно-функциональная структура управления характеризуется тем, что во главе каждого структурного подразделения стоит руководитель, осуществляющий все функции управления и руководство подчиненными работниками, а им оказывают помощь функциональные органы.

Структура управления организацией ООО «Продвижения» представлена на рисунке 1.1.



Рисунок 1.1 – Структура управления ООО «Продвижение»

Суть линейно-функциональной организационной структуры управления состоит в том, что выполнение отдельных функций по конкретным вопросам возлагается на специалистов.

Элементами структуры являются отдельные работники и службы аппарата управления, отношения между ними поддерживаются благодаря связям, которые носят линейный и функциональный характер.

Линейные звенья принимают решения, а функциональные подразделения информируют и осуществляют вопросы и подготовку соответствующих решений, освобождают линейных руководителей от планирования, финансовых расчетов, программ, необходимых планов для принятия решений.

Линейно-функциональная структура управления имеет следующий ряд преимуществ:

- быстрое осуществление действий по распоряжениям и указаниям, отдающимся вышестоящими руководителями нижестоящим;
- рациональное сочетание линейных и функциональных взаимосвязей, стабильность полномочий и ответственности за персоналом,
- единство и четкость распорядительства;
- более высокая оперативность принятия и выполнения решений;
- личная ответственность каждого руководителя за результаты деятельности;
- профессиональное решение задач специалистами функциональных служб.

Основные проблемы краулеров является скорость парсинга и хранения просканированной информации. В большинстве сайтов стоит блокировка на парсинг страниц, поскольку это создает большую нагрузку на сервер и на сам сайт. Нормой парсинга страниц – это 5-6 ссылок в секунду, но на некоторых сайтах возможно еще меньше, поскольку есть вероятность блокированием системой управления сайтом. С хранением просканированных данных стоит сложнее цель, на данный момент

все существующие краулеры хранят все данные в ОЗУ, даже при сохранении данных на диск они все равно при запуске помещаются в оперативную память, что затрачивает большие ее объемы, в среднем на 120000 ссылок необходимо 4гб ОЗУ.

1.2 Экономическая сущность задачи

Целью выпускной квалификационной работы является автоматизация процесса краулинга в компании ООО «Продвижение».

Разрабатываемая подсистема позволит упростить процессы технической оптимизации сайтов, путем сканирования и представление в виде отчета о результатах.

В результате выпускной квалификационной работы ожидается автоматизация процесса краулинга для анализа контента сайта в виде подсистемы, которая позволит устранить недостатки традиционного анализа контента сайта.

Для решения поставленной цели можно выделить следующие задачи:

- анализ деятельности seo-специалиста по технической оптимизации;
- изучение процессов и технологий существующих программных средств автоматизации процесса краулинга с целью выделения основных функциональных возможностей системы;
- усовершенствование процесса автоматизации процесса деятельности организации с использованием информационной подсистемы.

1.3 Обоснование необходимости и цели использования вычислительной техники для решения задачи

Для проектирования информационной подсистемы автоматизации процесса краулинга для деятельности ООО «Продвижение» изучены выполня

емые функции seo-специалиста и рассмотрен процесс технической оптимизации сайтов.

Основной задачей seo-специалиста является анализ оптимизирующего сайта на различные технические ошибки, после его разработки.

Контекстная диаграмма seo-оптимизации сайта 1.2

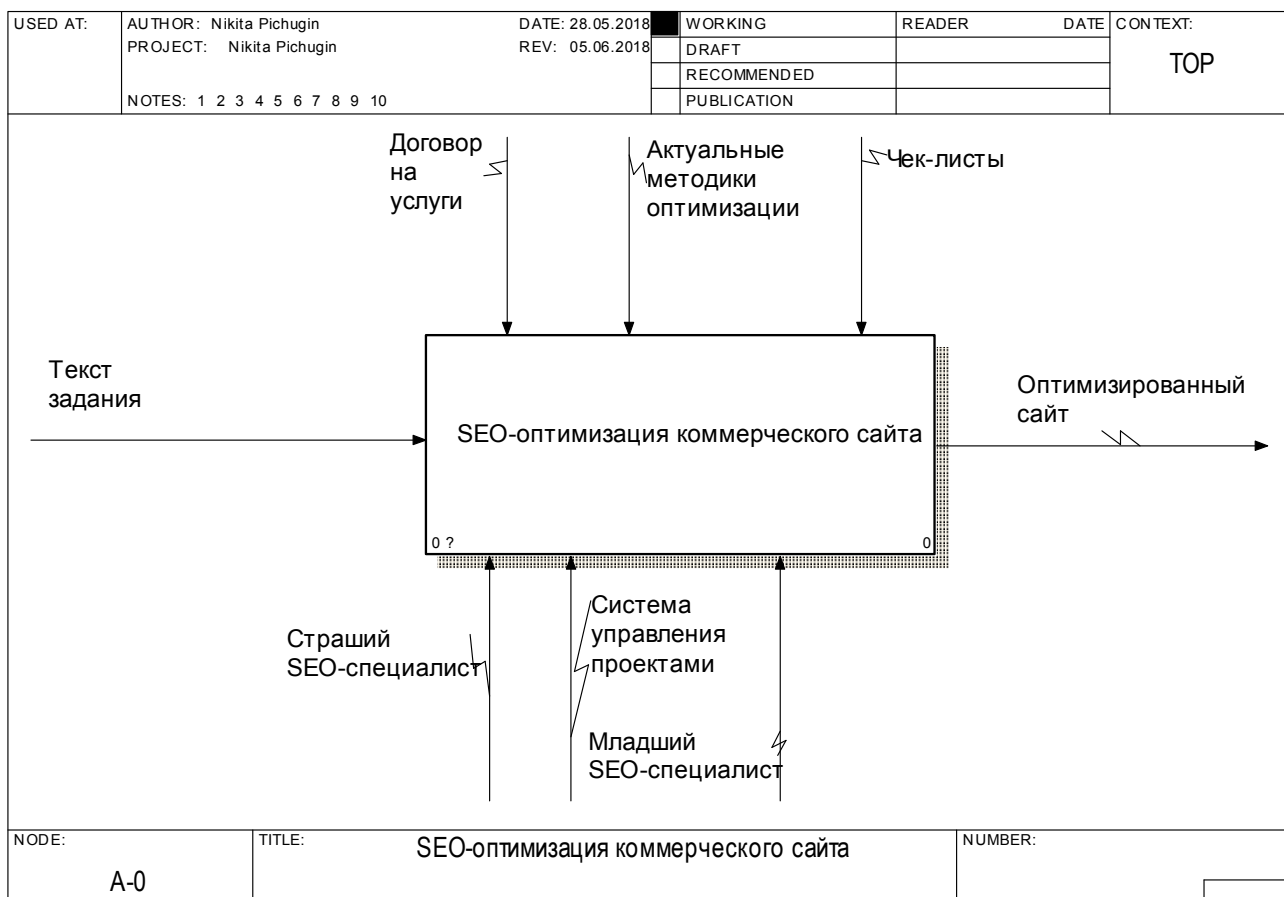


Рисунок 1.2 – Контекстная диаграмма seo-оптимизации коммерческого сайта

Все проекты по поисковому продвижению сайта регулируются договором на услуги продвижения, составленным ранее аккаунт-менеджером, актуальными методиками, и составленными чек-листами.

Управление распределением информационных потоков осуществляется сотрудниками организации и вспомогательными системами управления проектами. Старший seo-специалист отвечает за организацию, координацию, руководство, контроль и реализацию работ по поисковому продвижению. Младший seo-специалист подчиняется и выполняет задания и поручения старшего seo-специалиста организации, а также взаимодействует с базовыми или техническими задачами.[1]

Входными потоками фирмы являются данные, необходимые для организации для правильной постановки задачи и выполнения. К таким данным относятся текст задания, в котором согласованы разделы или запросы, которые необходимо продвинуть по поисковым системам, информационно-справочная информация об организациях.

Декомпозиция процесса seo- оптимизации коммерческого сайта представлена на рисунке 1.3.

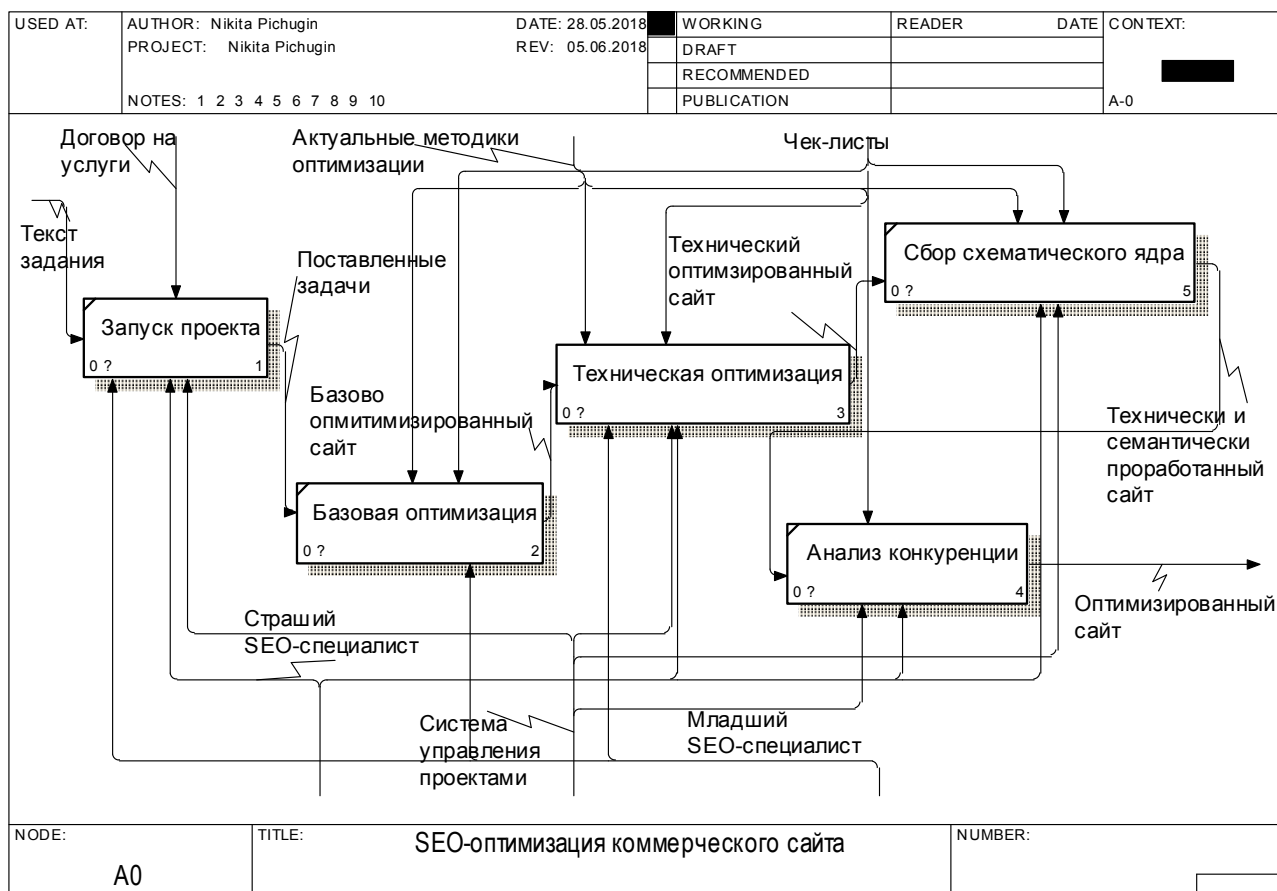


Рисунок 1.3 – Декомпозиция контекстной диаграммы

В процессе продвижения сайта в поисковых системах разделяется на 4 сущности. Запуск проекта проставить базовые чек-листы для проверки сайта на способность ранжирования в поисковых системах. Техническая оптимизация выявляет ошибки допущены разработчикам и в процессе разработки сайта. Сбор семантического ядра позволяет проанализировать весь охват аудитории и спрос на определённую услугу или товар. Анализ конкуренции позволяет выявить упущенные и недостающие решения для повышения конверсии и привлечения клиентов на сайт.[1]

Декомпозиция процесса «Техническая оптимизация» представлена на рисунке 1.4.

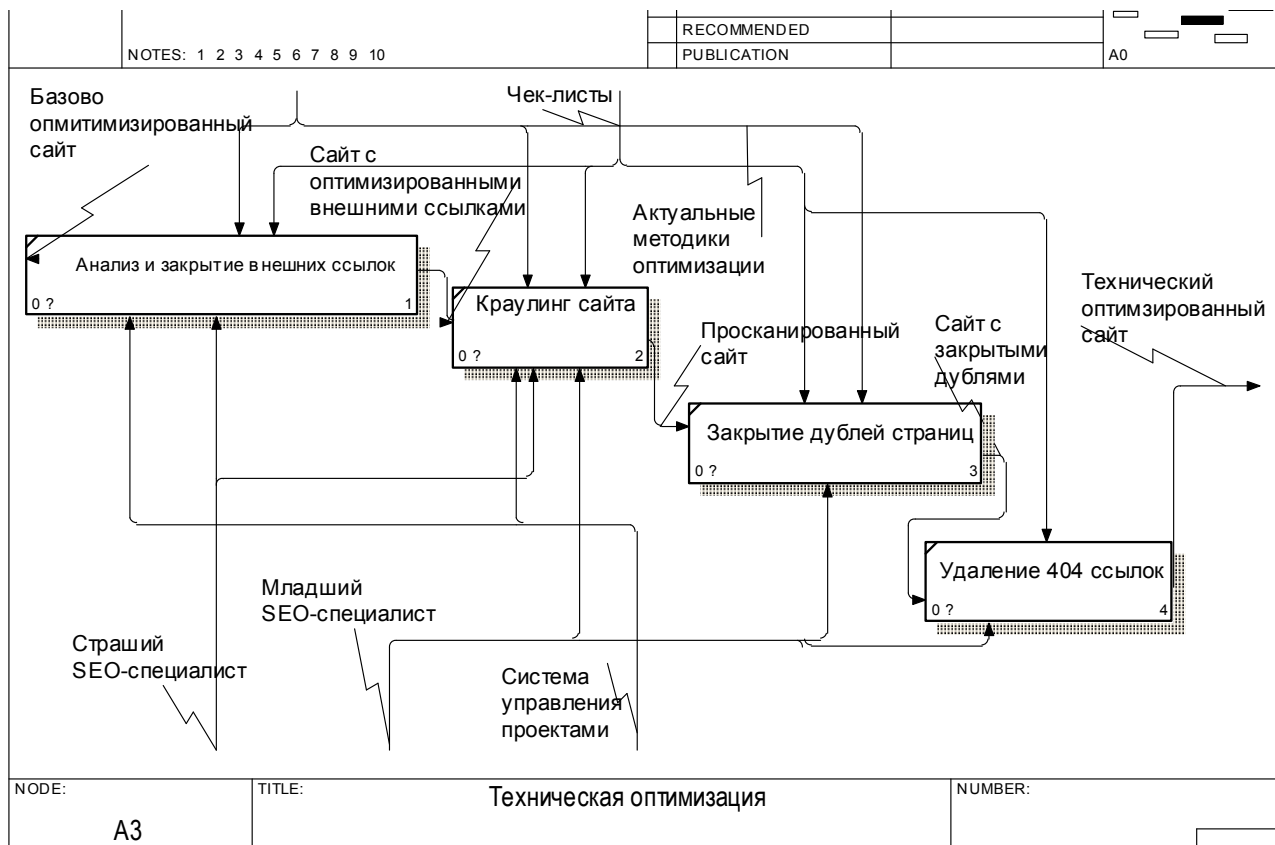


Рисунок 1.4 – Декомпозиция процесса «Техническая оптимизация»

Первым шагом технической оптимизации сайта является анализ внешних ссылок на сайте, в результате которого осуществляется проверка и закрытия ссылок. После первичной обработки выполняется краулинг сайта с целью последующего распределения задач на исправление технических ошибок, допущенных при разработке. Одним из базовых технических ошибок является наличие дублей страниц сайта, которые закрываются от поисковых систем. Последним этапом является удаление 404 ошибок на сайте. [3]

Организация и технология процесса краулинга в digital-agency ООО «Продвижение» осуществляется seo-

оптимизатором вручную или платными программными приложениями и обладает следующими характеристиками:

- написание отчетов о сделанном аудите сайта;
- анализ и выявления технических ошибок,
- отсутствуют средства представления информации в ввиду графиков ;
- отсутствуют бесплатные программные средства для быстрой проверки и аудита сайта;

Совершенно очевидно, что традиционная схема проверки технических ошибок сайта не способна выявить быстрое решение. В деятельности различных организаций при ведении экспресс-аудита сайта или выявление технических ошибок, как правило, через некоторое время возникают следующие проблемы:

- бесплатное использование дополнительного программного обеспечения для автоматизации процесса краулинга;
- невозможность обеспечить быструю передачу входящих документов и информации;
- большие затраты времени на подготовку и согласование отчетов;

Перечисленные проблемы и недостатки организации и технологии работы специалиста по поисковому продвижению позволяют сделать вывод о необходимости разработки информационной подсистемы с целью обеспечения повышения оперативности и качества работы с отчетами о аудитах и предоставление бесплатного использования программного обеспечения для автоматизирования процесса краулинга.

1.4 Постановка задачи

1.4.1 Цель и назначение автоматизированного варианта решения задачи

К автоматизированным информационным подсистемам относится упорядоченная совокупность экономико-математических методов и моделей, технических и программных средств, организованных на базе новой информационной технологии в решении технических задач seo-специалистов.

Цель создания информационной подсистемы процесса краулинга контента сайта является обеспечение повышения качества страниц, при оптимизации, а также нахождение всех технических ошибок, сокращение сроков прохождения аналитики, формирование отчетности для аккаунт-менеджеров.

Информационная подсистема процесса краулинга направлена на упорядочение работы со страницами сайта и обеспечение следующих функциональных возможностей:

- сканирование всех ссылок сайта;
- проверка ответа сервера;
- сканирование мета-тегов сайта;
- проверка на дубли страниц;
- анализ контента сайта;
- получение отчетности результатов анализа сайта;
- результат выводит в виде диаграмм;
- реализация поиска страниц и мета-тегов просканированных страниц сайта.

Таким образом, для повышения уровня эффективности технической оптимизации ООО «Продвижения» необходимо создать профессионально сп

роектированную и выполненную подсистему сканирования сайта на проверку технические ошибки сайтов.

1.4.2 Общая характеристика организации решения задачи на ЭВМ

Автоматизация процессов краулинга основывается на технологии «клиент-сервер». Принцип технологии «клиент-сервер» состоит в распределении функций стандартного интерактивного приложения на группы:

- презентационная логика – функции ввода и отображения данных;
- бизнес-логика –
 прикладные функции, определяющие основные алгоритмы решения задач приложения;
- процессор управления данными –
 функции хранения и управления информационными ресурсами;
- логика обработки данных –
 функции обработки данных внутри приложения;
- служебные функции, играющие роль связок между функциями первых четырех.

Хранилище данных и файлы сервера находятся на стационарном выделенном компьютере, выполняющим функцию сервера баз данных, который управляет данными и выполняет запросы клиентских приложений. Клиентские приложения предоставляют интерфейс. Приложения должны быть установлены на персональные компьютеры с доступом к сети интернет.

Методика разработки подсистемы включает в себя выполнение следующих этапов:

- изучение деятельности seo-специалиста ООО «Продвижения»;

- выявление необходимых функций подсистемы для автоматизации и организации процесса краулинга и выявление на технические ошибки;
- проектирование и анализ функциональных диаграмм;
- проектирование подсистемы анализа страницы сайта;
- разработка информационной подсистемы.

Для результативного решения поставленной задачи необходимо соответствующее техническое обеспечение. Под техническим обеспечением понимают состав, формы и способы различных технических устройств, необходимых для выполнения информационных процедур: сбора, регистрации, передачи, хранения, обработки и использования информации [5].

Техническое обеспечение данного проекта имеет в своем составе ЭВМ, монитор, манипулятор типа мышь, клавиатуру.

При выборе ЭВМ следует руководствоваться рядом характеристик. К основным характеристикам вычислительной техники относят эксплуатационно-технические характеристики: объем памяти, надежность, стоимость, быстродействие, точность вычислений. Существенное значение имеют и другие характеристики, такие как: программная совместимость, универсальность, габариты, вес и энергопотребление. Именно от этих значений перечисленных параметров ЭВМ зависит возможность работы с необходимым программным обеспечением.

Для решения задач обработки данных наиболее подходят ПЭВМ, которые характеризуются относительно невысокой стоимостью, легко конфигурируемыми, обладают простой системой обучения, хорошей производительностью и подходящими характеристиками надежности, и объемом памяти.

Чтобы реализовать комплекс поставленных задач, при выборе ПЭВМ необходимо учитывать тактовую частота процессора, объем и скорость ра

боты жесткого диска, объем оперативной памяти, наличие периферийных устройств и другие технические характеристики.

Значительным требованием к информационному обеспечению служит уровень хранения данных в подсистеме, построенный на основе современной реляционной СУБД. Для обеспечения целостности данных должны использоваться встроенные механизмы СУБД. Средства операционной системы и СУБД должны обеспечивать документирование и протоколирование обрабатываемой в системе информации.

Для решения поставленной задачи подходит диалоговый режим взаимодействия пользователя с вычислительной системой, который обеспечивает возможность оперативного вмешательства пользователя в процесс обработки данных. Диалоговый режим характеризуется отсутствием жестко закрепленной последовательности операций обработки данных.

Важными критериями при разработке подсистемы автоматизации процесса краулинга является структура и функциональность разрабатываемой системы, а также предоставление хранимой информации базы данных в объективной форме.

1.5 Анализ существующих разработок и обоснование выбора технологии проектирования

В последние годы алгоритмы поисковых систем стали намного эффективнее различать и находить ошибки, допущенные в результате разработки или оптимизирования сайта. Существует несколько программных разработок систем автоматизирования процесса краулинга и анализа контента сайта на технические ошибки сайта, которые похожи между собой. Для выявления функциональных решений подсистемы необходимо проанализировать существующие программные средства.

Для анализа и оценки программных средств выделили критерии:

- а) функциональные возможности (от 1 до 5 баллов);
- б) открытость (1 балл – нет, 5 баллов – есть);
- в) свободная лицензия (1 балл – нет, 5 баллов – есть);
- г) простота внедрения и использования (от 1 до 5 баллов);
- д) масштабируемость (от 1 до 5 баллов).

Система автоматизирования процесса краулинга – программный комплекс, который обеспечивает процесс сканирование сайта, выявление технических ошибок сайта, а также контроль просканированных документов сайта.

Наиболее распространенными для малых предприятий являются Screaming Frog Seo Spider, Comparser и Xenus Link Sleuth.

Первой отечественной системой автоматизации документооборота является система «Screaming Frog Seo Spider», разработанная командой разработчиков. Система ориентирована на полное автоматизирование процесса краулинга, анализа всего контента сайта на технические ошибки, поддержку других известных сервисов для аналитики и выявления проблем сайта.

Система «Comparser» - также является автоматизирующей системой краулинга, с выявлением различных технических ошибок, однако имеет ряд других функций, позволяющих подойти к решению технических ошибок другими методами. Создал данное приложение один российский разработчик Александр Алаев.

Xenus Link Sleuth – программная обеспечение являющееся самым ранним продуктом из всех перечисленных ранее. Изначально программа являлась приложением проверяющая веб-сайты на битые ссылки. Все что являлось ссылкой на сайте сканировалось и в дальнейшем могло быть отсортировано и выгружено в формат CSV. В дальнейшем компания Lick Sleuth выпустили обновление позволяющая с

мотреть не только ссылки, но искать страницы с большой отдачей времени, создавать карту сайта и т.д.

Сравнительная характеристика программных средств представлена в таблице 1.1.

Таблица 1.1 – Сравнительная характеристика программных средств

Критерий оценки	Название СЭД		
	Screaming Frog Spider	Seo Compar ser	Xenus Link Sl euth
Функциональные возможности	5	4	3
Открытость	4	2	4
Свободная лицензия	4	3	3
Простота внедрения и использования	4	4	2
Масштабируемость	4	3	1
Сумма	21	16	13

Процесс краулинга предполагает учет больших объемов информации, хранение и обработку входящей, внутренней и исходящих метаданных, проверку на технические ошибки, что обуславливает широкое применение программ. Все описанные выше системы автоматизации процесса краулинга являются очень крупными и слишком универсальными.

Отличиями разрабатываемой подсистемы от других систем автоматизации процесса краулинга являются функциональные возможности, спроектированные под деятельность организации, масштабируемость и бесплатное сопровождение. Подсистема проста во внедрении и использовании, не требует дополнительного обучения.

Технология проектирования ИС представляет собой совокупность методов и средств проектирования ИС, а также организации и управления, внедрения и модернизации системы.

Проектирование ИС охватывает три основные области:

- проектирование объектов данных, которые будут реализованы в базе данных;
- проектирование программ, экранных форм, отчетов, которые будут обеспечивать выполнение запросов к данным;
- учет конкретной среды или технологии, а именно: топологии сети, конфигурации аппаратных средств, используемой архитектуры, параллельной обработки, распределенной обработки данных.

Существуют разные модели жизненного цикла: каскадная, итерационная, спиральная. Все они включают одинаковые стадии, но различаются последовательностью перемещения от стадии к стадии. К данным стадиям относятся: формирование требований к системе, проектирование, реализация, тестирование, эксплуатация.

Для проектирования подсистемы автоматизации процесса круалинга выбрана V-образная модель, которая позволяет сконцентрировать внимание на проверке результатов разработки, точно спланировать свойства программного обеспечения на различных этапах проектирования.

На рисунке 1.7 представлен жизненный цикл V-образной модели.



Рисунок 1.5 – Жизненный цикл V-образной модели

V-

образная модель жизненного цикла подсистемы имеет следующий ряд преимуществ:

- планирование действий происходит с ранних стадий разработки;
- упрощение отслеживания хода процесса разработки, так как завершение каждой фазы является контрольной точкой;
- обеспечение аттестации и верификации не только программного продукта, но и всех полученных внутренних и внешних данных.

Вывод по первому разделу

В разделе рассмотрена организационная структура управления ООО «Продвижение» и характеристика его видов деятельности, произведен сбор необходимой информации о деятельности seo-специалиста ООО «Продвижение», построена контекстная диаграмма организации, выявлены цели и функциональное назначение разрабатываемой подсистемы, необходимые для обеспечения повышения оперативности и качества работы технической оптимизации сайта, а также выбрана технология проектирования подсистемы.

2 Обоснование проектных решений

2.1 Обоснование проектных решений по техническому обеспечению

Техническое обеспечение представляет собой совокупность объединенных и взаимодействующих технических средств (ТС), предназначенных для автоматизированной обработки данных, а также включает методические материалы и персонал по разработке, внедрению и эксплуатации технических средств [6].

Одной из важнейших компонентов информационной системы является комплекс технических средств (КТС), состоящий из средств подготовки, регистрации, сбора, передачи, хранения, обработки, вывода и воспроизведения информации.

Технические средства должны соответствовать следующим требованиям:

- быть информационно совместимыми между собой и обслуживающим персоналом;
- структура КТС должна соответствовать структуре управления объектом, обеспечивая автоматизированное управление выполняемых функций;
- для обеспечения быстрого решения задач ТС должны быть качественной конструкции, иметь удобный интерфейс для работы пользователя;
- должен выполняться принцип экономичности выбора и использования ТС.

Необходимое для функционирования информационной подсистемы автоматизации процесса краулинга техническое обеспечение имеется в организации. Конфигурация компьютера с частотой процессора не менее 1,6 ГГц

ц, жестким диском от 120Гб и объемом оперативной памяти не менее 4Гб . Для одновременной работы пользователей подсистемы организация имеет локальную корпоративную сеть на основе сервера и принтер для вывода информации.

2.2 Обоснование проектных решений по информационному обеспечению

Информационное обеспечение (ИО) – важнейшая обеспечивающая подсистема автоматизированной информационной системы, предназначенная для снабжения пользователей информацией, характеризующей состояние управляемого объекта и являющейся основой для принятия управленческих решений. ИО представляет собой определенную совокупность средств и методов построения информационной системы экономического объекта [7].

Главным принципом создания ИС является решение задачи удовлетворения информационных потребностей пользователя и системы управления объектом.

Создание ИО проходит следующие этапы:

- исследование информационных потоков;
- разработка системы классификации и кодирования;
- разработка унифицированных форм представления данных в информационной базе;
- накопление массивов данных и работа с ними.

ИО определяет размещение формы организации информации в системе, представляя собой совокупность методов и средств построения и реализации информационной базы.

Проектные решения по информационному обеспечению обосновываются с точки зрения внешнего и внутримашинного обеспечения.

Внешнее (немашинное) ИО учитывает принципы автоматизации информационных процессов и состоит из системы классификации и кодирования, нормативно-справочных документов, оперативных документов, методических и инструктивных материалов. В соответствии с организационной структурой управления осуществляется движение этих документов.

Внутримашинное ИО включает информационные массивы, составляющие информационную базу системы и пакеты программ.

ИО реализуется в виде банков данных и банков знаний, в основе построения которых лежат модели накопления данных и представления знаний. Эти процессы должны быть формализованы на концептуальном и логическом уровнях.

2.3 Обоснование проектных решений по программному обеспечению

Программное обеспечение (ПО) – совокупность программных средств, выполненных вычислительной системой и обеспечивающих работу комплекса её технических средств, реализацию целей и задач информационной системы [7].

Состав программного обеспечения показан на рисунке 2.1.



Рисунок 2.1 – Состав программного обеспечения

К выбираемому программному обеспечению в данном случае относится среда программирования и программы функционального назначения.

Основным назначением ОС является осуществление вычислительного процесса в вычислительной системе, рациональное управление данными, заданиями, задачами, процессами и распределение вычислительных ресурсов между решаемыми задачами, а также обеспечение связи человека с компьютером.

Операционная система – комплекс системных и управляющих программ, обеспечивающих управление аппаратными средствами компьютера и предназначенных для эффективного использования всех ресурсов вычислительной системы [8].

Автоматизация задачи предполагается на операционной системе Microsoft Windows, так как она установлена в качестве рабочей станции организации.

Пакеты прикладных программ представляют собой совокупность программ, совместимых между собой и обеспечивающих решение задач из области знаний пакета. Могут быть программы общего назначения и программы функционального назначения [7].

К программам общего назначения можно отнести системы программирования на языках высокого уровня, систему управления базами данных (СУБД), программы-редактора текстов, изображений, издательские системы. Они реализуют типовые режимы работы вычислительной системы.

Предпочтение для разработки клиентской части информационной подсистемы автоматизации процесса краулинга было отдано изученной в процессе обучения в университете системе объектно-ориентированного программирования JavaFX, которая представляет собой инструмент быстрой разработки приложений и интегрированную среду прог

раммирования. Интегрированная среда программирования обеспечивает высокую скорость визуальной разработки, позволяет эффективно использовать компоненты, улучшенные инструментами и разномасштабными средствами доступа к базе данных.

2.4 Обоснование проектных решений по технологическому обеспечению

Технологический процесс обработки данных – комплекс технических средств, предназначенный для работы информационной системы, а также соответствующая документация на эти средства и технологические процессы [9].

Технологический процесс реализуется различными инструментальными средствами и определенными методами сбора, обработки и передачи результирующей информации пользователю для выполнения функций управления.

Под технологической операцией понимается законченная заключительная часть операции, обеспечивающая условия для начала следующей технологической операции.

По цели и месту выполнения технологический процесс можно разделить на четыре класса операций, отличающихся трудовыми затратами, связанными с их реализацией и распределением ошибок, вносимых в технологический процесс.

Первый класс операций характеризуется получением первичной информации, отражающей содержание процессов обработки документации организации. А именно к таким операциям относится получение количественной характеристики показателей, регистрация и сбор первичной информации.

Операции этого класса наиболее трудоемкие (трудовые затраты на выполнение операций составляют до 50% процентов всех работ), дорогостоящие и дающие наибольший процент ошибок в получаемых данных.

Целью второго класса операций служит ввод данных в ЭВМ и загрузка данных в информационную базу. В состав класса входят операции приема, ввода данных, контроля ошибок, загрузка данных в информационную базу и ведение информационной базы. Операции отличаются высокой трудоемкостью и множеством ошибок.

Операции третьего класса предназначены для выполнения обработки данных базы по алгоритмам и получение результативной информации. Операции этого класса характеризуются наибольшей степенью автоматизации процессов, наименьшей трудоемкостью и наименьшим количеством допускаемых ошибок.

Четвертый класс технологических операций имеют целью обеспечение достоверности и получение определенных отчетов по документам, подлежащих группировке по определенным ключевым признакам и анализ полученных данных с последующей распечаткой отчетных документов. К основным операциям относятся анализ и контроль полученных результатов, выявление и исправление ошибок по причине неправильности введенных исходных данных, сбоев в работе машины, ошибок пользователя оператора и ли программиста.

Недостатки технической оптимизации будут устранены с внедрением разрабатываемой информационной подсистемы автоматизации процесса краулинга. В подсистеме вся информация о входящих, внутренних и исходящих данных будет расположена в единой информационной базе. Подсистема будет иметь возможность парсить все ссылки сайта, а также анализировать код ответа сервера, упрощение сбора метатегов и анализа полученной обработанной информации в виде отчетных форм. Все перечисленное позволит сократить и упростить процесс техничес

кой оптимизации сайта, снизится время анализа оптимизации, тем самым увеличивается производительность работы seo-специалиста.

2.5 Обоснование выбора программных средств

При создании и проектировании информационной подсистемы автоматизации процесса краулинга предполагается использование следующих технологий:

- case-технологии;
- объектно-ориентированной технологии;
- дополнительных фреймворков;
- среды программирования.

CASE-

технология представляет собой методологию проектирования информационной системы, а также набор инструментальных средств, позволяющих в наглядной форме моделировать предметную область, анализировать модель на всех этапах разработки и сопровождения информационной подсистемы, разрабатывать приложения в соответствии с информационными потребностями пользователя.

Проведя анализ программных средств, и учитывая специфику разработки и преимущества продуктов, решено использовать для реализации клиентского приложения язык Java и фреймворк JavaFX.

Вывод по второму разделу

Необходимость покупки дополнительного оборудования отсутствует, так как в организации имеется необходимое оборудование для функционирования информационной подсистемы. Описаны требования к информационной подсистеме, выбраны и обоснованы проектные решения по информац

ионному, программному, технологическому обеспечению. Данные по разделу позволили перейти к разработке информационной подсистемы автоматизации процесса краулинга для организации ООО «Продвижение».

3 Проектная часть

3.1 Информационное обеспечение задачи информационной подсистемы автоматизации процесса краулинга

3.1.1 Информационная модель и ее описание

Под информационной моделью понимается модель объекта, представленная в виде информации, описывающей существенные для данного рассмотрения параметры и переменные величины объекта, связи между ними, входы и выходы объекта и позволяющая путем подачи на модель информации об изменениях входных величин моделировать возможные состояния объекта [10].

Проведя анализ технико-экономической составляющей организации, была разработана информационная модель «Как есть» и выявлены недостатки технической оптимизации сайта. [13]

Разработка информационной модели предполагает моделирование нового варианта организации технической оптимизации сайта ООО «Продвижение» –

информационной модели «Как должно быть». Описание автоматизации процесса краулинга производится с использованием известной и широко используемой методологии функционального моделирования IDEFO.

Предлагаемая модель seo-оптимизации коммерческого сайта представлена на рисунке 3.1.

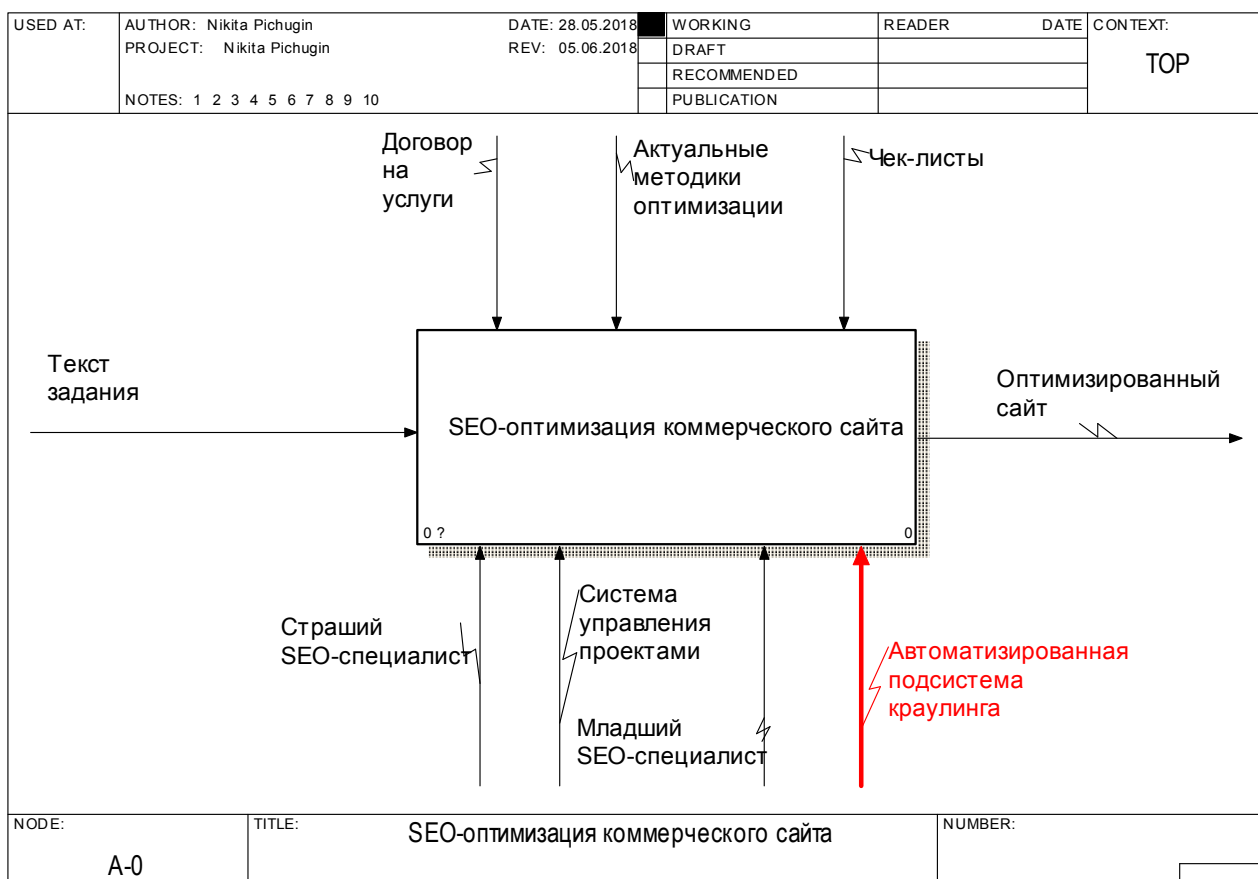


Рисунок 3.1 – Предлагаемая модель seo-оптимизации коммерческого сайта

На рисунке 3.2 представлена функциональная декомпозиция второго уровня подсистемы – процесса seo-оптимизации коммерческого сайта, состоящего из обработки текста задания и формирования оптимизирования сайта. На данной контекстной диаграмме автоматизированная система краулинга входит только техническую оптимизацию, именно этот процесс является одним из самых важных в процессе продвижения сайта в поисковой выдаче.

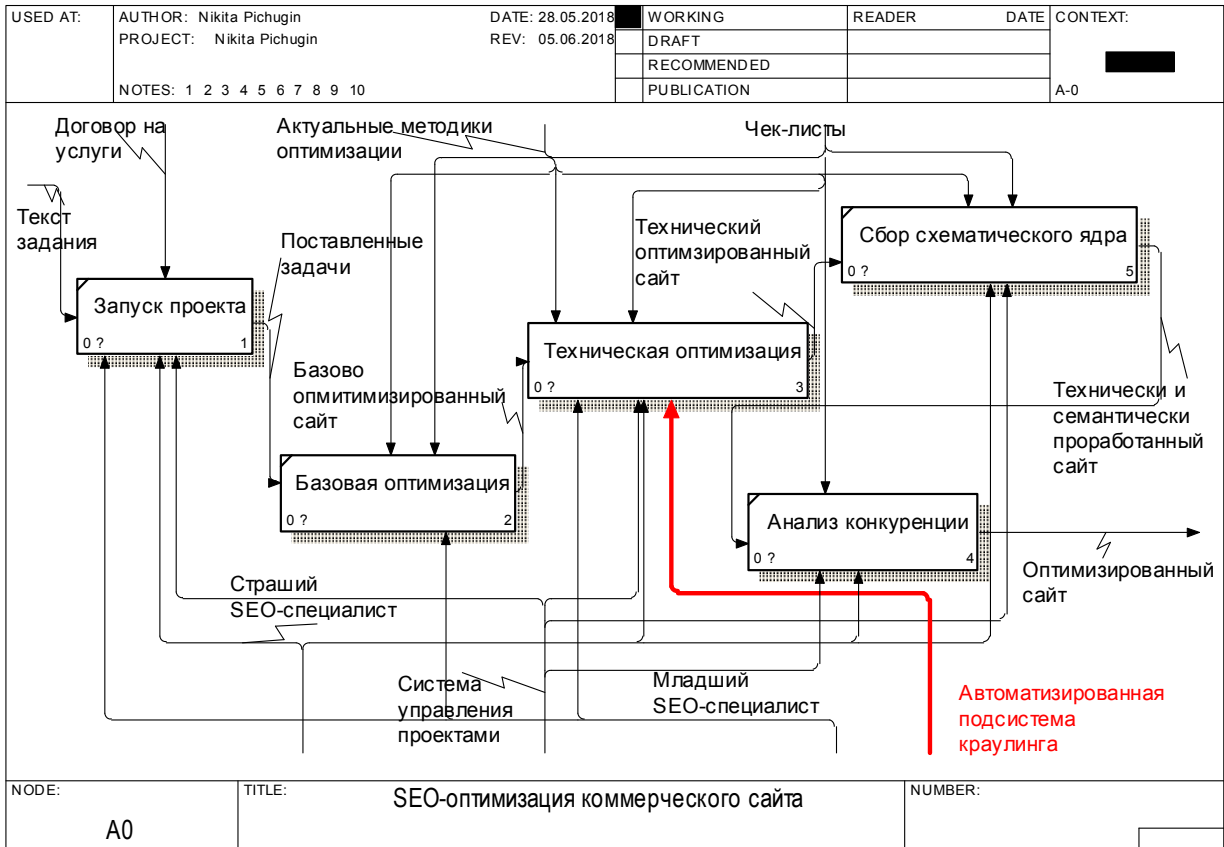


Рисунок 3.2 – Диаграмма декомпозиции процесса seo-оптимизации коммерческого сайта.

Автоматизация деятельности специалиста по поисковому продвижению происходит на этапах «Технической оптимизации». Более подробный процесс автоматизации процесса краулинга. Информационная подсистема автоматизации процесса краулинга в процессе парсинга сайта получает информацию о различных типах ошибок и не корректно заполненных данных. Самой распространённой проблемой технической оптимизации – это дубли страниц. Это может возникать в процессе разработки или не закрытия различных настроек в файле robots.txt, представлена реализация на рисунке 3.3

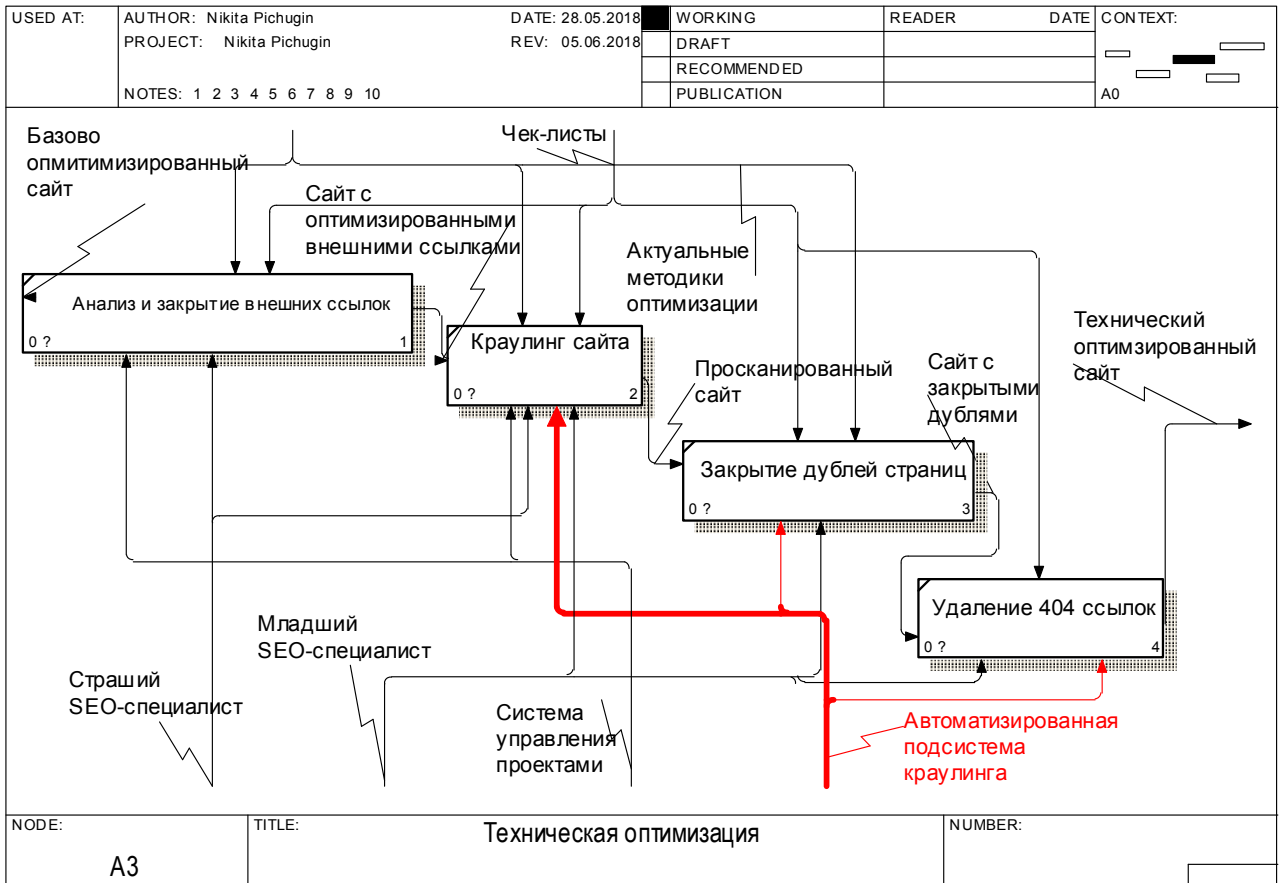


Рисунок 3.3 – Декомпозиция процесса «Техническая оптимизация»

Результатом проведенной работы является модернизация процесса краулинга путем использования информационной подсистемы для парсинга страниц сайта и обработки, анализа полученных данных. Это позволит решить проблему с скоростью обработки информации, получение оценки качества оптимизации сайта на уровне технических ошибок, а также ускорит производительность специалиста по поисковому продвижению.

3.1.2 Характеристика данных полученных при парсинге сайта

Входной оперативной информацией для информационной подсистемы являются ссылки страниц и код ответа сервера. Краулер сканирует все сс

ылки, находящиеся на странице сайта, собирает их в память, после обхода каждой ссылки он заносит их в таблицу, которая будет представлена после сканирования. Код ответ сервера –

это часть первой строки ответа сервера по протоколу HTTP, одна из самых встречаемых в интернете это 404 ошибка сервера, значащая, что запрошенный ресурс не найден, но в будущем возможно появится. Также на втором месте –

это 301 код сервера или другими словами редирект. Он позволяет перенаправить пользователя на другой ресурс, а также расценивается поисковыми и роботами, что страница существует и не удаляет из кеша «поисковиков»

Входной информацией для анализа контента сайта служит мета-теги Title, H1. Одни из самых важных тегов находящиеся на странице сайта, по которым поисковый робот и пользователь могут понять, что находится на странице и дать точный ответ на забитый в поиске запрос.

Для регистрации входящих ссылок присваивается уникальный номер и страницы, поступившего при сканировании сайта. В последствии можно узнать на какой стадии краулер просканировал определённую ссылку.

Для обновления входной информации подсистемы автоматизации процесса краулинга необходимо предусмотреть возможность корректировки или пересканирования сайта. Например, добавление диапазона сканирования или сканирования определённых ссылок.

3.1.4 Характеристика выходной и результативной информации

В информационной подсистеме процесса краулинга выходной информацией являются отчет о анализе и мониторинга ошибок сайта. Который в последствии возможно сохранить и передать на обработку.

Данные о входящем документе поступают в результате просканированных ссылок, найденных на сайте. Ссылки сканируются исходя из основного домена, поскольку если сайт имеет внешние ссылки, может быть просканированы другие данные исходящих ссылок.

Данные об исходящих документах формируются из колонок «Title», «Код ответа сервера», «H1». Самые важные данные для технической оптимизации и проработки семантического ядра, по которым специалист по поисковому продвижению может оценить оптимизацию сайта.

Результативными документами информационной подсистемы является аналитическая отчетность по трем видам потоков данных. Эти отчеты получаются в результате обработки информации из колонок «Title», «Код ответа сервера», «H1». Сравнение идет по количеству просканированных страниц сайта и найденных ошибок на сайте.

3.2 Программное обеспечение задачи информационной подсистемы

3.2.1 Структурная схема пакета и описание программных модулей

Разрабатываемая информационная подсистема автоматизирует процесс краулинга: просмотр мета-тега Title и H1, внутренних и исходящих страниц, выполнения анализа сайта на основе полученных данных.

Все действия программного продукта можно разделить на служебные и основные функции управления и обработки. Служебные функции призваны обеспечить безопасность ввода, обработки и хранения информации, облегчить работу с системой и сделать её приспособленной для пользователя. Основные функции отражают особенности процесса обработки информации, получения результатов.

Иерархия функций управления и обработки данных информационной подсистемы автоматизации процесса краулинга представлена в виде дерева функций на рисунке 3.4.

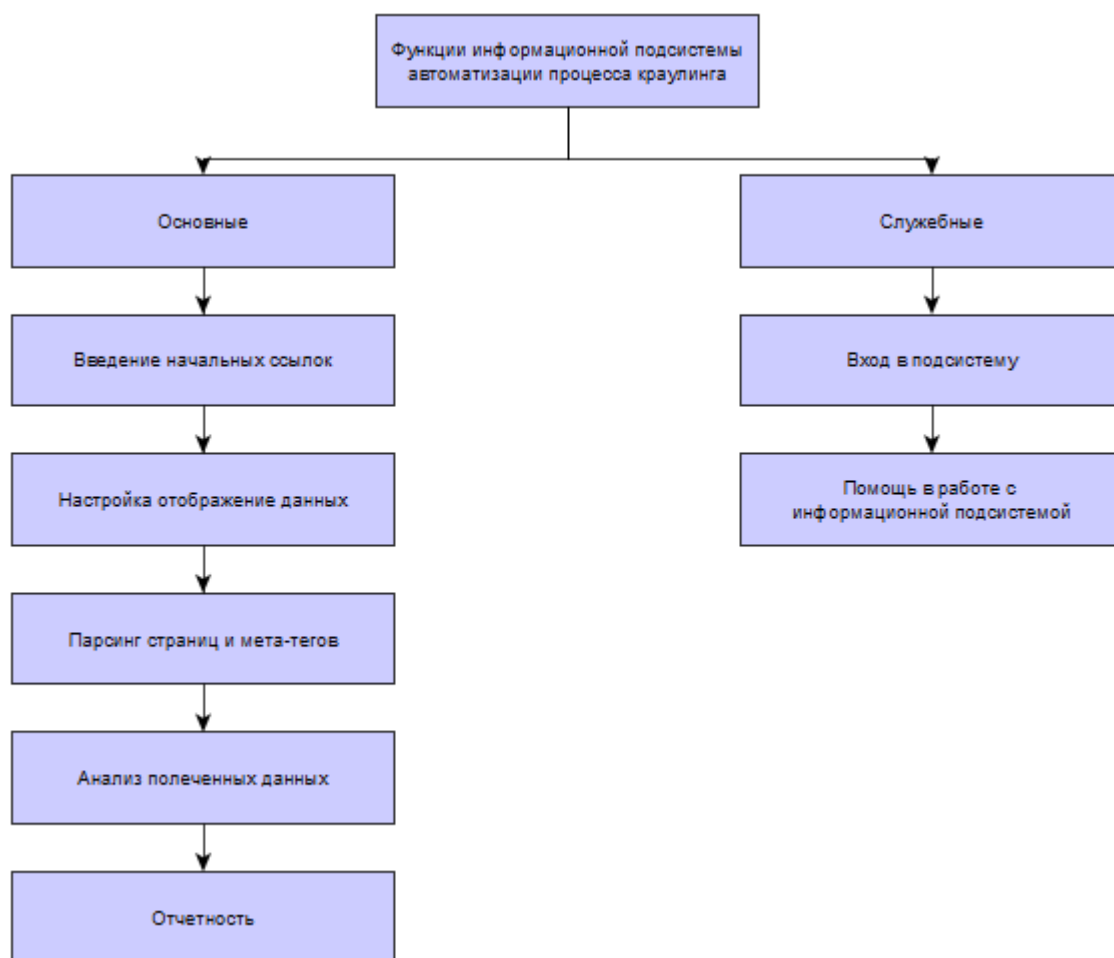


Рисунок 3.4 – Дерево функций информационной подсистемы

Основными функциями информационной подсистемы автоматизации процесса краулинга является парсинг страниц и мета-тегов, анализ полученных данных, а также формирование отчетности. К служебным функциям относится вход в подсистему и помощь в работе с информационной подсистемой.

На данном этапе разработки информационной подсистемы выбран диалоговый режим взаимодействия пользователя с подсистемой, который обе

спечивает возможность оперативного вмешательства человека в процесс обработки данных.

Сценарий диалога информационной подсистемы с пользователем состоит из следующих составляющих: основных и сервисных меню.

Основные меню, которые видит пользователь до выполнения основных функций, и предполагают обязательные действия пользователя при работе с подсистемой.

Процесс проектирования программного обеспечения начинается с уточнения и определения структурных компонентов и связей между ними.

Выделяют следующие исполняемые модули:

- модули, которые предназначены для ввода, обработки, хранения и выдачи информации,
- управляющие модули, обеспечивающие интерфейс с пользователем и переход к другим модулям.
- модули, которые выполняют дополнительные функции.

Описание программных модулей информационной подсистемы представлено в таблице 3.1.

Таблица 3.1 – Модули информационной подсистемы

№	Наименование модуля	Функции модуля информационной подсистемы
1	Модуль пользователя информационной подсистемы	Из данного модуля осуществляется переход ко всем модулям информационной подсистемы автоматизации процесса краулинга
2	Модуль получения информации о документах организации	Включает информацию по входящим, внутренним и исходящим страницам.
3	Модуль парсинга страниц	Содержит вводные начальные ссылки, которые необходимо парсить.
4	Модуль анализа контента сайта	Включает функции анализа контента сайта по полученным просканированным данным.
5	Модуль работы с отчетами и по обработке данных	Предназначение модуля состоит в создании отчета по обработке входящих, внутренних и исходящих данным.

3.3 Технологическое обеспечение задачи

Технологическое обеспечение задачи включает сбор и регистрацию информации, передачу, обработку, хранение, поиск, анализ и подготовку принятия решения.

Схема работы системы отображает управление операциями и потоками и данных в системе. Схема работы системы состоит из следующих символов:

- символов данных, указывающих на наличие данных;
- символов процесса, указывающих операции, которые следует выполнить над данными, а также определяющий логический путь, которого следует придерживаться;
- линейных символов, указывающих потоки данных между процессами или носителем данных, а также поток управления между процессами;
- специальных символов, используемых для облегчения написания и чтения блок-схемы [14].

Для начала сканирования сайта на технические ошибки необходимо выбрать в меню «Новый» и заполнить поле основным доменом и добавить несколько внутренних страниц сайта, для более корректного краулинга сайта. Если данные введены корректно, то происходит краулинг сайта по всем ссылкам, которые находятся на страницах сайта, иначе могут возникнуть ошибки при сканировании. Важно, что все ссылки необходимо брать и исключительно одного домена, чтобы в последствии не были просканированы внешние ссылки сайтов.

Для людей плохо видящих и других категории людей с ограниченными возможностями здоровья есть возможность настройки отображения ссылок любым удобным цветом.

3.4 Описание контрольного примера реализации подсистемы

После запуска информационной подсистемы автоматизации процесса краулинга необходимо ввести ссылки главного домена и внутренних страниц как это представлено на рисунке 3.5.

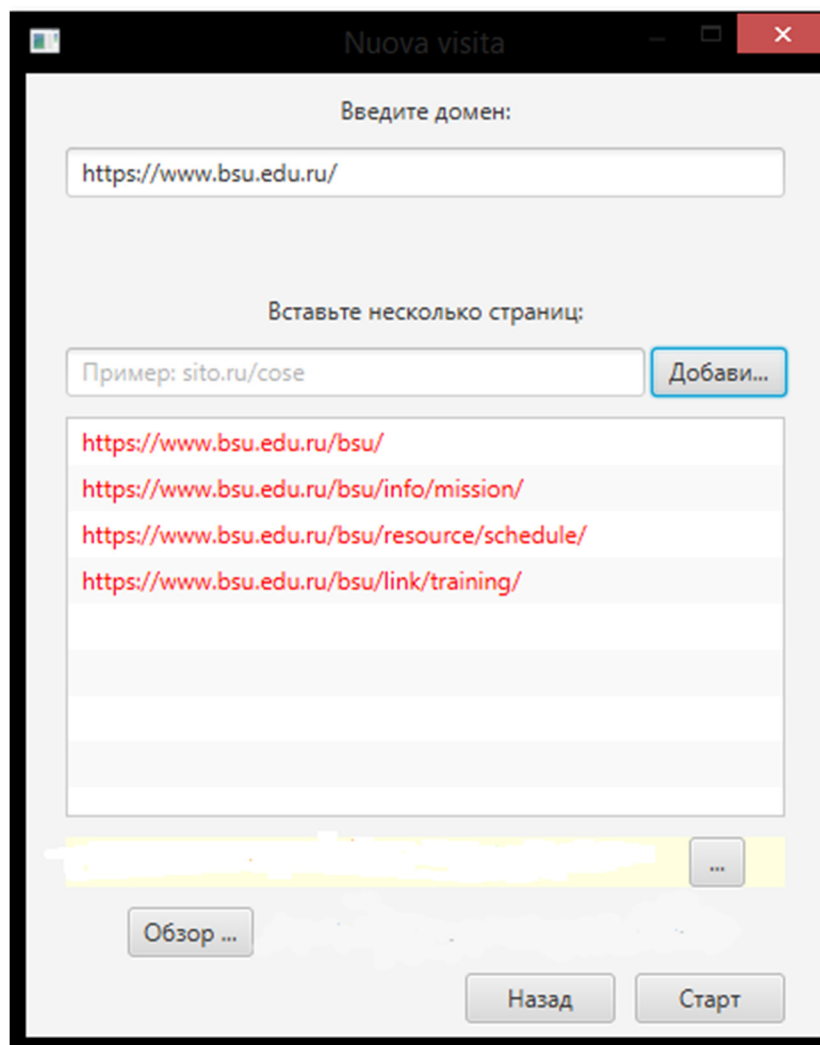


Рисунок 3.5 – Ввод главного домена и внутренних страниц

При вводе некорректных или неполных данных в поля осуществляется вызов информационного окна об ошибке, представленного на рисунке 3.6.

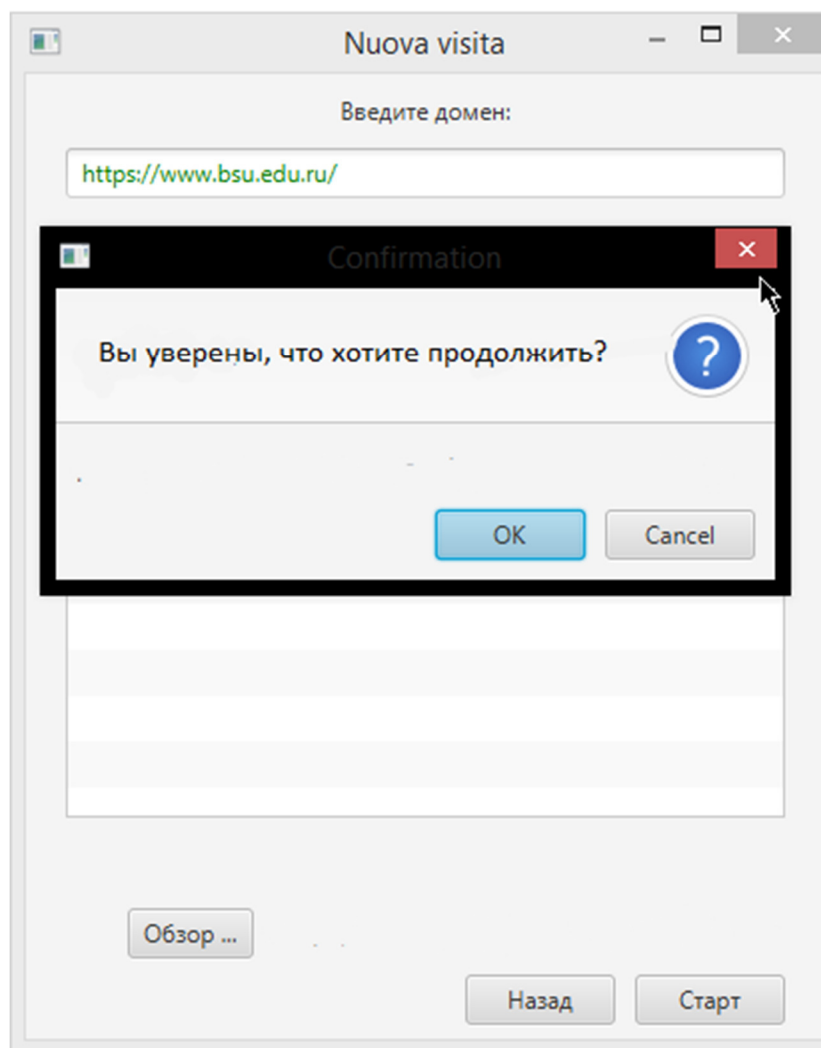


Рисунок 3.6 – Неправильный ввод данных домена

После успешного запуска подсистемы открывается новое и главное диалоговое окна, которое представляет визуализацию процесса краулинга. После успешного краулинга сайта мы можем увидеть таблицу с ссылками, размещенными на сайте и их мета-тегами.

Главное окно подсистемы автоматизации процесса краулинга представлено на рисунке 3.7.

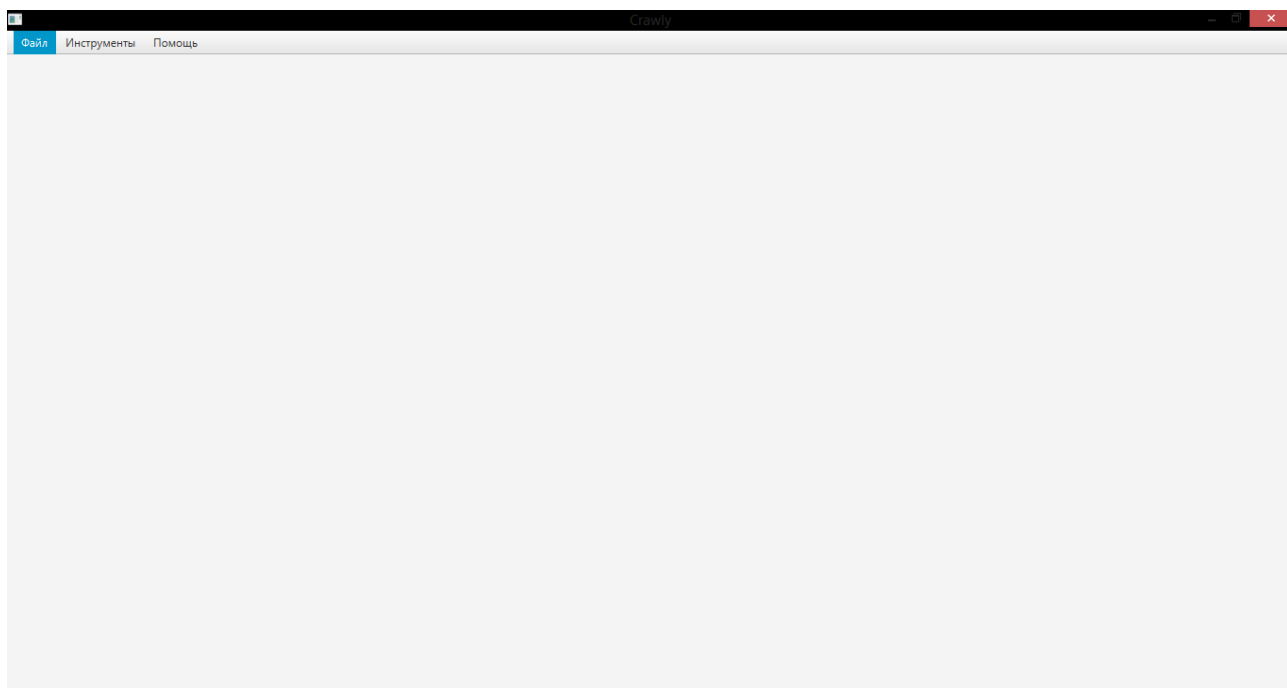


Рисунок 3.7 – Главное окно информационной подсистемы

Для осуществления открытия страницы в режиме браузера необходимо нажать два раза на ссылку, пример представлен на рисунке 3.8.

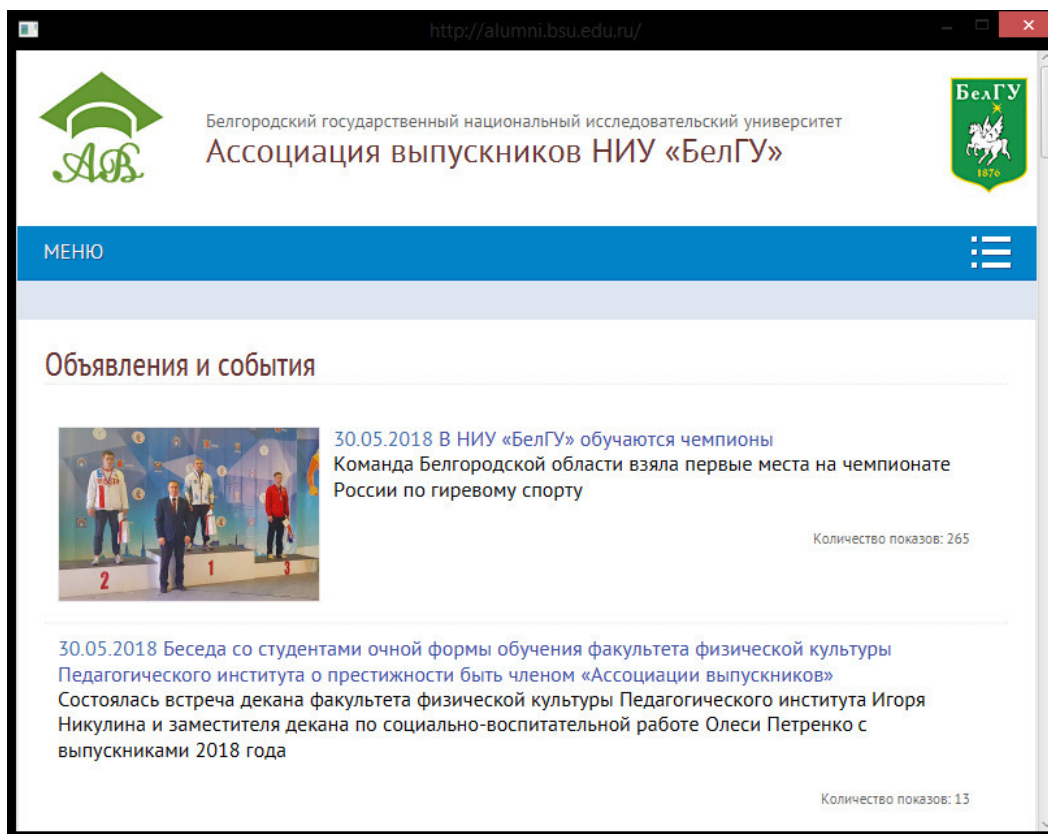


Рисунок 3.8 – Окно открытия страницы в режиме браузера

Каждая ссылка подсвечивает исходя того находится ли она выбранной странице или нет, что позволяет определить внутренние ссылки страницы, пример представлен на рисунке 3.9.

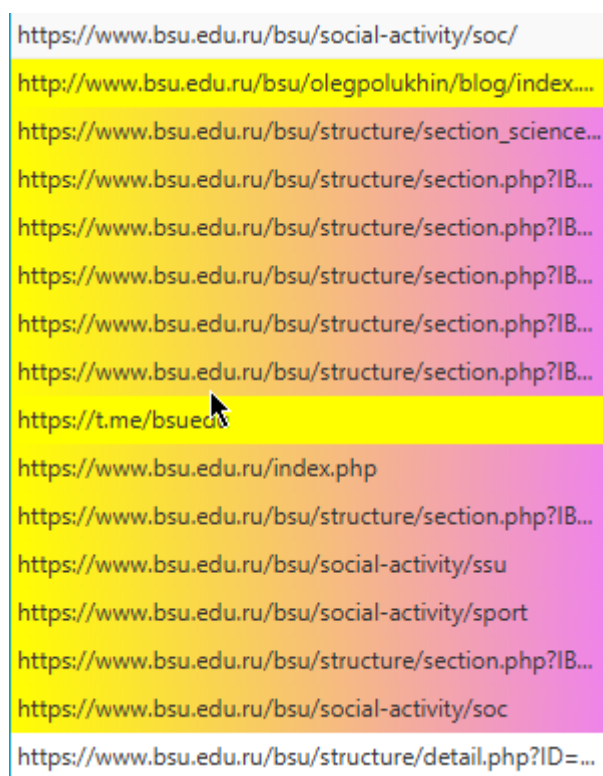


Рисунок 3.9 – Окно отображение связей между ссылками

Для настройки отображения ссылок и связи между ними, необходимо о выбрать «Настройки» в главном меню, есть возможность любой цвет, п одходящий пользователю, пример представлен на рисунке 3.10.

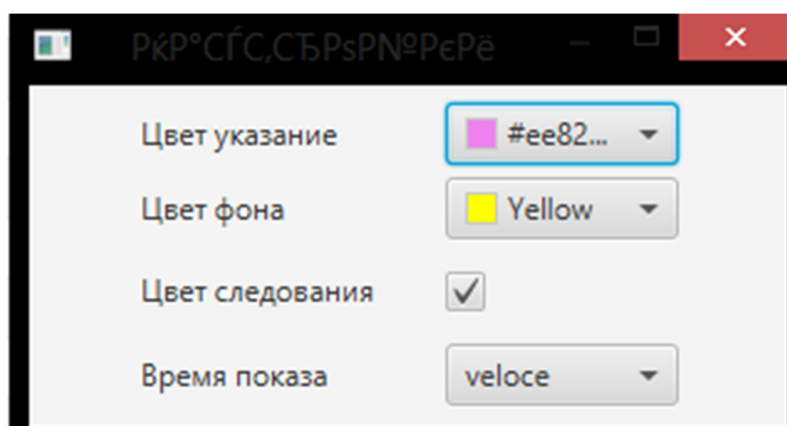


Рисунок 3.10 – Окно настройки отображения ссылок

После введение правильных данных домена и ссылок внутренних страниц сайта запускаем процесс краулинга. Робот обходит все страницы, которые найдет, кроме внешних ссылок. Выводит информацию о каждой просканированной странице, а именно: метатеги Title и H1, номер ссылки, код ответа сервера. Справа диаграммы показывают наличие ссылок с кодом ответа 200 имеют красный цвет, 404 синий, 3xx зеленый эта визуализация позволяет пользователю сразу обратить внимание на просканированные данные и проанализировать процесс краулинга, представлено на рисунке 3.11.

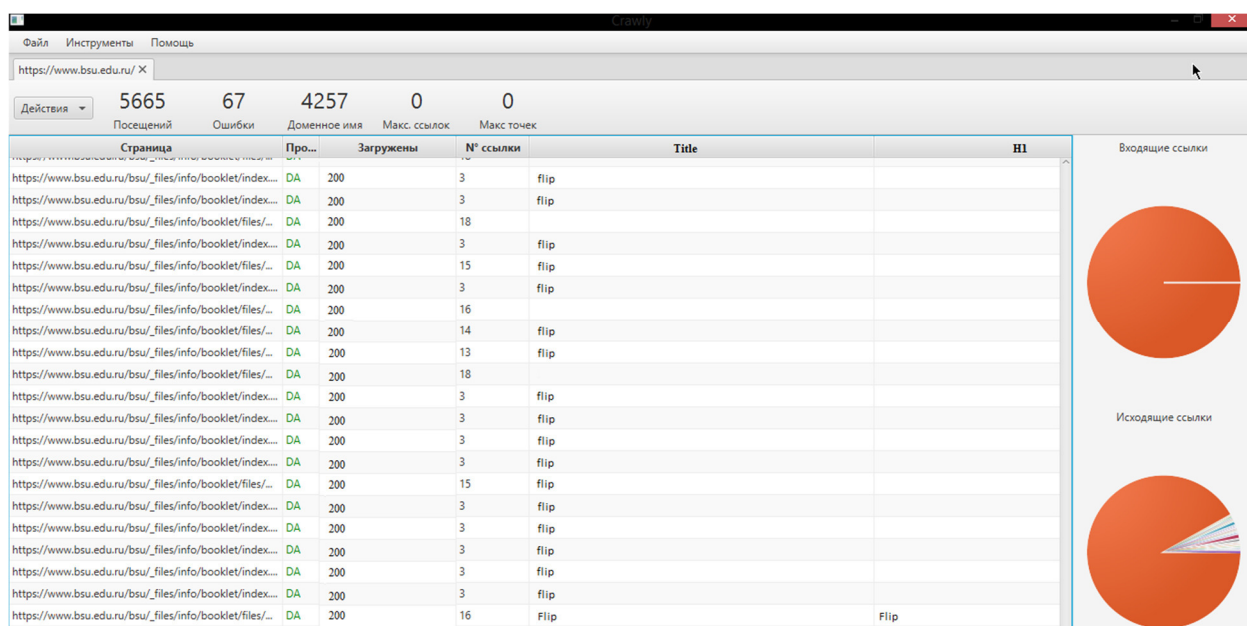


Рисунок 3.11 – Окно просканированного сайта

После успешного сканирования сайта, можно добавить отдельные ссылки для повторного сканирования (поддомены, внешние ссылки, не просканированные внутренние ссылки). Так же во время процесса сканирования краулером можно приостановить процесс, для этого необходимо выбрать в верхней части меню «Пауза» и так же можно возобновить процесс на прошлом месте, представлено на рисунке 3.12.

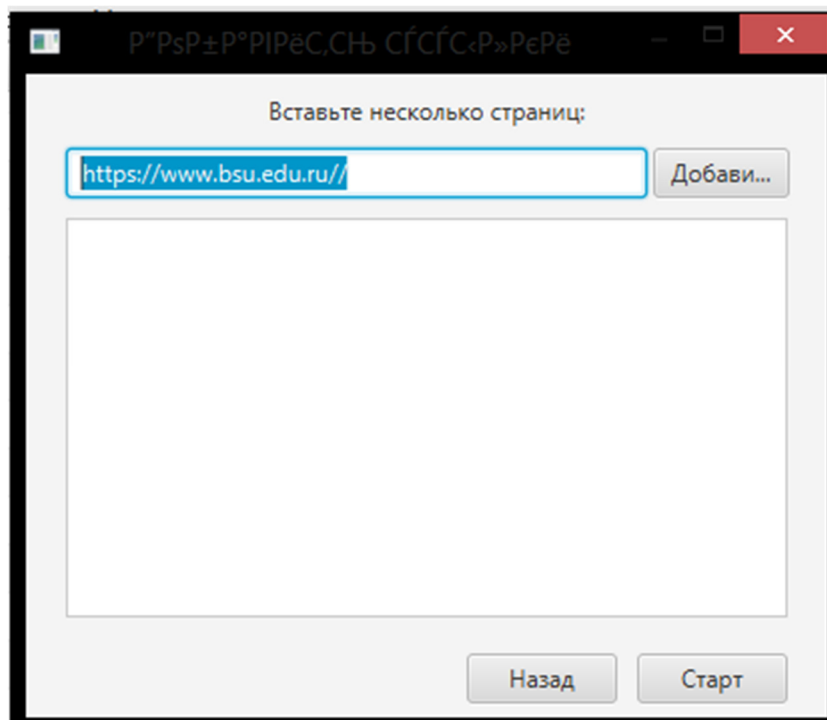


Рисунок 3.12 – Ввод дополнительный ссылок

Перед началом сканирования или при повторном сканировании сайта краулером, есть возможность выделение диапазона сканирования. Диапазон можно выбирать от и до количества сканируемых ссылок, одна из особенностей от других краулеров –

это возможно выделения диапазона по вложенности, что позволяет видеть несколько вложенностей, а не все (например, определенные группы, но не товары, вложенные в них), представлено рисунок 3.13.

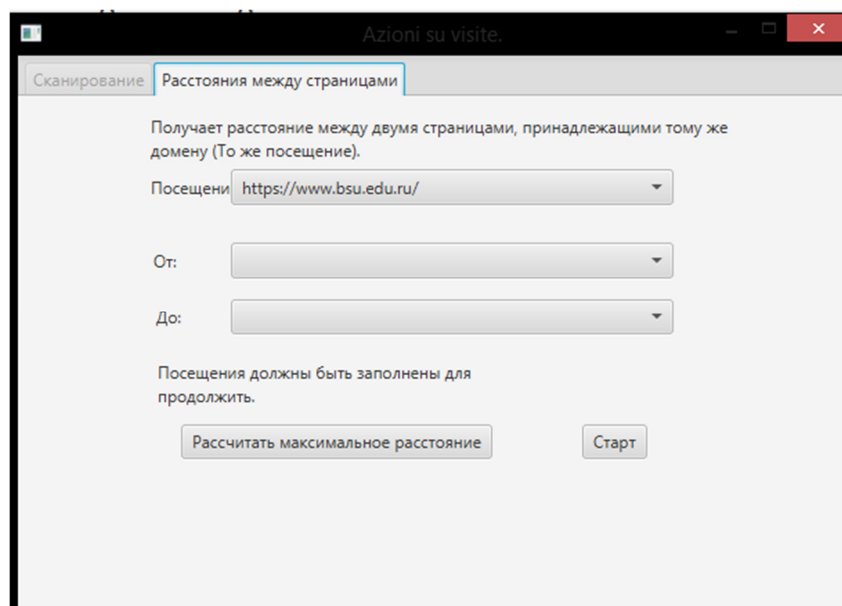


Рисунок 3.13 – Окно выделение диапазона сканирования

Если есть необходимость просканировать именно те страницы, которые необходимы, то необходимо добавить их в файл формата CSV в один столбик, после на окне добавления ссылок на сайт, необходимо нажать на кнопку «Обзор» и выбрать файл CSV, представлено на рисунке 3.14.

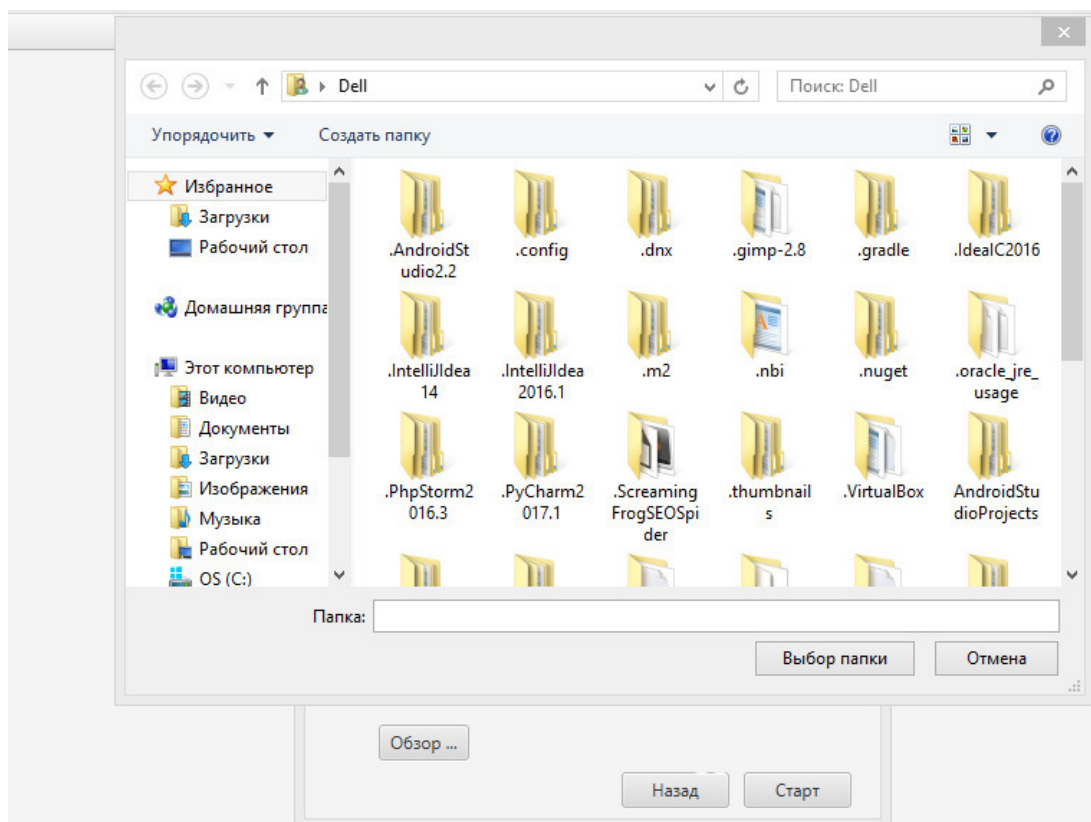


Рисунок 3.14 – Форма «Документация организации»

После окончания сканирования всего сайта, есть функция оценки сайта по шкале от 0 до 100. Оценка происходит по найденным ошибкам в процессе сканирования краулером (например, открытые фильтры, дублирование мета-тегов, количество страниц с ответом сервера 4xx, 3xx.). Алгоритм оценки работает в соотношении просканированных страниц и найденных ошибок. После получения оценки, есть возможность формирования файла PDF, представлено на рисунке 3.15.

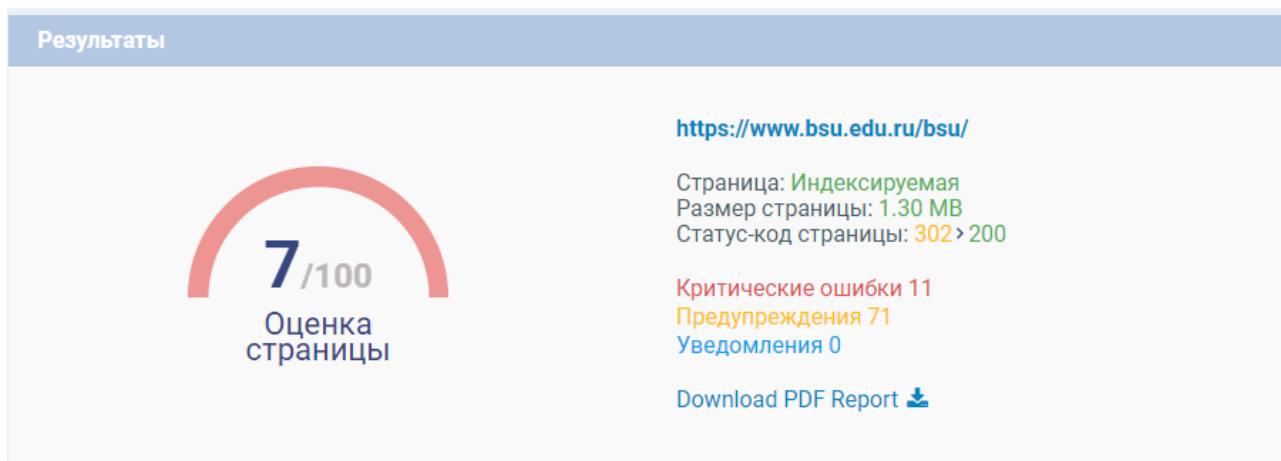


Рисунок 3.15 – Результат поиска внутреннего документа

Таким образом, были протестированы функциональные возможности информационной подсистемы автоматизации процесса краулинга для деятельности организации ООО «Продвижение».

3.5 Расчет экономической эффективности

Основными задачами разработки информационной подсистемы являлись уменьшение объема непроизводительной работы по технической оптимизации сайта, обеспечение увеличения скорости анализа контента сайта и создание отчета, создаваемых в организации, а также осуществление контроля своевременного исполнения аудита или поставленных сроков задач.

Экономический эффект от внедрения информационной подсистемы можно оценить на основании прямой и косвенной эффективности.

Прямой экономический эффект – экономия материально-трудовых ресурсов и денежных средств, полученная в результате сокращения расхода основных и вспомогательных материалов вследствие автоматизации работ.

Косвенный эффект – экономия средств в процессе производства, проявляющаяся в конечном результате хозяйственной деятельности предприятия.

Осуществить оценку проекта на степень его реализуемости – это значит оценить социально-экономический эффект проекта на той стадии, когда мало достоверной информации, поэтому и оценки должны быть интервальными.

Поскольку в подсистеме автоматизации процесса краулинга используется централизованное хранение информации, то увеличится скорость обработки информации, что приведет к росту производительности труда seo-оптимизатора. Календарный план-график проектирования информационной подсистемы включает этапы, представленные в таблице 3.2.

Таблица 3.2 – Календарный план-график проектирования подсистемы

Этап	Длительность
Предпроектные исследования	1,5 недели
Проектирование	1,5 недели
Разработка и отладка программного кода	1,5 недели
Тестирование системы	1 неделя
Внедрение проекта	1 неделя
Итого:	7 недель

Единовременные затраты (Z_k), связанные с разработкой и внедрением данной информационной подсистемы, будут состоять из заработной платы разработчика ПП ($Z_{зп}$), накладных расходов (Z_n), капитальных затрат (K), включающих затраты на приобретение вычислительной техники ($K_{вт}$), приобретение пакетов прикладных программ ($K_{ппп}$) и операционных систем ($K_{ос}$). Единовременные затраты вычисляются по формуле 3.1.

$$Z_k = K + Z_{зп} = K_{ппп} + K_{ос} + K_{вт} + Z_{зп}, \quad (3.1)$$

Накладные расходы (Z_n), связанные с проектированием и отладкой и информационной подсистемы, в том числе стоимость расходных материалов представлены в таблице 3.3.

Таблица 3.3 – Накладные расходы

Материалы	Потребность, шт.	Стоимость, руб.
Бумага А4	1упак.	180
Картридж для принтера	1 шт.	1945
USB-Flash накопитель	1 шт.	275
Итого		2400

Дополнительные затраты на покупку оборудования не потребуются, так как все сотрудники обеспечены компьютерами.

Заработная плата разработчика ПО определяется по формуле 3.2.

$$Z_{зп} = Z_{ср,дн} \cdot T, \quad (3.2)$$

где $Z_{ср,дн}$ –

среднедневная зарплата разработчика ПО, рассчитанная на основе его месячного оклада и числа рабочих дней в месяце;

T – суммарная длительность дней на разработку системы.

Так как в неделе 5 рабочих дней, то суммарная длительность дней на разработку равна

$$T = 7 \text{ недель} \cdot 5 \text{ рабочих дней} = 35 \text{ рабочих дней}$$

Ставка социального налога состоит из отчислений в Фонд медицинского страхования (5,1%), отчислений в Пенсионный фонд (22%), отчисления в Страховой фонд (2,9%). Ставка единого социального налога на фонд оплаты труда равна

$$22\% + 2,9\% + 5,1\% = 30\%$$

Приняли $Z_{ср,дн}$ равной 600 руб. Исходя из этого, сумма заработной п

латы с учетом ставки единого социального налога для разработчика на период работ составит

$$Z_{\text{зн}} = Z_{\text{ср}} \cdot T \cdot 1,3 = 600 \cdot 35 \cdot 1,3 = 27300 \text{ руб.}, \quad (3.3)$$

Общая сумма единовременных затрат (Зк)

$$Z_{\text{к}} = 2400 + 27300 = 29700 \text{ руб.}$$

В ходе эксплуатации подсистемы будут возникать постоянные затраты, представленные в таблице 3.4.

Таблица 3.4 – Постоянные затраты системы

Постоянные затраты	Стоимость, руб/год
Затраты на ремонт оборудования	2917,5
Затраты на электроэнергию	2738,54
Амортизационные отчисления	3890
Итого	9546,04

Ремонт и обслуживание подсистемы включает в себя администрирование и сопровождение (затраты на поддержку, услуги по сопровождению работы системы). Ремонт и обслуживание системы составит 15% в год от стоимости технического обеспечения.

$$C_{\text{рем}} = P \cdot 0,15, \quad (3.4)$$

Основным сотрудником, производящий анализ и разработку стратегии продвижения сайта, является seo-оптимизатор. Стоимость оборудования рабочего места seo-оптимизатора можно оценить по остаточной стоимости так, как указано в таблице 3.5.

Таблица 3.5 – Стоимость технического средства

Оборудование	Сумма, руб
Системный блок	53200
2 Монитора	24300

Клавиатура, мышь	2650
Итого	80150

Затраты на ремонт компьютера делопроизводителя организации составляют

$$C_{\text{рем}} = 53200 \cdot 0,15 = 7980 \text{ руб./год}$$

Затраты на электроэнергию можно рассчитать по формуле 3.5.

$$C_{\text{эл}} = C_{\text{э}} \cdot \Phi_{\text{д}} \cdot M, \quad (3.5)$$

где $C_{\text{э}}$ – стоимость 1 кВт/ч электроэнергии;

$\Phi_{\text{д}}$ – действительный годовой фонд времени работы ЭВМ в часах;

M – мощность ЭВМ в кВт/ч электроэнергии.

Поправочный коэффициент для расчета действительного годового фонда работы ЭВМ равен 0,95, так как компьютер работает не полные 8 часов.

$$\Phi_{\text{д}} = 237 \text{ рабочих дней} \cdot 8 \text{ часов/день} \cdot 0,95 = 1801,2 \text{ ч}$$

$$C_{\text{эл}} = 3,62 \cdot 1801,2 \cdot 0,42 = 2738,54 \text{ рублей/год}$$

Согласно утвержденной классификации ОС по амортизационным группам компьютерная техника относится к 3-ей амортизационной группе. Срок эксплуатации ЭВМ составляет 3-5 лет. Норма начисления амортизации на вычислительную технику = 100% / 5 лет = 20% в год.

$$C_{\text{ам}} = P \cdot 0,20, \quad (3.6)$$

где P – стоимость ЭВМ.

$$C_{\text{ам}} = 53200 \cdot 0,2 = 10640 \text{ руб.}$$

$$\text{Получим: } C_{\text{экс}} = 7980 + 2738,54 + 10640 = 21358,54 \text{ руб.}$$

В среднем на сбор, фиксацию, обработку и анализ информации, а также на ручное составление необходимой отчетности в месяц тратится 672

часа рабочего времени, что составляет в денежном выражении 65520 руб. (21 рабочий день * 600 руб. дневная ставка * 1,30 ставка единого социального налога * 4 человека).[20]

Затраты до внедрения информационной подсистемы составляют 65520 руб. в месяц.

Прямые затраты после внедрения подсистемы составят 168 часов (21 рабочий день*8 часов), что в денежном выражении равно 16380 руб. (21 рабочий день * 600 руб. дневная ставка*1,3 ставка социального налога). В среднем прямая экономия на заработной плате и обязательных отчислениях с фонда оплаты труда составит 49140 рублей в месяц (65520 руб. – 16380 руб.), что в годовом выражении примерно равно 589680 руб. (49140 руб. * 12 месяцев).

Поскольку в первом году 6 месяцев выпадает, так как именно в это время велась работа по разработке и внедрению подсистемы, то сумма составит

$$589680/12*8 = 393120 \text{ руб.}$$

Расчет экономического эффекта произведен на основании чистого дисконтированного дохода (ЧДД), определенного как сумма текущих эффектов за весь расчетный период, приведенная к начальному шагу.

Под дисконтированием понимается способ приведения будущей стоимости инвестируемых средств к их текущей стоимости.

Дисконтная ставка –

процентная ставка, по которой осуществляется приведение будущей стоимости инвестируемых средств к их текущей стоимости. Норма дисконта определяется по формуле 3.7.

$$\alpha = \frac{E_H - B}{1+B}, \quad (3.7)$$

где α – норма дисконта, равная приемлемой для инвестора норме дохода на капитал;

E_H – ставка рефинансирования, устанавливаемая Центробанком РФ;

v – прогнозируемый уровень инфляции.

С 27 марта 2017 года ставка рефинансирования составляет 9,75 процентов годовых (информация Банка России от 24 марта 2017 г.). Согласно прогнозам Министерства экономического развития, уровень инфляции в России будет составлять 7 %, то норма дисконта

$$\alpha = (0,0975 - 0,07) / 1,07 = 0,026$$

Инвестиции представляют собой вложения капитала в различные объекты предпринимательской деятельности с целью получения прибыли, а также сохранения и увеличения капитала. Чистый дисконтированный доход определяется как сумма текущих эффектов за весь расчётный период и вычисляется по формуле 3.8. [23]

$$\text{ЧДД} = \sum_{t=1}^T (D^t - P^t) \cdot \frac{1}{(1+\alpha)^t}, \quad (3.8)$$

где D^t – доходы t -го периода;

P^t – расходы t -го периода;

α –

норма дисконта, равная приемлемой для инвестора норме дохода на капитал;

t – годы.

Индекс доходности вычисляется по формуле 3.9.

$$\text{ИД} = \frac{\sum_{t=1}^T D^t \cdot \frac{1}{(1+\alpha)^t}}{\sum_{t=1}^T P^t \cdot \frac{1}{(1+\alpha)^t}}, \quad (3.9)$$

Дисконтный множитель вычисляется по формуле 3.10.

$$DM = \frac{1}{(1+\alpha)^t}, \quad (3.10)$$

Экономические показатели информационной подсистемы представлены в таблице 3.6.

Таблица 3.6 –

Экономические показатели разработанной информационной подсистемы

	Периоды					
	0 год	1 год	2 год	3 год	4 год	5 год
Инвестиции	29700					
Экономия		294840	589680	589681	589682	589683
Расходы		9546,04	9546,04	9546,04	9546,04	9546,04
Дисконтный множитель	1	0,99122	0,98253	0,9739	0,96536	0,95688
Текущий эффект	-29700	383573,96	580133,96	580134,96	580135,96	580136,96
Дисконтированный эффект	-29700	380206,18	569999,02	564993,44	560040,05	555121,45
ЧДД	-29700	350506,18	920505,20	1485498,64	2045538,69	2600660,14

Индекс доходности строится из тех же элементов, что и чистый дисконтированный доход. Индекс доходности больше единицы, так как чистый дисконтированный доход положителен. На рисунке 3.34 приведен график, показывающий величину ЧДД нарастающим итогом за 5 лет использования информационной подсистемы. [23]

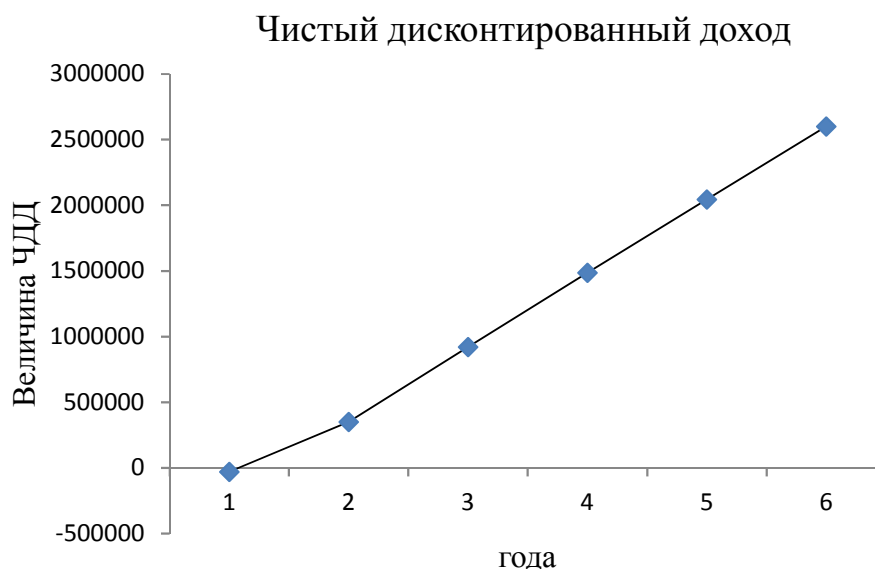


Рисунок 3.16 – График срока окупаемости подсистемы

Подсистема окупится уже в первый год после ее внедрения. Экономия от внедрения информационной подсистемы во много раз превышает расходы, поэтому чистый и дисконтированный эффекты видны уже в течение первого года эксплуатации. Представленный расчет можно считать приблизительным.

Вывод по третьему разделу

Была описана созданная информационная подсистема автоматизации процесса краулинга, ее структура и работа. Представлена модель «Как должно быть» по усовершенствованию процессов технической оптимизации сайтов, выявлены основные и служебные функции подсистемы, предоставлена схема работы программы, наглядно предоставлены и протестированы результаты работы информационной подсистемы автоматизации процесса краулинга, а также рассчитана экономическая эффективность подсистемы для ООО «Продвижение».

ЗАКЛЮЧЕНИЕ

Для достижения поставленной цели выпускной квалификационной работы было определено технико-экономическое состояние предприятия и проанализирована деятельность SEO-оптимизатора, выявлены проблемы и недостатки автоматизации технической оптимизации сайта, определена цель и назначение автоматизированного варианта решения задачи, проведен анализ существующих разработок и обоснование выбора технологии проектирования информационной подсистемы.

Проведено обоснование проектных решений по техническому, информационному, программному, технологическому обеспечению, а также выбраны программные средства. В процессе проектирования была обоснована необходимость создания информационной подсистемы автоматизирующую процесс краулинга и сформулированы требования к ней.

Была описана созданная информационная подсистема, ее структура и функциональные возможности. Были наглядно представлены результаты работы информационной подсистемы автоматизации процесса краулинга. После чего, была рассчитана экономическая эффективность подсистемы для ООО «Продвижение».

Результатом выпускной квалификационной работы стала информационная подсистема автоматизации процесса краулинга для анализа контента сайта.

Информационная подсистема автоматизации процесса краулинга удовлетворяет установленным требованиям организации и планируется к внедрению во внутренние процессы технической оптимизации сайта специалистом по поисковому продвижению.

Цель выпускной квалификационной работы достигнута в полном объеме.

Поставленные задачи выпускной квалификационной работы были успешно решены:

- проведено исследование деятельности организации по технической автоматизации сайта;
- определен путь решения проблем и недостатков в работе seo-оптимизатора организации;
- спроектирована информационная подсистема;
- реализована информационная подсистема;
- протестирована информационная подсистема автоматизации процесса краулинга для анализа контента сайта.

Это позволило решить ряд проблем в деятельности специалиста по поисковому продвижению:

- ускорение в анализе технической оптимизации сайта;
- выделение диапазона по вложенностям сайта;
- составления аккаунт-менеджеру оценки о состоянии технической части сайта;
- составление отчета о дублирующихся страниц сайта.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Ашманов, И.С. Оптимизация и продвижение сайта в поисковых системах / А.С. Иванов. – Санкт-Петербург.: Питер, 2013.– 464 с.
2. Баскакова, О. В. Экономика предприятия (организации) / О.В. Баскакова, О.В. Сейко. – М.: Издательско-торговая корпорация «Дашков и К», 2013.– 372 с.
3. Неелова, Н.В. SEMBOOK. Энциклопедия поискового продвижения / Е.С. Кондюкова. – Екатеринбург.: ГОУ ВПО УГТУ-УПИ, 2006.– 116 с.
4. ГОСТ Р. 15489-1-2007 Система стандартов по информатизации, библиотечному и издательскому делу. Управление документами. Общие требования. – М.: Стандартинформ, 2007.– 34 с.
5. Ясенев, В.Н. Автоматизированные информационные системы в экономике: учебно-методическое пособие [Электронный ресурс] / В.Н. Ясенев. – Н. Новгород, 2007.–
Режим доступа: <http://www.iee.unn.ru/files/2014/09/Kniga-slajdy.pdf>.
6. ГОСТ 2.601-2013 Единая система конструкторской документации. Эксплуатационные документы. – М.: Стандартинформ, 2014.– 36 с.
7. Гвоздева, В.А. Основы построения автоматизированных информационных систем/ В.А. Гвоздеева, И.Ю. Лаврентьева. – М.: ИД «Форум»: ИНФРА-М, 2009.– 320 с.
8. Морозевич, А.Н. Основы информатики / А.Н. Морозевич, Н.Н. Говядинова. – Мн: Новое знание, 2003.– 543 с.

9. Моисеенко, Е.В. Информационные технологии в экономике [Электронный ресурс] / Е.В. Моисеенко, Е.Г. Лаврушина. – Владивосток: Издательство ВГУЭС, 2004.–
Режим доступа: www.vvsu.ru/files/579B03E9-4BE4-4A07-98F4-E386FCEA1E8B.pdf.
10. ГОСТ 34. 003-90 Информационная технология (ИС). Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения. – М.: Стандартиформ, 2009.– 26 с.
11. ГОСТ 19.701-90 ЕСПД. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения. – М.: Стандартиформ, 2010.– 21 с.
12. ГОСТ 7.32-2001 Отчет о научно-исследовательской работе. Структура и правила оформления. – М.: Стандарт, 2002.– 27 с.
13. ГОСТ 7.1-2003 Библиографическая запись. Библиографическое описание. Общие требования и правила составления. – М.:ИПК Издательство стандартов, 2004.– 57 с.
14. ГОСТ 19.102-77 ЕСПД Стадии разработки. – М.:Стандартиформ, 2010.– 3 с.
15. Рогожин, М.Ю. Документационное обеспечение управления: учебно-практическое пособие/М.Ю. Рогожин. – Тк Велби, Изд-во Проспект, 2008.– 384 с.
16. Таненбаум, Э. Современные операционные системы [Электронный ресурс] / Э. Таненбаум, Х. Бос. – СПб.: Питер, 2015.–
Режим доступа: http://math.kubsu.ru/Debian_Tanenbaum.pdf.
17. Гахов, Р.П. Методы и средства проектирования информационных систем и технологий: учебно-

методический комплекс [Электронный ресурс] / Р.П. Гахов, 2013. Режим доступа: <http://pegas.bsu.edu.ru/course/view.php?id=5906>.

18. Пшенко, А.В. Документационное обеспечение управления / А.В. Пшенко, Л.А. Доронина. – М.: Издательский центр «Академия», 2014.– 224 с.

19. Хоменко, А.Д. Работа с базами данных в JavaFX/ А.Д. Хоменко, С.Е. Ададунов. – СПб.: БХВ-Петербург, 2006.– 496 с.

20. Могилев, А.В. Информатика: учебное пособие /А.В. Могилев, Н.И. Пак, А.В. Хеннер. – М.: Издательский центр «Академия», 2004.– 843 с.

21. Маторин, С.И. Теория систем и системный анализ / С.И. Маторин, О.А. Зимовец. – Белгород: Изд-во НИУ «БелГУ», 2012.– 288 с.

22. Файзрахманов, Р.А. Проектирование автоматизированных информационных систем на основе объектно-ориентированного подхода: учебное пособие [Электронный ресурс] / Р.А. Файзрахманов, А.В. Архипов. – Пермь: Изд-ва Перм. гос. техн. Ун-та, 2011.– Режим доступа: <http://elib.pstu.ru/docview/?id=2719.pdf>.

23. Брябин, В.М. Программное обеспечение персональных ЭВМ / В.М. Брябин. – М.: Наука, 1989.– 272с.

24. Лопатникова, Е.А. Делопроизводство: образцы документов с комментариями /Е.А. Лопатникова. – М.:Омега-Л, 2008.– 319 с.

25. Михелев, В.М. База данных и СУБД: учебное пособие / В.М. Михелев. – Белгород: Изд-во БелГУ, 2007.– 200с.

26. Полякова, Л.Н. Основы Java: Курс лекций: учебное пособие / Л.Н. Полякова. – М.: ИНТУИТ.РУ, 2004.– 368 с.

27. Смирнова, Е.П. Делопроизводство для секретаря/ Е.П. Смирнова, Ю.А. Петрова. – Саратов: Корпорация «Диполь», 2012.– 165с.

28. Соколов, В.С. Документационное обеспечение управления / В.С. Соколов – М.: ФОРУМ – ИНФРА-М, 2005.– 233с.

29. Коберн, А. Быстрая разработка программного обеспечения / А. Коберн. – М.: Лори, 2013.– 336 с.

30. Федоренко, Ю.П. Алгоритмы и программы на Java / Ю. П. Федоренко. – М.: ДМК Пресс, 2010.– 544с.

ПРИЛОЖЕНИЕ А

Контекстная диаграмма seo-оптимизации сайта А.1

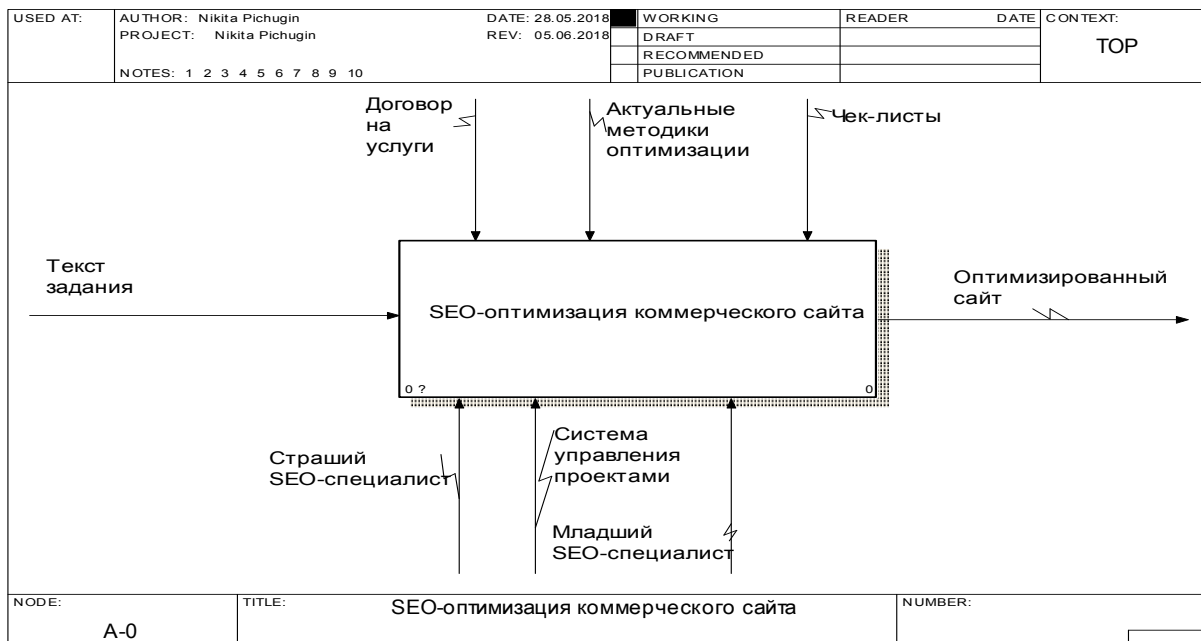


Рисунок А.1 – Контекстная диаграмма seo-оптимизации коммерческого сайта

Декомпозиция процесса seo-оптимизации коммерческого сайта представлена на рисунке А.2.

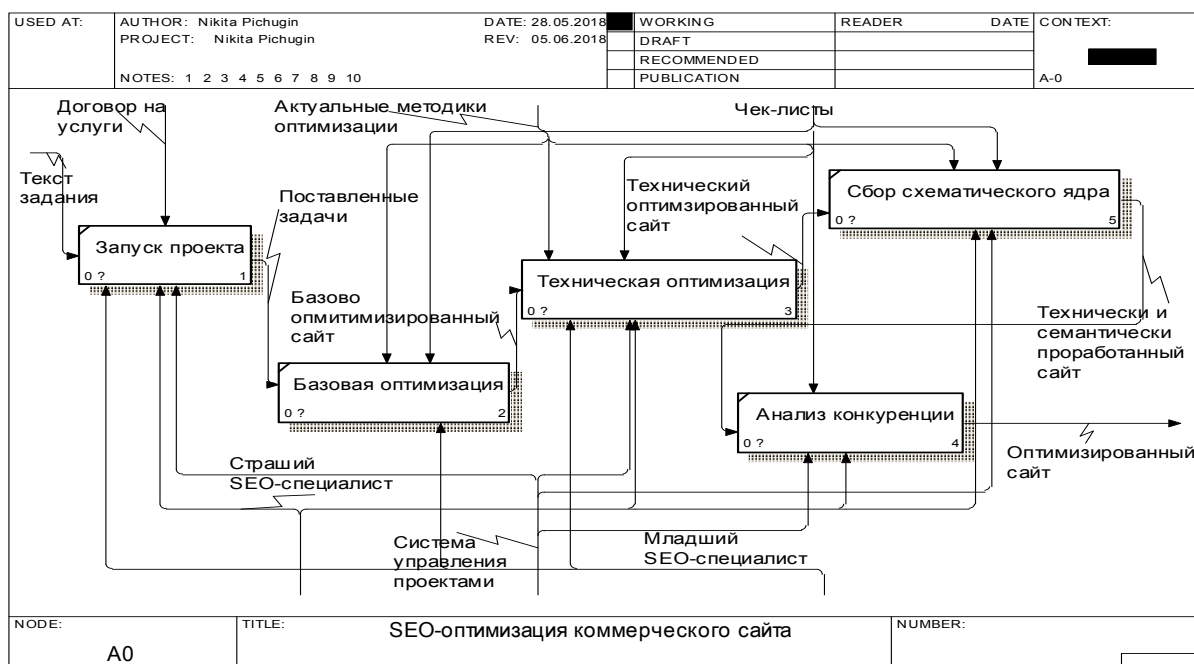


Рисунок А.2 – Декомпозиция контекстной диаграммы

Декомпозиция процесса «Техническая оптимизация» представлена на рисунке А.3.

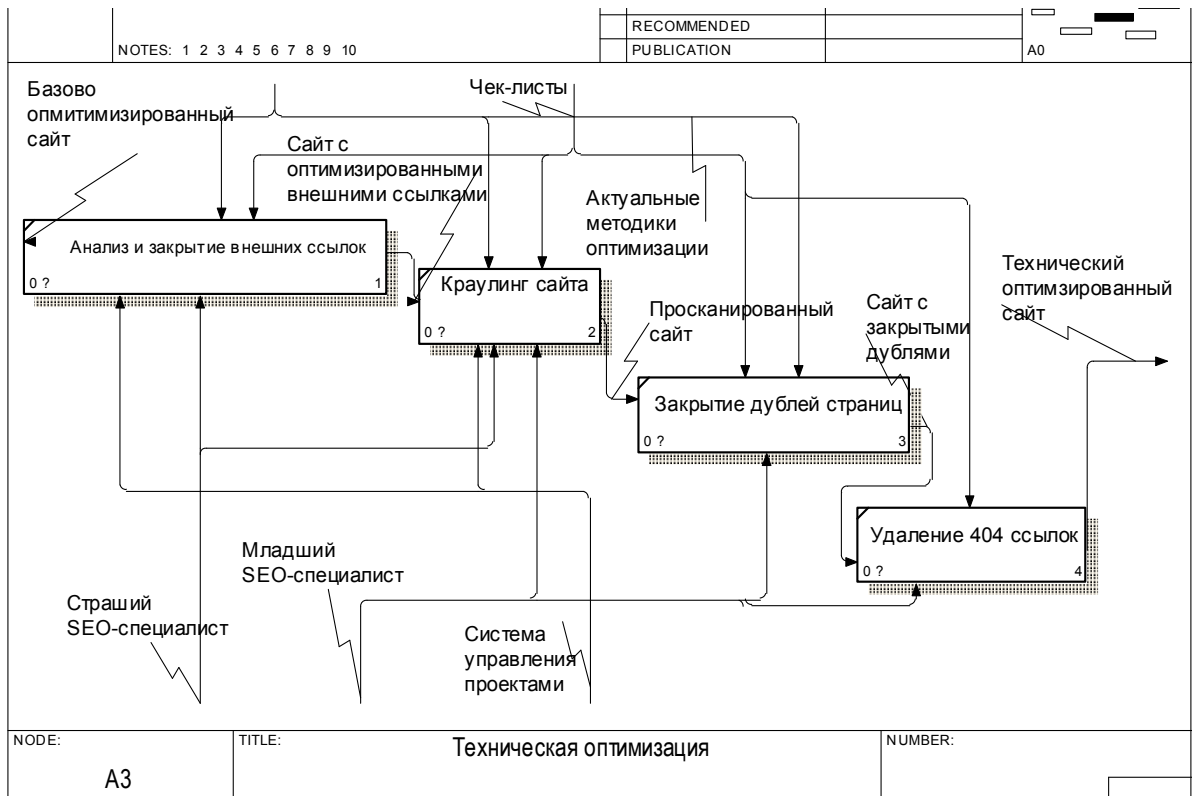


Рисунок А.3 – Декомпозиция процесса «Техническая оптимизация»

ПРИЛОЖЕНИЯ Б

```
package wsa.gui;
import javafx.application.Platform;
import javafx.beans.InvalidationListener;
import javafx.beans.Observable;
import javafx.beans.property.SimpleObjectProperty;
import javafx.collections.FXCollections;
import javafx.collections.ObservableList;
import javafx.collections.ObservableMap;
import javafx.concurrent.Worker;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.geometry.Insets;
import javafx.scene.Scene;
import javafx.scene.chart.PieChart;
import javafx.scene.control.*;
import javafx.scene.layout.Background;
import javafx.scene.layout.BackgroundFill;
import javafx.scene.layout.CornerRadii;
import javafx.scene.paint.Color;
import javafx.scene.text.TextAlignment;
import javafx.scene.web.WebView;
import javafx.stage.Modality;
import javafx.stage.Stage;
import wsa.API.Wrap;
import wsa.Settings;
import wsa.exceptions.EventFrame;
import wsa.session.*;
import java.io.IOException;
import java.net.URI;
import java.nio.file.Path;
import java.util.List;
public class TabFrame extends Tab {

    private Dominio dom;
    private Path path;
    private GestoreDownload gd;
    private DataGate dg = null;
    private Scene rootScene;
    private boolean pausedByUser = false;
    private boolean paused = false;

    private TableView<Wrap<String, Integer>> entersTable = new TableView<>();
    private TableView<Wrap<String, Integer>> exitTable = new TableView<>();
```

```

private TableColumn<Wrap<String, Integer>, String> classEnters = new TableColumn<>("n° ссылок");
private TableColumn<Wrap<String, Integer>, Integer> valueEnters = new TableColumn<>("страницы");
private TableColumn<Wrap<String, Integer>, String> classExit = new TableColumn<>("n° ссылки");
private TableColumn<Wrap<String, Integer>, Integer> valueExit = new TableColumn<>("страницы");

Stage entST = new Stage();
Stage exitST = new Stage();
{
    entST.setTitle("Входящие ссылки");
    exitST.setTitle("Исходящие ссылки");
    entersTable.setColumnResizePolicy(TableView.CONSTRAINED_RESIZE_POLICY);
    exitTable.setColumnResizePolicy(TableView.CONSTRAINED_RESIZE_POLICY);
    classEnters.setCellValueFactory(param -> param.getValue().key);
    valueEnters.setCellValueFactory(param -> param.getValue().val);
    classExit.setCellValueFactory(param -> param.getValue().key);
    valueExit.setCellValueFactory(param -> param.getValue().val);
    entersTable.getColumns().add(classEnters);
    entersTable.getColumns().add(valueEnters);
    exitTable.getColumns().add(classExit);
    exitTable.getColumns().add(valueExit);
    entST.setScene(new Scene(entersTable));
    exitST.setScene(new Scene(exitTable));
}
private ObservableList<Wrap<String, Integer>> entersWrapList = FXCollections.observableArrayList();
private ObservableList<Wrap<String, Integer>> exitWrapList = FXCollections.observableArrayList();
private ContextMenu contestuale = new ContextMenu();
{
    try{
        FXMLLoader loader = new FXMLLoader(getClass().getResource("./tabFrame.fxml"));
        loader.setController(this);
        rootScene = new Scene(loader.load());
        this.setContent(rootScene.getRoot());
    }catch (Exception ex){
        ex.printStackTrace();
        new EventFrame(ex, Platform::exit);
    }
}

public TabFrame(Dominio dominio, List<Seed> s, Path p, Integer m, boolean run){

```

```

this.dom = dominio; // Il dominio di visita
this.path = p; // Il path di visita
this.setText("Preparazione visita in corso");
Thread t = new Thread() -> {
    try {
        this.gd = new GestoreDownload(dominio, s, path, m, this); // Inizializza gestore.
    } catch (Exception e) {
        Platform.runLater()-> new EventFrame(e, null);
        Platform.runLater()-> this.setText("Неудачный визит");
        return;
        //TODO notify the users about IOExceptions.
    }
    if(dominio == null) this.dom = Dominio.getDomainSneaky(gd.getDomain().toString());
    Platform.runLater() -> this.setText(dom != null ? dom.toString() : "Неизвестно");

    this.tableData.setItems(gd.getDataStruct().getDataList());
    tableData.getItems().addListener((InvalidationListener) observable -> {
        long visitati = tableData.getItems().stream().filter(page -
> page.getExc() == null).count();
        long errori = tableData.getItems().stream().filter(page -
> page.getExc() != null).count();
        long indom = tableData.getItems().stream().filter(Page::getPageLink).count();
        Platform.runLater() -> {
            labelVisitati.setText(String.valueOf(visitati));
            labelErrori.setText(String.valueOf(errori));
            labelDominio.setText(String.valueOf(indom));
        });
        if (gd.getPageWithMaxPointers().get() != null && dominio == null) {
            //in caso di recupero visita
            Platform.runLater() -> {
                labelMaxPointers.setText(gd.getPageWithMaxPointers().getValue().ptrNumbe
rs().toString());
                Tooltip tt = new Tooltip(gd.getPageWithMaxPointers().getValue().getURI().t
oString());
                labelMaxPointers.setTooltip(tt);
            };
        }
    });
    gd.getPageWithMaxLinks().addListener(observable -> {
        labelMaxLinks.setText(gd.getPageWithMaxLinks().getValue().linksNumber().toString());
        Tooltip tt = new Tooltip(gd.getPageWithMaxLinks().getValue().getURI().toString());
        labelMaxLinks.setTooltip(tt);
    });

    gd.getPageWithMaxPointers().addListener(observable -> {

```

```

        labelMaxPointers.setText(gd.getPageWithMaxPointers().getValue().ptrNumbers().toStri
ng());
        Tooltip tt = new Tooltip(gd.getPageWithMaxPointers().getValue().getURI().toString())
;
        labelMaxPointers.setToolTipText(tt);
    });

    this.pieEntranti.setData(gd.getEntrantiPieData());
    this.pieUscenti.setData(gd.getUscentiPieData());
    this.entersTable.setItems(entersWrapList);
    this.exitTable.setItems(exitWrapList);

    gd.getEntrantiPieData().addListener((InvalidationListener) observable -> {
        entersWrapList.clear();
        if (gd != null)
            gd.getEntrantiPieData().forEach(data -
> entersWrapList.add(new Wrap<>(data.getName(), (int) data.getPieValue())));
    });
    gd.getUscentiPieData().addListener((InvalidationListener) observable -> {
        exitWrapList.clear();
        if (gd != null)
            gd.getUscentiPieData().forEach(data -
> exitWrapList.add(new Wrap<>(data.getName(), (int) data.getPieValue())));
    });
    if (run) this.run(); /* Avvia la visita se true */
});
t.start();

this.setOnCloseRequest(e -> {
    MainFrame.getMainFrame().getTabs().remove(this);
    new EventFrame(e, Alert.AlertType.WARNING,
        ButtonType.CANCEL,
        this::dispose, ()-> MainFrame.getMainFrame().getTabs().add(this));
});
}
GestoreDownload getWorker(){return this.gd;}
public Worker.State getStato(){
    return gd.getStato();
}
private void cancel(){
    if (gd == null) return;
    if (gd.getStato() != null && gd.getStato() == Worker.State.CANCELLED) return;
    gd.cancel();
    Platform.runLater(() ->{

```

```

        pausa.setDisable(true);
        cancella.setDisable(true);
    });
}
public boolean isPaused(){
    if (!paused && !pausedByUser) return false;
    return true;
}
public void pause(){
    if (gd == null) return;
    if (gd.getStato() != null && gd.getStato() == Worker.State.CANCELLED) return;
    gd.pause();
    paused = true;
    Platform.runLater(()-> {
        pausa.setText("Avvia");
        if(gd.getStato() == Worker.State.SUCCEEDED && !pausedByUser){
            pausa.setDisable(true);}
    });
}
public void start() {
    if (gd == null || gd.getStato() == Worker.State.CANCELLED) return;
    if(gd.getStato() == Worker.State.SUCCEEDED && !pausedByUser) return;
    gd.start();
    paused = false;
    pausedByUser = false;
    pausa.setText("Pausa");
    pausa.setDisable(false);
}
public void enablebuttons(){
    if(gd == null || gd.getStato() == Worker.State.CANCELLED) return;
    System.out.println("heyyyyyyyy");
    gd.start();
    pausa.setText("nayza");
    pausa.setDisable(true);
}
public Dominio getDom(){
    return this.dom;
}
void dispose(){
    try {
        gd.cancel();
    }
    catch (Exception e){return;}
    dom = null;
    gd = null;
}

```



```

}
private void run(){
    System.out.println(gd.getStato());
    if (gd.getStato() != Worker.State.READY) return;
    //TODO unire il metodo con start()
    Thread laucher = ThreadUtilities.CreateThread(() -> this.gd.start());
    laucher.setDaemon(true);
    laucher.setName("Посмотреть профиль");
    laucher.start();
    System.out.println("Запущенный профиль");
}
public void addSeed(Seed seed){
    this.gd.addSeed(seed);
}
public ObservableList<Page> getData(){
    return FXCollections.unmodifiableObservableList(gd.getDataList());
}
public ObservableMap<URI, Page> getDatamap(){
    return FXCollections.unmodifiableObservableMap(gd.getResults());
}
private void loadInNewWindow(String url){
    WebView ww = new WebView();
    Stage stg = new Stage();
    stg.setTitle(url);
    stg.setScene(new Scene(ww));
    ww.getEngine().load(url);
    stg.show();
}

private enum links {puntati, puntanti}
private void showSelectedInNewWindow(links tipo){
    Page selectedElement = tableData.getSelectionModel().getSelectedItem(); // Pagina seleziona
ta.
    TableView<URI> tabella = new TableView<>();
    tabella.setColumnResizePolicy(TableView.CONSTRAINED_RESIZE_POLICY);

    TableColumn<URI, String> uriCol = new TableColumn<>("URI");
    uriCol.setCellValueFactory(param -
> new SimpleObjectProperty<>(param.getValue().toString()));
    TableColumn<URI, Boolean> uriStat = new TableColumn<>("Загружено");
    tabella.getColumns().add(uriCol);
    tabella.getColumns().add(uriStat);
    ObservableList<URI> data = FXCollections.observableArrayList();

```

```

        if (tipo == links.puntanti) {
            data.addAll(selectedElement.getPtr());
            uriStat.setCellValueFactory(param ->
                new SimpleObjectProperty<>(selectedElement.getPtr().contains(param.getValue()
            ))
        );
        selectedElement.getPtrMap().addListener((InvalidationListener) observable -> {
            data.clear();
            data.addAll(selectedElement.getPtr());
        });
    }else{
        data.addAll(selectedElement.getPtd());
        uriStat.setCellValueFactory(param ->
            new SimpleObjectProperty<>(selectedElement.getPtd().contains(param.getValue(
    )))
        );
        selectedElement.getPtdMap().addListener((InvalidationListener) observable -> {
            uriCol.setVisible(false);
            uriCol.setVisible(true);
        });
    }

    tabella.setItems(data);
    Stage st = new Stage();
    st.setScene(new Scene(tabella));
    st.initModality(Modality.APPLICATION_MODAL);
    st.setTitle((tipo == links.puntanti? "Я точно" : "правильно сделал ") + tabella.getItems().si
ze());
    st.show();
}
@Override
public String toString(){
    return this.dom.getURI().toString();
}

private @FXML PieChart pieEntranti;
{
    pieEntranti.setLegendVisible(false);
    pieEntranti.setLabelsVisible(false);
    pieEntranti.setAnimated(false);
    pieEntranti.setOnMouseClicked(e -> {
        if (e.getClickCount() == 2){
            entST.show();
        }
    }
}

```

```

    });
}
private @FXML PieChart pieUscenti;
{
    pieUscenti.setLegendVisible(false);
    pieUscenti.setLabelsVisible(false);
    pieUscenti.setAnimated(false);
    pieUscenti.setOnMouseClicked(e -> {
        if (e.getClickCount() == 2){
            exitST.show();
        }
    });
}
private @FXML ToolBar toolbarTab;
private @FXML TableView<Page> tableData;
private @FXML TableColumn<Page, URI> domColumn;
private @FXML TableColumn<Page, Boolean> followColumn;
private @FXML TableColumn<Page, String> statusColumn;
private @FXML TableColumn<Page, Integer> linksNumColumn;
{
    domColumn.setCellValueFactory(param -
> new SimpleObjectProperty<>(param.getValue().getURI()));
    followColumn.setCellValueFactory(param -
> new SimpleObjectProperty<>(param.getValue().getPageLink()));
    statusColumn.setCellValueFactory(param -
> new SimpleObjectProperty<>(param.getValue().getExc() == null ? "" : param.getValue().getExc().getM
essage()));
    linksNumColumn.setCellValueFactory(param -
> new SimpleObjectProperty<>(param.getValue().linksNumber()));

    // Menu contestuale.
    MenuItem info = new MenuItem("Информация");
    info.setOnAction(e -> {
        if (tableData.getSelectionModel().getSelectedItem() != null)
            new InfoFrame(tableData.getSelectionModel().getSelectedItem()).show();
    });
    MenuItem getPuntanti = new MenuItem("Цель этой страницы.");
    getPuntanti.setOnAction(event -> showSelectedInNewWindow(links.puntanti));
    MenuItem getPuntati = new MenuItem("Укажите на эту страницу.");
    getPuntati.setOnAction(event -> showSelectedInNewWindow(links.puntati));
    contestuale.getItems().addAll(info, getPuntanti, getPuntati);

    tableData.setContextMenu(contestuale);
    tableData.setOnMouseClicked(e ->{
        Page pagina = tableData.getSelectionModel().getSelectedItem();

```

```

        if (e.getClickCount() == 2 && pagina != null){
            if (pagina.getExc() != null){
                new EventFrame(e,
                    Alert.AlertType.WARNING,
                    "Страница была загружен с ошибками.\n" +
                    "Открыть все-равно?",
                    ButtonType.CANCEL,
                    () -
> loadInNewWindow(tableData.getSelectionModel().getSelectedItem().getURI().toString()
                );
            }else if (pagina.linksNumber() == 0 && pagina.getPageLink()){
                new EventFrame(e,
                    Alert.AlertType.WARNING,
                    "На странице 0 ссылок.\n" +
                    "Обычно это означает файл \n" +
                    "мультимедиа.\n" +
                    "Открыть все-равно?",
                    ButtonType.CANCEL,
                    () -
> loadInNewWindow(tableData.getSelectionModel().getSelectedItem().getURI().toString()
                );
            }else{
                loadInNewWindow(tableData.getSelectionModel().getSelectedItem().getURI().toSt
ring());
            }
        }
    });
}

private @FXML Label labelVisitati;
private @FXML Label labelErrori;
private @FXML Label labelDominio;
private @FXML Label labelMaxLinks;
private @FXML Label labelMaxPointers;

private MenuItem pausa;
private MenuItem cancella;

private @FXML MenuButton menuButtonAzioni;
{
    aggiungiSeed.setOnAction(e -> new VisitFrame(this).show());
    pausa = new MenuItem("Пауза");
    pausa.setOnAction(e -> {
        if (gd.getStato() == Worker.State.CANCELLED) return;
        if (!paused){
            System.out.println("Пауза");

```

```

        pausedByUser = true;
        pause();
    }else{
        System.out.println("Продолжить");
        start();
    }
});
cancella = new MenuItem("Четкое сканирование");
cancella.setOnAction(e -> {
    new EventFrame(e, Alert.AlertType.WARNING, "Удаление посещения\n" +
        "Осановить?Или продолжить\n" +
        "для загрузки страниц. Отменить", ButtonType.CANCEL, this::cancel);
});
MenuItem operazioni = new MenuItem("Операции");
operazioni.setOnAction(event -> {
    OperationsFrame of = new OperationsFrame();
    of.show();
    of.select(1);
});
menuButtonAzioni.getItems().addAll(aggiungiSeed, pausa, operazioni, new SeparatorMenuItem(), cancella);
}
{
    tableData.getSelectionModel().selectedItemProperty().addListener((observable, oldValue, newValue) -> {
        domColumn.setVisible(false);
        domColumn.setVisible(true);
    });
}
private void updateNull(TableCell cell){
    cell.setText(null);
    cell.setGraphic(null);
    cell.setBackground(null);
}
public void forceTableRefresh(){
    domColumn.setVisible(false);
    domColumn.setVisible(true);
}
{
    domColumn.setCellFactory(param ->
        new TableCell<Page, URI>(){
            @Override
            public void updateItem(URI item, boolean empty){
                super.updateItem(item, empty);
                if (item == null){

```

```

        updateNull(this);
        return;
    }

    setText(item.toString());
    setBackground(null);

    if (param.getTableView().getSelectionModel().getSelectedItem() == null) re
turn;

    Page data = param.getTableView().getSelectionModel().getSelectedItem();
    // prendo la pagina per comodit?.
    if (data.getPtr().contains(item) && !data.getPtd().contains(item)){
        setBackground(new Background(new BackgroundFill(Settings.config().C
R_PTD.get(), new CornerRadii(0), Insets.EMPTY)));
    }else if(!data.getPtr().contains(item) && data.getPtd().contains(item)){
        setBackground(new Background(new BackgroundFill(Settings.config().C
R_PTR.get(), new CornerRadii(0), Insets.EMPTY)));
    }else if(data.getPtr().contains(item) && data.getPtd().contains(item)){
        this.setBackground(new Background(new BackgroundFill(Settings.conf
ig().CR_PTDandPTR.get(), new CornerRadii(0), Insets.EMPTY)));
    }
}
});
statusColumn.setCellFactory(param ->
    new TableCell<Page,String>(){
        @Override
        public void updateItem(String item, boolean empty) {
            super.updateItem(item,empty);
            if(item == null){
                this.setText(null);
                this.setBackground(null);
            }
            else{
                if(item.equalsIgnoreCase("")){
                    this.setText("Заружено");
                    this.setTextFill(Color.GREEN);
                    this.setTextAlignment(TextAlignment.CENTER);
                }
                else{
                    this.setText(item);
                    this.setTextFill(Color.RED);
                    this.setTextAlignment(TextAlignment.CENTER);
                }
            }
        }
    }
}

```

```

        }
    }
}
);
followColumn.setCellFactory(param ->
    new TableCell<Page, Boolean>(){
        @Override
        public void updateItem(Boolean item, boolean empty){
            super.updateItem(item, empty);
            if (item == null){
                updateNull(this);
                return;
            }
            setText(item ? "DA" : "Net");
            setBackground(null);

            if (item && Settings.config().CR_FOLLOW.getValue()){
                setTextFill(Color.GREEN);
            }else if (!item && Settings.config().CR_FOLLOW.getValue()){
                setTextFill(Color.RED);
            }
        }
    });
}
}
}

```

```
package wsa;
```

```

import javafx.beans.property.ObjectProperty;
import javafx.beans.property.SimpleBooleanProperty;
import javafx.beans.property.SimpleObjectProperty;
import javafx.scene.paint.Color;
import javafx.scene.paint.CycleMethod;
import javafx.scene.paint.LinearGradient;
import javafx.scene.paint.Stop;
import lombok.Cleanup;
import lombok.Getter;
import lombok.NonNull;
import lombok.experimental.Accessors;

```

```

import java.io.BufferedReader;
import java.io.File;
import java.nio.file.Files;
import java.util.ArrayList;
import java.util.HashMap;

```

```

import java.util.List;
import java.util.Random;

public class Settings {
    static class Setting{
        private String ID;          private String Value;
        Setting(String id, String value){
            ID = id;
            Value = value;
        }
    }
    private static File file = new File("./Configure.ini");
    @Accessors(fluent=true) @Getter(lazy = true) private static final Settings config = new Settings(
file);
    public HashMap<String, Setting> settings = new HashMap<>();
    public Integer RES_GRABBER_MILLIS = 1000;
    public final SimpleBooleanProperty CR_FOLLOW = new SimpleBooleanProperty(true);
    public boolean RUN_WITH_LOGO = false;
    public ObjectProperty<Color> CR_PTD = new SimpleObjectProperty<>(Color.VIOLET);

    public ObjectProperty<Color> CR_PTR = new SimpleObjectProperty<>(Color.YELLOW);
    private ObjectProperty<Stop[]> stops = new SimpleObjectProperty<>(new Stop[]){
        new Stop(0, CR_PTR.get()),
        new Stop(1, CR_PTD.get())
    });
    public ObjectProperty<LinearGradient> CR_PTDandPTR = new SimpleObjectProperty<>(
        new LinearGradient(0, 0, 1, 1, true, CycleMethod.NO_CYCLE, stops.get())
    );

    private Settings(File conf){
        if(conf.exists()) {
            try (BufferedReader file = Files.newBufferedReader(conf.toPath())){
                file.lines();
                return;
            } catch (Exception e) {}
        }
    }

    public boolean save(){return false;}

    public <T> T getSetting(String id){
        return null;
    }
    public void addSetting(Setting setting ){

```



```

    }
}

package wsa.API;
import java.util.concurrent.Semaphore;
import java.util.concurrent.atomic.AtomicBoolean;
import java.util.function.Predicate;
public class ConditonSemaphore<T> {
    private volatile Semaphore semaforo = null;
    private Predicate<T> condition = null;
    private T conditionedobject = null;
    private int numeromaxthread = 0;
    private final AtomicBoolean disabled = new AtomicBoolean(false);
    private boolean negated;

    public ConditonSemaphore(Predicate<T> pred , T conditionedObject, int numeromaxthreads, boolean negate){
        semaforo = new Semaphore(0);
        condition = pred;
        conditionedobject = conditionedObject;
        numeromaxthread = numeromaxthreads;
        negated = negate;
    }

    public void acquire() throws InterruptedException {
        if(disabled.get()) return;
        if(negated) {
            if (!condition.test(conditionedobject)) {
                System.out.println("Locked " + conditionedobject.toString());
                semaforo.acquire();
            }
        }
        else {
            if (condition.test(conditionedobject)) {
                System.out.println("Locked " + conditionedobject.toString());
                semaforo.acquire();
            }
        }
    }

    public void release(){
        if(disabled.get()) return;
        if(negated) {
            if (condition.test(conditionedobject)) {
                semaforo.release(numeromaxthread + 1);
                semaforo = new Semaphore(0);
            }
        }
    }
}

```

```

    }
    else {
        if (!condition.test(conditionedobject)) {
            semaforo.release(numeromaxthread + 1);
            semaforo = new Semaphore(0);
        }
    }
}
private void releaseForced(){
    semaforo.release(numeromaxthread + 1);
    semaforo = new Semaphore(0);
}
public void Disable(){
    disabled.set(true);
    this.releaseForced();
}
public void Enable(){
    disabled.set(false);
}
}
}

```

Обозначение	Наименование	Дополнительные сведения	
	Текстовые документы		
1. 11070002.09.03.02.420.ПЗВКР	Пояснительная записка	98 с.	
	Графические документы		
2. 11070002.09.03.02.420.ДМВКР	Демонстрационные материалы (презентация)	12 сл.	
	Демонстрационные материалы (пл. ф. А4)	12 лист *5 экз.	
	Другие документы		
3.	Документы на компакт-диске	1 CD	
11070002.09.03.02.420.ПЗВКР			
Изм.	Лист.	Номер докум.	
Разработала	Степанченко В.С.	Подп.	
Проверила	Щербинина Н.В.	Дата	
Н.контр.	Нестерова Е.В.	Разработка информационной п одсистемы документационного обеспечения управленческой деятельности организации ОО О «ДомРемСтрой» Ведомость ВКР	
Утвердил	Гахов Р.П.		
			Лит. Лист Листов 99 100 НИУ «БелГУ» гр.07001306

Выпускная квалификационная работа выполнена мной совершенно самостоятельно. Все использованные в работе материалы и концепции из опубликованной научной литературы и других источников имеют ссылки на них.

«___» _____ г.

(подпись)

Пичугин Н.И.

(Ф.И.О.)