Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Факультет математики и информатики
Кафедра информационных и управляющих систем
Направление подготовки 09.03.02 – Информационные системы и технологии
Направленность (профиль) образовательной программы Безопасность информационных систем

ДОПУСТИТЬ К ЗАЩИТЕ
И.о. зав. кафедрой
_____ А.В. Бушманов
«26» 06 2020 г.

БАКАЛАВРСКАЯ РАБОТА

на тему: Разработка информационной системы для анализа популярности социальных сетей среди различных возрастных групп с использованием Больших Данных

Исполнитель студент группы 655-об	21.06.2020 (подпись, дата)	Д.О. Шуваев
Руководитель доцент, канд. физ.-мат. наук	21.06.2020 (подпись, дата)	В.В. Еремина
Консультант по части безопасности и экологичности, доцент, канд. техн. наук	19.06.2020 (подпись, дата)	А.Б. Булгаков
Нормоконтроль доцент, канд. техн. наук	25.06.2020 (подпись, дата)	О.В. Жилиндина

Благовещенск 2020

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Факультет математики и информатики
Кафедра информационных и управляющих систем

УТВЕРЖДАЮ
И.о. зав. кафедрой
_____ А.В. Бушманов
«20» 02 2020 г.

З А Д А Н И Е

К выпускной квалификационной работе студента Шуваева Дмитрия Олеговича

1. Тема дипломной работы: Разработка информационной системы для анализа популярности социальных сетей среди различных возрастных групп с использованием Больших Данных.

(утверждена приказом от 30.04.2020 №810-уч)

2. Срок сдачи студентом законченной работы: 26.06.2020 г.

3. Исходные данные к выпускной квалификационной работе: отчет о прохождении преддипломной практики, нормативная документация, специальная литература.

4. Содержание выпускной квалификационной работы (перечень подлежащих разработке вопросов): современные методики и инструменты анализа больших данных, анализ данных социальных сетей, обработка данных, разработка информационной системы, безопасность и экологичность.

5. Консультанты по дипломной работе:
по безопасности и экологичности – Булгаков А.Б., доцент, кандидат технических наук.

7. Дата выдачи задания: 20.02.2020 г.

Руководитель дипломной работы:_____Еремина В.В., доцент, канд. физ.-мат. наук.

Задание принял к исполнению: _____ Д.О. Шуваев

РЕФЕРАТ

Бакалаврская работа содержит 65 с., 28 рисунков, 10 таблиц, 21 источник.

ПРОЕКТИРОВАНИЕ, РАЗРАБОТКА, СИСТЕМА, БОЛЬШИЕ ДАННЫЕ, АНАЛИЗ ДАННЫХ, СОЦИАЛЬНЫЕ СЕТИ, ПАРСИНГ, БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ

Целью бакалаврской работы является разработка информационной системы для получения и обработки данных о пользователях из популярных социальных сетей для их последующего анализа.

В настоящей работе разработана система сбора и анализа данных таких социальных сетей, как ВКонтакте, Twitter и Одноклассники. Система позволяет находить закономерности, на основе которых можно выявить основные характеристики целевой аудитории, ее интересы, увлечения и наиболее обсуждаемые темы.

СОДЕРЖАНИЕ

Введение 7
1 Современные методики и инструменты анализа больших данных 8
1.1 Определение «Большие данные» 8
1.2 «Data Mining» 11
1.2.1 Определение и основные понятия «Data Mining» 11
1.2.2 Задачи анализа данных 12
2 Анализ данных социальных сетей 15
2.1 Исследуемые социальные медиа-платформы 15
2.1.1 «ВКонтакте» 15
2.1.2 «Twitter» 17
2.1.3 «Одноклассники» 19
2.2 Способы сбора данных 20
2.2.1 Использование API 20
2.2.2 Семантический разбор веб-страниц 22
2.2.3 Эмуляция поведения пользователя в браузере 25
3 Обработка данных 27
3.1 Описание собираемых данных 27
3.2 Методы анализа данных о возрасте 30
3.3 Определение наиболее обсуждаемых тем на основании публикаций пользователей 31
4 Разработка информационной системы 34
4.1 Выбор средств разработки 34
4.2 Проектирование схемы базы данных 37
4.3 Разработка экранных форм 42
4.4 Использование proxy и многопоточности 44
4.4.1 Эмуляторы пользовательских инструментов 45
4.4.2 Прокси для эмуляции поведения человека 46

4.4.3 Многопоточность 47
4.5 Применение ИС 47
5 Безопасность и экологичность 51
5.1 Безопасность 52
5.1.1 Вредоносные и неблагоприятные факторы на рабочем месте пользователя ПК 52
5.1.2 Организация автоматизированного рабочего места 53
5.1.3 Освещение 54
5.1.4 Шум 55
5.1.5 Микроклимат 55
5.1.6 Исследование помещения с ПК 56
5.2 Экологичность 58
5.3 Чрезвычайные ситуации 58
5.3.1 Аварийные ситуации 58
5.3.2 Меры пожарной безопасности на рабочих местах 59
5.4 Комплексы физических упражнений для сохранения и укрепления индивидуального здоровья и обеспечения полноценной профессиональной деятельности 60
Заключение 63
Библиографический список 64

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ

АРМ – автоматизированное рабочее место;

БД – база данных;

ООП – объектно-ориентированное программирование;

ПК – персональный компьютер;

ПО – программное обеспечение;

ПЭВМ – персональная электронно-вычислительная машина;

СМИ – средства массовой информации;

СУБД – система управления базой данных;

API – (Application Programming Interface) программный интерфейс приложения;

CSV – (Comma Separated Values) формат файлов .csv;

FK – (Foreign Key) внешний ключ;

HDFS - (Hadoop Distributed File System) — распределенная файловая система;

HTTP – (HyperText Transfer Protocol) протокол передачи гипертекста;

HTML – (HyperText Markup Language) язык гипертекстовой разметки;

IP – (Internet Protocol) протокол интернета;

JSON – (JavaScript Object Notation) формат файлов;

PHP – (Personal Home Page) язык веб-разработки;

SMM – (Social Media Marketing) – маркетинг в социальных медиа;

SQL – (Structured Query Language) структурированный язык запросов;

VK – (Vkontakte) соцсеть ВКонтакте.

ВВЕДЕНИЕ

За прошедшие несколько лет популярность социальных ресурсов существенно выросла благодаря тому, что все больше и больше пользователей стало обмениваться разнообразными видами информации их посредством. По статистическим данным от wearesocial.com и hootsuite.com в 2017 году в мире насчитывалось 2,80 млрд пользователей социальных сетей, а в январе 2020 года это значение увеличилось до 3,80 млрд. Таким образом, аудитория социальных медиа выросла на 35,7% (1 млрд уникальных пользователей) всего за 3 года.

Многие фирмы используют социально-медийные платформы для продвижения своих товаров и услуг, профессионалы регистрируют учетные записи для налаживания связей, а обычные пользователи занимаются обсуждением любых тем. Увеличение числа пользователей также означает увеличение объемов данных, ожидающих обработки и анализа. Тут на помощь приходят технологии больших данных.

Большие данные (англ. big data) — это совокупность инструментов, которые призваны совершать три операции. Во-первых, обрабатывать большие объемы данных. Во-вторых, уметь работать с быстро поступающими данными в очень больших объемах. То есть данных не просто много, но их постоянно становится все больше и больше. В-третьих, они должны уметь работать со структурированными и плохо структурированными данными параллельно в разных аспектах. Большие данные предполагают, что на вход алгоритмы получают поток не всегда структурированной информации, и что из него можно извлечь больше, чем какую-то одну идею.

Целью данной работы является разработка информационной системы для получения и обработки данных о пользователях из популярных социальных сетей для их последующего анализа. Это позволит находить закономерности, на основе которых можно выявить основные психологические и поведенческие характеристики целевой аудитории, ее интересы и профессиональные увлечения.

1 СОВРЕМЕННЫЕ МЕТОДИКИ И ИНСТРУМЕНТЫ АНАЛИЗА БОЛЬШИХ ДАННЫХ

1.1 Определение «Большие данные»

Несмотря на то, как часто термин «Большие данные» используется во всевозможных статьях, публикациях и обсуждениях современных компьютерных технологий, у него отсутствует единое общепризнанное определение. Подавляющая часть используемых определений термина «Большие данные» можно отнести к одной из трех групп: атрибутивные определения, сравнительные и архитектурные.

Атрибутивные определения. Данный тип определений берет свое начало от определения «VVV» (Volume, Velocity, Variety), которое было предложено работниками компании Meta Group в 2001 году с целью указать на равную значимость управления данными по всем трем аспектам. Для того чтобы являться «Большими данными» по атрибутивному определению, данные должны обладать тремя ключевыми свойствами:

- **Объем** (Volume) – Термин «Большие данные» предполагает большой информационный объем (от гигабайт до эксабайт информации), с которым стандартное ПО не может справиться за удовлетворительное время. Также важно понимать, что для решения определенной задачи ценность обычно имеет не весь объем, а лишь незначительная часть. Однако заранее эту составляющую без анализа невозможно определить.
- **Скорость накопления** (Velocity) – Объемы «Больших данных» растут с чрезвычайно-высокими скоростями. Данные регулярно обновляются и требуют постоянной обработки. Как следствие, обновление зачастую приводит к росту объема.
- **Разнообразие** (Variety) – «Большие данные» это данные со значительным разнообразием в своих источниках и своей природе. Они могут иметь

различные форматы и быть структурированы лишь частично или вовсе быть разнородными.

Однако периодически к VVV добавляют и четвертую V – Veracity (достоверность/правдоподобность данных) и даже пятую V – Value (ценность).

Сравнительные определения. Определения данной группы характеризуют «Большие данные» как такие наборы данных, которые из-за своего размера не могут быть сохранены или обработаны при помощи типичных СУБД (SQL-подобные). Сравнительные определения являются субъективными и не определяют «Большие данные» в терминах какой-либо конкретной метрики. С другой стороны, определения данного класса включают эволюционный аспект в описание тех наборов данных, которые могут быть причислены к «Большим данным».

Архитектурные определения. С точки зрения определений данной группы, «Большими данными» считаются такие данные, размеры, скорость обновления или особенности обработки которых не позволяют вести эффективный анализ, используя традиционные реляционные подходы. Архитектурный подход к определению «Больших данных» позволяет выделить две подобласти: «Big Data Science», т.е. изучение и исследование техник по сбору и оценке «Больших данных» и «Big Data Frameworks», т.е. фреймворки (библиотеки) ПО и соответсвующие им алгоритмы для выполнения распределенной обработки и анализа «Больших данных» на вычислительных кластерах.

Так как различия между этими способами определения «Больших данных» заключаются в основном в том, на каком из выделенных аспектов «Больших данных» концентрируется внимание, возможно объединить их в единое целое. Такой подход позволяет составить следующую таблицу для сравнения традиционных данных с «Большими данными».

Таблица 1 – Сравнение «Больших» и традиционных данных

Характеристика	**Традиционные данные**	**«Большие данные»**
Объемы информации	Гигабайты	От гигабайт до эксабайт
Структурированность	Жестко структурированы	Частично структурированы или вовсе неструктурированы
Способ хранения	Централизованный	Полностью распределенный
Типы используемых БД	Реляционные	NoSQL БД, HDFS
Модель обработки	Вертикальная	Горизонтальная
Взаимосвязь данных	Сильная	Слабая
Сложность интеграции	Низкая	Высокая

Как видно из табл. 1, «Большие данные» отличаются от традиционных данных бóльшим объемом, причем нижняя граница того, что может считаться «Большими данными» постоянно повышается. В настоящее время «Большие данные» начинаются как минимум от нескольких терабайт, а как максимум – от нескольких петабайт. Причем эти данные генерируются с очень большой скоростью: например, статистика показывает, что в базы данных Facebook ежедневно загружается 500 терабайт новых данных. Они генерируются в основном из-за загрузок фото и видео на серверы социальной сети, обмена сообщениями, комментариями под постами и так далее.

Помимо своих больших объемов и высокой скорости генерации, «Большие данные» также порождаются большим количеством источников самой разной природы. К тому же, эти данные обычно не имеют строгой внутренней структуры, благодаря чему их интеграция для использования в рамках некоторой единой ИС является задачей не из простых.

В рамках данной работы будет преимущественно использоваться атрибутивный подход к определению «Больших данных». Это связано с тем, что данные, собираемые в социальных сетях не всегда обязаны подходить под определения сравнительного и архитектурного подходов, по крайней мере с точки зрения их конечного потребителя.

1.2 «Data Mining»

В настоящий момент элементы искусственного интеллекта активно применяются в практической деятельности человека. В отличие от традиционных систем искусственного интеллекта, разработка интеллектуального поиска и изучения данных или «добыча данных» (англ. Data Mining), не имитирует естественный интеллект, а увеличивает его возможности мощностью современных вычислительных систем, поисковых систем и хранилищ данных. Часто рядом со словами «Data Mining» попадаются такие понятия как «информационный поиск» (англ. IR – Information Retrieval) и «интеллектуальный анализ данных» (англ. KDD – Knowledge Data Discovery). Их можно считать синонимами Data Mining. Возникновение всех обозначенных определений связано с новым витком в развитии инструментов и методов обработки данных.

1.2.1 Определение и основные понятия «Data Mining»

Знания есть не только у человека, но и в накопленных данных, которые подвергаются анализу. Такие знания часто называют «скрытыми», т. к. они содержатся в гигабайтах и терабайтах информации, которые человек не в состоянии исследовать самостоятельно. В связи с этим существует высокая вероятность пропустить гипотезы, которые могут принести значительную выгоду.

Очевидно, что для обнаружения скрытых знаний необходимо применять специальные методы автоматического анализа, при помощи которых приходится практически добывать знания из «завалов» информации. За этим направлением прочно закрепился термин «добыча данных» или Data Mining. Классическое определение этого термина дал в 1996 г. один из основателей этого направления – Григорий Пятецкий-Шапиро.

Data Mining – исследование и обнаружение вычислительными системами (алгоритмами, средствами искусственною интеллекта) в «сырых» данных скрытых знаний, которые ранее не были известны, нетривиальны, практически полезны и доступны для интерпретации человеком.

Рассмотрим подробнее свойства обнаруживаемых знаний, приведенные в определении:

- **Знания должны быть новые,** ранее неизвестные. Нет никакой пользы от траты усилий на открытие знаний, которые уже известны пользователю. Поэтому интерес вызывают именно те знания, которые ранее не были известны.
- **Знания должны быть нетривиальны.** Результаты анализа должны отражать непредвиденные, неочевидные закономерности в данных. Иначе результаты не оправдывают привлечение методов Data Mining, когда можно получить их более простыми способами.
- **Знания должны быть практически полезны.** Полученные знания должны легко применяться и на новых данных, с достаточной степенью достоверности. Польза заключается в том, что эти знания позволяют получить выгоду при их применении.
- **Знания должны быть доступны для интерпретации человеком.** Выявленные закономерности должны быть логически объяснимы, в ином случае есть вероятность того, что данные были получены случайно. Помимо этого, полученные данные должны представляться в понятном для человека виде.

Инструменты Data Mining могут находить неочевидные закономерности самостоятельно и также самостоятельно строить гипотезы о взаимосвязях. Потому что именно формулировка гипотезы относительно зависимостей считается самой трудной и сложной задачей.

1.2.2 Задачи анализа данных

Классификация. Классификация это самая простая и широко распространенная задача Data Mining. В результате её решения обнаруживаются особенно-

сти, которые характеризуют группы объектов исследуемого набора входных данных – классы; по данным различиям новый объект можно отнести к тому или иному классу. Для решения такой задачи могут применяться такие методы: метод ближайшего соседа; метод К-ближайшего соседа; найвный байес; деревья решений; машинное и глубокое обучение.

Кластеризация. Кластеризация практически продолжает смысл классификации. Это задача более сложная, особенность кластеризации заключается в том, что набор групп (классов), на которые стоит делить объекты изначально не определены. Пример метода решения задачи кластеризации: обучение "без учителя" особого вида глубоких сетей – самостоятельных карт Кохонена.

Ассоциация. В ходе решения задачи поиска некоторых ассоциативных правил определяются закономерности между сопряженными фактами в наборе данных. Отличие ассоциации от 2-х предыдущих задач заключается в том, что поиск закономерностей осуществляется не на основе характеристик объекта, а между некоторыми событиями, которые происходят в один момент. Самый популярный алгоритм решения задач ассоциации – алгоритм Априори.

Последовательность. Последовательность обеспечивает удобный поиск временных закономерностей между транзакциями. Задача последовательности схожа с задачей ассоциации, но ее идеей является определение закономерностей не между одновременно наступающими событиями, а между событиями, которые связаны во времени (т.е. происходящими с определенным интервалом во времени). Другими словами, последовательность устанавливается большой вероятностью цепи связанных в некотором временном отрезке событий. Фактически, ассоциация есть частный случай последовательности с временным отставанием равным нулю. Эту задачу также называют задачей нахождения последовательных шаблонов. Правило последовательности заключается в следующем: после события «А» через какой-то промежуток времени настанет событие «Б». Например, после покупки квартиры большинство в течение двух недель покупают микроволновку, а в течение трех месяцев – кондиционер. Или может существовать последовательная связь между покупкой кровати и простыней для нее.

Последовательности достаточно широко применяется в маркетинге и менеджменте.

Прогнозирование. В результате решения задачи прогнозирования на основе набора уже произошедших и зафиксированных данных дается оценка пропущенным или же будущим значениям численных показателей. При решении таких задач часто применяются методы математической статистики, теории вероятностей, математического моделирования и теории нейронных сетей.

Определение и анализ выбросов (отклонений). Цель решения этой задачи – нахождение и анализ данных, которые отличаются от общего набора данных.

Анализ связей – задача нахождения зависимостей во множестве данных. Например, анализ связи дружбы людей в социальных сетях и определения на основе этих данных лидеров мнений.

Визуализация. В результате визуализации образуется графический образ анализируемых данных. Для решения задачи визуализации используются графические методы, которые могут отобразить существование некоторых закономерностей в данных. Пример методов визуализации – представление данных в двухмерном и трехмерном пространстве.

2 АНАЛИЗ ДАННЫХ СОЦИАЛЬНЫХ СЕТЕЙ

Понятие социальной сети использовалось социологами еще в 20-х годах прошлого века для изучения взаимосвязей между участниками различных сообществ. Психолог и психотерапевт Якоб Морено предложил социограммы, на которых отдельные индивиды представлялись в виде точек, а взаимосвязи между ними – в виде линий. Идею использования аппарата теории графов для изучения взаимоотношений и взаимосвязей между людьми подхватили специалисты в области социологии, психологии, антропологии, политологии, экономики – так сформировалось направление Social Network Analysis (анализ социальных сетей), изучающее структурные свойства социальных взаимосвязей, моделируемых в виде графов и сетей. Важным, но весьма трудоемким этапом такого исследования было построение модели на основе различных данных из печатных источников, дополнительных опросов и анкетирования.

Современные социальные сети существенно изменили постановку вопроса – сегодня у исследователей имеется «бесплатный» ресурс для изучения, а стремительное распространение социальных сервисов и развитие технологий «Больших данных» активизировали интерес к использованию сведений из социальных сетей в различных отраслях. Совместное использование структурных и контентных данных потенциально позволяет применять социальные сети для решения широкого круга бизнес-задач: борьбы с мошенничеством, управления брендом, рекламы товаров и услуг, формирования новых каналов сбыта и др.

В социальных сетях, на форумах, новостных и развлекательных порталах, в блогах содержится много ценного материала, из которого можно добыть информацию о предпочтениях и особенностях людей.

2.1 Исследуемые социальные медиа-платформы

2.1.1 «ВКонтакте»

«ВКонтакте» (международное название: VK) — российская социальная сеть со штаб-квартирой в Санкт-Петербурге. Сайт доступен более чем на 90 языках, но особенно популярен среди русскоязычных пользователей. «ВКонтакте»

(рис. 1) позволяет пользователям отправлять друг другу сообщения, создавать собственные страницы и сообщества, обмениваться изображениями, тегами, аудио- и видеозаписями, играть в браузерные игры.

Рисунок 1 – Интерфейс «ВКонтакте»

Запущенный 10 октября 2006 года, ресурс изначально позиционировал себя в качестве социальной сети студентов и выпускников российских вузов, позднее стал называть себя «*современным, быстрым и эстетичным способом общения в сети*». По данным SimilarWeb на сентябрь 2019 года, сайт «ВКонтакте» занимал 12 место по популярности в мире.

Основная статистика сайта на сегодняшний день:

- 97 миллионов активных пользователей в месяц;
- 6,5 миллиардов сообщений в сутки;
- 1 миллиард отметок «Нравится» в сутки;
- 9 миллиардов просмотров записей в сутки;
- 550 миллионов просмотров видео в сутки;
- 1-е место по длительности пребывания мобильной аудитории;
- 77 % пользователей от всей мобильной аудитории Рунета.

Функциональность «ВКонтакте».

Пользователям «ВКонтакте» доступен характерный для многих социальных сетей набор возможностей: создание профиля с информацией о себе, производство и распространение контента, гибкое управление настройками доступа, взаимодействие с другими пользователями приватно (через личные сообщения) и публично (с помощью записей на «стене», а также через механизм групп и встреч), отслеживание через ленту новостей активности друзей и сообществ.

Кроме возможности писать сообщения, пользователь может оставлять комментарии под уже опубликованным контентом. К своим сообщениям можно «прикреплять» фотографии, аудиозаписи и видеозаписи (в том числе и полнометражные фильмы), документы и опросы.

Возможность загружать на сайт собственные записи и использовать файлы, загруженные другими пользователями, делает «ВКонтакте» одним из крупнейших медиаархивов Рунета. Из всех имеющихся на сайте файлов пользователь может создавать в своём профиле личную коллекцию записей, при желании группируя их в отдельные альбомы. При этом введено предельное количество для одного альбома — 10 000 изображений.

«ВКонтакте» предлагает сторонним ресурсам использовать специально разработанные инструменты для глубокой интеграции с социальной сетью — виджеты. Эти решения позволяют встраивать в сайты систему комментариев для пользователей, сообщества, систему опросов, а также возможность легко поделиться ссылкой на материал с другими пользователями и авторизоваться на сайте.

2.1.2 «Twitter»

«Твиттер» (рис. 2) — социальная сеть, в которой пользователи публикуют записи и взаимодействуют друг с другом посредством сообщений, называемых «твитами». Английский глагол «to tweet» переводится как «щебетать, болтать», а концепция «Твиттера» сводится к тому, что каждый твит не должен превышать 280 символов, а иначе это уже не чириканье, а ток-шоу. Публикация коротких

заметок в формате блога получила название «микроблогинг». Зарегистрированные пользователи могут размещать сообщения, ставить отметку «Нравится», и делать так называемые «ретвиты». Ретвит – это вторичная публикация сообщения, размещенного другим пользователем в Твиттере, со ссылкой на источник.

Рисунок 2 – Интерфейс «Twitter»

Все твиты, публикуемые конкретным пользователем, могут увидеть его подписчики – люди, которые связаны с ним в этой соц. сети. Подписчики (или «читатели») могут быть как односторонними, так и взаимными.

Основной способ представления твитов — это представление в виде ленты. Пользователь, войдя на сайт, видит сообщения от всех читаемых им людей, отсортированные в порядке удаления времени от настоящего момента. Дальше он может перейти на страницу конкретного человека и прочитать только его сообщения, но обычно информация воспринимается именно в форме потока, уходящего назад, до момента регистрации читающего пользователя на сайте. Так он

узнаёт, что нового произошло в жизни его знакомых, о чём рассказывают интересные ему аккаунты и какие события обсуждаются в мире. При помощи ретвитов информация действительно распространяется очень быстро.

2.1.3 «Одноклассники»

Одноклассники (рис. 3) — одна из крупнейших социальных сетей в России и странах ближнего зарубежья, входит в холдинг Mail.Ru Group. Сайт был создан в 2006 году и в настоящее время переведен на 16 языков, включая русский. 43 миллиона жителей России ежемесячно используют Одноклассники: общаются с друзьями и близкими с помощью сообщений, голосовых и видеозвонков, открыток и стикеров.

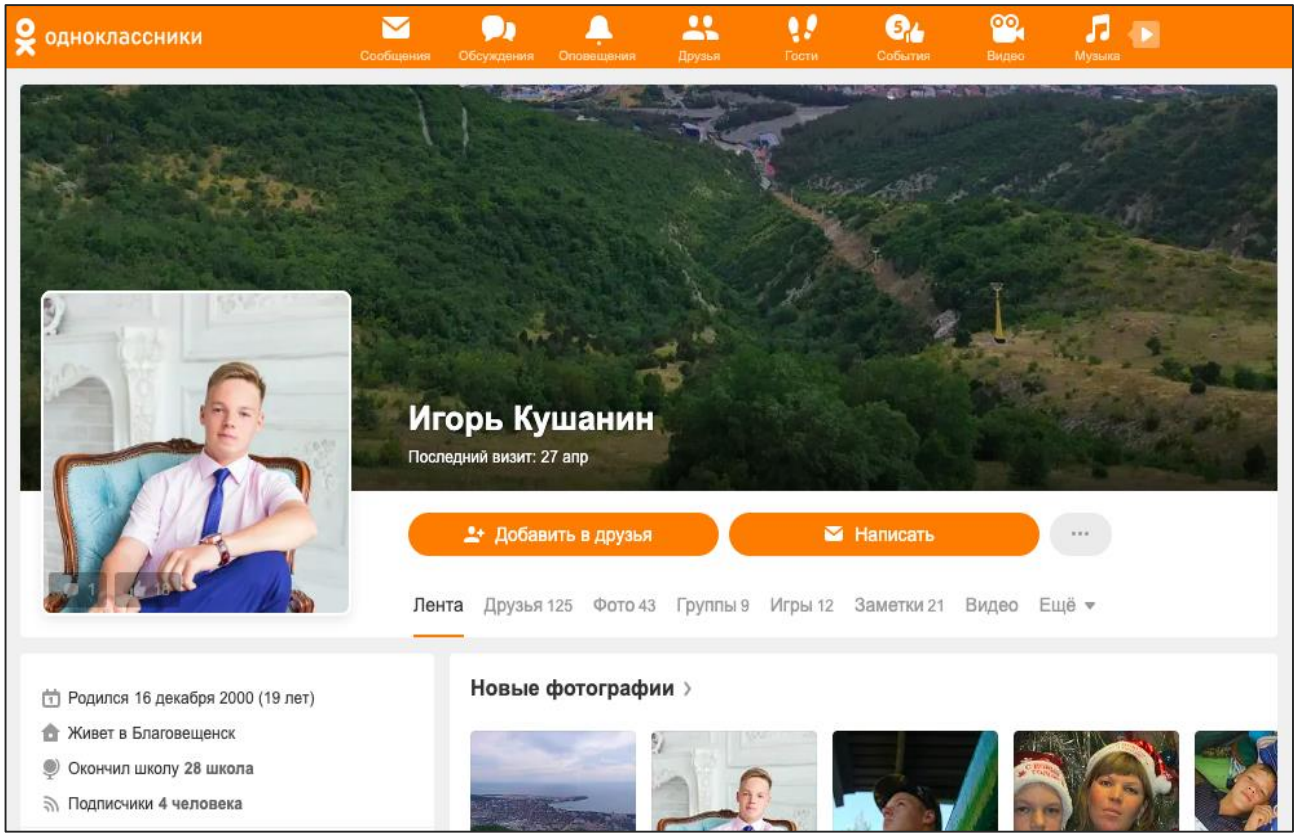

Рисунок 3 – Интерфейс Одноклассников

Одноклассники — это технологичная контентная и сервисная платформа: в социальной сети можно смотреть трансляции в качестве 4К, слушать актуальную музыку, покупать товары и услуги и осуществлять денежные переводы в 18 стран мира.

Одноклассники — лидер рынка онлайн-видео и первая социальная сеть в России по просмотрам видеоконтента: в пике в сутки видео в ОК набирают более 1 млрд просмотров.

Социальная сеть Одноклассники располагает более чем 7 тыс. серверами, работающими в Облаке собственной разработки, распределенным по 5 хостингам в Москве.

На платформе Одноклассников доступны видео, музыка, игры и онлайн-сервисы, а также уникальные способы выражения эмоций: «подарочки», «оценки», кнопка «класс» и другие эмоции на посты. На площадке зарегистрировано более 14 млн групп, включая группы брендов, звезд и СМИ. С 2015 года ОК активно развивают присутствие ТВ-каналов и изданий, имея уникальные возможности видео- и радиотрансляций в режиме реального времени, рекламного продвижения, статистики. К настоящему моменту в Одноклассниках открыли свои сообщества более 600 медиа.

2.2 Способы сбора данных

2.2.1 Использование API

API (от англ. application programming interface - программный интерфейс приложения) – это интерфейс взаимодействия между сервером приложения и сторонними программами и сайтами. API содержит в себе набор готовых функций, процедур и структур, предоставляемых приложением (библиотекой, сервисом) или операционной системой для использования во внешних программных продуктах. На сегодняшний день практически все социальные медиа предоставляют API сторонним разработчикам. Это позволяет фактически любому человеку получить доступ к информации, размещенной на страницах пользователей социальных сетей.

Таким образом, если мы хотим получить какую-либо информацию, например, получить список подписчиков конкретного пользователя, необходимо сделать следующий HTTP-запрос (рис. 4):

Рисунок 4 – API-запрос к серверу «ВКонтакте»

Здесь видно протокол, по которому осуществляется соединение (HTTPS), адрес API-сервиса, название метода API и параметры запроса. В данном примере у запроса есть всего 2 параметра: **user_id** и **count**. Первый передает серверу идентификатор пользователя, данные о подписчиках которого требуется вернуть. Второй параметр указывает, информацию о скольки пользователях мы хотим получить.

Выполнив этот запрос, сервер вернет данные в формате JSON (рис. 5). JSON (JavaScript Object Notation) - простой формат обмена данными, удобный для чтения и написания как человеком, так и компьютером. Он основан на подмножестве языка программирования JavaScript, определенного в стандарте ECMA-262. JSON - текстовый формат, полностью независимый от языка реализации, но он использует соглашения, знакомые программистам С-подобных языков, таких как C, C++, C#, Java, JavaScript, Python и многих других. Эти свойства делают JSON идеальным языком обмена данными.


```
"response": {
  "count": 40160,
  "items": [{
    "id": 22065950,
    "first_name": "Алексей",
    "last_name": "Сысоев"
  }, {
    "id": 594640648,
    "first_name": "Никита",
    "last_name": "Саврасов"
  }, {
    "id": 597543506,
    "first_name": "Ирина",
    "last_name": "Гриневич"
  }, {
    "id": 401499250,
    "first_name": "Фёдор",
    "last_name": "Иванов",
    "deactivated": "banned"
  }]
}
```


Рисунок 5 – Ответ сервера

Несмотря на все удобство использования API, существуют некоторые ограничения – соцсеть не может отдавать все данные, которые видны в пользовательском интерфейсе.

Появлению ограничений способствовали следующие два фактора:

1. Социальные сети стараются сохранять приватность своих пользователей;

2. Некоторые API-методы слишком сильно нагружают серверную часть приложения. Например, если вызвать метод **users.search**, который возвращает данные о пользователях по введенному поисковому запросу, можно получить ответ в 500 тыс. найденных профилей. Обработка такого объема данных использует значительную часть ресурсов сервера. Поэтому для данного запроса установлено ограничение, из-за которого в ответе сервера возвращаются только первые 1000 результатов запроса.

Для того, чтобы избежать проблем, связанных с ограничениями API, можно собирать данные с помощью парсинга веб-сайтов.

2.2.2 Семантический разбор веб-страниц

В широком понимании парсинг – это сбор данных и синтаксический анализ информации, размещенной на веб-страницах в Интернете. Общий принцип его работы можно объянить следующим образом: некоторый автоматизированный программный код выполняет GET-запросы на требуемый интернет-ресурс и получая ответ, проходит по всему HTML-документу, ищет данные и преобразует их в необходимый формат, будь то JSON или CSV.

К категории полезных данных могут относиться:

- текстовый контент;
- изображения;
- видео;
- каталог товаров;
- открытые контактные данные — номера телефонов, e-mail и т.д.

Существует масса решений для парсинга веб-сайтов. Одно из них – библиотека BeatifulSoup. Она будет использована для получения некоторых данных в настоящей работе.

BeautifulSoup - это парсер для синтаксического разбора файлов HTML/XML, написанный на языке программирования Python, который может преобразовать даже неправильную разметку в дерево синтаксического разбора. Он поддерживает простые и естественные способы навигации, поиска и модификации дерева синтаксического разбора. В большинстве случаев он поможет программисту сэкономить часы и дни работы.

Пример работы BeautifulSoup:

Допустим, был выполнен GET-запрос и получен следующий HTML-документ (рис. 6):

```
<!DOCTYPE html>
<html>
  <head>
    <title>Header</title>
    <meta charset="utf-8">
  </head>
  <body>
    <h2>Operating systems</h2>
    <div id="description" style="width:150px">
      Debian is a Unix-like computer operating system that is composed entirely of free software.
    </div>
    <p>
      FreeBSD is an advanced computer operating system used to
      power modern servers, desktops, and embedded platforms.
    </p>
  </body>
</html>
```

Рисунок 6 – HTML-документ

Если нужно получить значение, к примеру, контейнера ** ** с идентификатором **id=”description”**, можно выполнить следующий программный код на Python (рис. 7):

```
from bs4 import BeautifulSoup

with open("example.html", "r") as f:
    contents = f.read()
    soup = BeautifulSoup(contents, 'lxml')
    print(soup.find("div", id="description").text.strip())
```

Рисунок 7 – Получение значения контейнера

Результатом выполнения будет необходимый нам текст: **«Debian is a Unix-like computer operating system that is composed entirely of free software.»**.

Наряду с библиотеками для парсинга HTML-содержимого страниц используются т.н. **регулярные выражения.** Регулярное выражение — это строка, задающая шаблон поиска подстрок в тексте. Идея регулярных выражений заключается в следующем: описывается какой-то шаблон (“регулярное выражение”, “regex”), а затем совершается поиск в текстовой строке, чтобы получить подходящие результаты. Некоторые из этих шаблонов могут выглядеть довольно сложно, так как они содержат не только содержимое, которое мы хотим найти, но и специальные символы, которые меняют восприятие этого шаблона. Регулярные выражения используются каждый раз, когда нужно обработать строковую информацию, поэтому важно применять их при работе с содержимым веб-страниц.

Например, если на странице требуется найти все почтовые ящики, то можно искать совпадения со следующим регулярным выражением:

\S+@\S+\.\S+

Если применить этот шаблон на страницу, где присутствует некоторый текст, то результатом будут только валидные e-mail адреса (рис. 8):

```
Lorem ipsum dolor sit shdm@mail.ru amet, consectetur adipiscing elit.
Ut ex mi, shdm2012 @ gmail.com imperdiet id libero vitae, hendrerit
hendreritshdm2012@gmail.com justo. Fusce condimentum mi non enim
accumsan, sit amet elementum augue egestas. dmitry.shu@yandex.ru Morbi
leo mi, iaculis quis hello@mail .ru rutrum sit amet, molestie non
lacus. Pellentesque vitae rutrum odio. Aliquam eu odio quis dolor
feugiat3764351@mail.ru sodales sed vel dui.
```

Рисунок 8 – Результаты поиска шаблона

Однако, использовать только регулярные выражения при разработке – довольно сложная задача, так как может потребоваться очень узко специализировать каждое регулярное выражение, а помимо этого, громоздкие регулярные выражения могут создавать значительную дополнительную нагрузку операционной системы.

Таким образом, недостатками парсинга могут быть как долгое время разработки, так и долгое время работы написанных программ. Кроме того, не все данные могут быть доступны без какой-либо аутентификации пользователя. Сейчас во многих социальных сетях есть функция скрытия своих данных от просмотра незарегистрированными пользователями. Поэтому, чтобы получить полный доступ к данным, можно использовать средства, позволяющие эмулировать поведение реального пользователя в браузере.

2.2.3 Эмуляция поведения пользователя в браузере

Одним из самых популярных средств в данной области является Selenium WebDriver. Selenium WebDriver – это программная библиотека для управления браузерами. WebDriver представляет собой драйверы для различных браузеров и клиентские библиотеки на разных языках программирования, предназначенные для управления этими драйверами. По сути своей использование такого веб-драйвера сводится к созданию бота, выполняющего всю ручную работу с браузером автоматизированно. Библиотеки WebDriver доступны на языках Java, .Net (C#), Python, Ruby, JavaScript, драйверы реализованы для браузеров Firefox, InternetExplorer, Safari, Andriod, iOS (а также Chrome и Opera).

Selenium WebDriver — это в первую очередь набор библиотек для различных языков программирования. Эти библиотеки используются для отправки HTTP запросов драйверу (отсюда и название WebDriver), с помощью протокола JsonWireProtocol, в которых указано действие, которое должен совершить браузер в рамках текущей сессии. Примерами таких команд могут быть команды нахождения элементов по локатору, переход по ссылкам, парсинг текста страницы/элемента, нажатие кнопок или переход по ссылкам на странице веб-

сайта. Существуют как официальные привязки библиотеки к популярным языкам программирования, так и любительские. К примеру, библиотека для поддержки языка PHP не является официальной и разрабатывается Facebook.

В данной работе, в связи с ограничением платформы Twitter на максимальное получение 1800 твитов в 15 минут, будет использоваться Selenium WebDriver в связке с BeautifulSoup, что позволит получать до 16000 твитов в интервале 15 минут. Данный эффект будет достигаться за счет распараллеливания процессов и использования прокси-серверов.

Подводя итоги текущей главы, можно подчеркнуть, что социальные сети служат новым полезным источником данных о пользователях любой социальной группы. Использовать этот источник не так просто, и возникающие на этом пути проблемы требуют специализированных технологий и инструментов.

3 ОБРАБОТКА ДАННЫХ

Назначение разрабатываемой программы заключается в отображении данных о возрасте пользователей для составления портрета типичного пользователя и определения целевой аудитории той или иной социальной сети. Также предусматривается сбор записей, недавно опубликованных пользователями для определения актуальных и обсуждаемых тем на данный момент.

Имея весьма большие наборы слабоструктурированных данных, объемы которых ограничиваются лишь мощностью имеющейся вычислительной техники, задача данной работы сводится к следующим подзадачам:

1. Сбор данных из 3-х вышеупомянутых социальных медиа-платформ;
2. Приведение разнородной информации к общему виду;
3. Выполнение предобработки текстовых данных;
4. Применение методов анализа данных;
5. Вывод результатов.

3.1 Описание собираемых данных

У каждой соцсети своя структура хранения и отображениях данных. Ниже представлены собираемые данные.

«ВКонтакте»

Социальная сеть ВКонтакте предоставляет API для получения практически всех общедоступных данных (которые может получить любой зарегистрированный пользователь через веб-версию vk.com), но и вводит ограничительные барьеры, причины существования которых были разобраны в главе 2. Также, для приложений введены различные виды токенов доступа, что разграничивает способы получения информации (рис. 9).

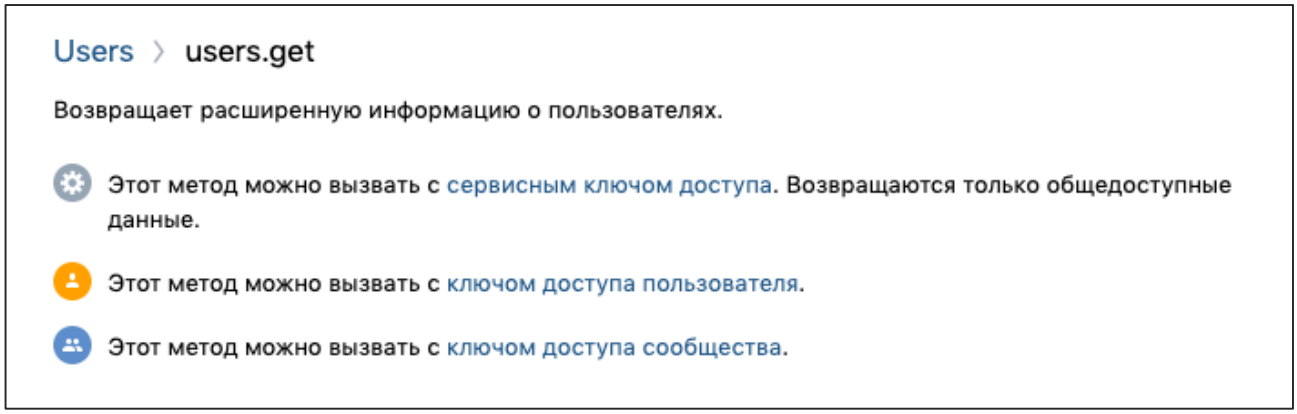

Рисунок 9 – Токены доступа

Итак, из ВКонтакте были собраны следующие поля и помещены в объект User:

- ***vk_url:*** поле *domain* объекта Пользователь;
- ***vk_id:*** уникальный идентификатор объекта Пользователь;
- ***first_name***: имя пользователя;
- ***last_name:*** фамилия пользователя;
- ***sex:*** пол;
- ***birth_date:*** дата рождения;
- ***hidden:*** скрыл ли пользователь свой профиль;
- ***deactivated:*** является ли профиль заблокированным или удаленным;
- ***counter_wall_posts:*** число записей на стене данного пользователя, в том числе и репостов записей;
- ***is_wall_open:*** открыта ли страница пользователя для публикации на ней записей другими пользователями;
- ***wall_post_text:*** текст записи;
- ***wall_post_date:*** дата размещения записи;
- ***wall_post_type:*** тип записи (текст, фото, видео);
- ***wall_post_id:*** уникальный идентификатор записи.

«Twitter»

Из-за ограничений Twitter API было решено написать модуль-парсер, который собирает следующую информацию о пользователях и публикуемых ими твитах:

- ***tweet_id:*** уникальный идентификатор твита;
- ***tweet_url:*** URL-адрес твита;
- ***tweet_text:*** текст твита;
- ***tweet_timestamp:*** отметка времени, когда твит был размещен;
- ***tweet_replies:*** количество ответов на данный твит;
- ***tweet_retweets:*** количество ретвитов;
- ***screen_name:*** имя пользователя, автора твита;
- ***user_id:*** уникальный идентификатор пользователя;
- ***user_verified:*** отметка, является ли пользователя подтвержденным;
- ***user_birth_date:*** дата рождения пользователя.

«Одноклассники»

Данные этой соцсети будут собираться при помощи API и механизма REST-запросов. Список полей следующий:

- ***accessible:*** открыта ли страница пользователя для неавторизованных посетителей;
- ***age:*** возраст пользователя;
- ***birthday:*** дата рождения;
- ***blocked:*** заблокирована ли таблица пользователя;
- ***first_name:*** имя пользователя;
- ***last_name:*** фамилия пользователя;
- ***gender:*** пол;
- ***uid:*** уникальный идентификатор пользователя;
- ***media_topic_id:*** уникальный идентификатор записи;
- ***media_topic_text:*** текст записи.

Собрав все необходимые данные, можно приступать к предобработке и анализу данных.

3.2 Методы анализа данных о возрасте

Информация о возрасте поступает в следующих форматах:

1) «ВКонтакте»
   а) «ДД.ММ.ГГГГ». Например: 05.08.1998;
   б) «ДД.ММ» Например: 05.08. Такая информация нам не интересна, также как и отсутствие информации о дате рождения впринципе, поэтому эти данные отсеиваются при предобработке;
2) «Twitter»
   а) “Дата рождения: 19 августа 1946 г.”. Из данного результата при помощи регулярных выражений удаляется лишняя информация, название месяца переводится в его порядковый номер и в конечном счете дата принимает формат «ДД.ММ.ГГГГ»;
   б) Информация о пользователях, указавших информацию о своей дате рождения не полностью, или не указавших информацию впринципе, удаляются из результирующей выборки;
3) «Одноклассники»
   а) Формат возвращаемой даты рождения: «ГГГГ-ММ-ДД». На этапе предобработки легко преобразуется в «ДД.ММ.ГГГГ»;
   б) Информация о профилях со скрытой информацией удаляется из результирующей выборки.

На основе обработанных данных будут вычисляться следующие значения:

- Среднее арифметическое выборки;

$$\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n}$$

- Медиана. Медиана выборки – это альтернатива средней арифметической, т.к. она устойчива к аномальным отклонениям (выбросам). Математиче-

ским свойством медианы является то, что сумма абсолютных (по модулю) отклонений от медианного значения дает минимально возможное значение, если сравнивать с отклонениями от любой другой величины;

$$M_e = X_{Me} + i_{Me} \cdot \frac{\frac{\sum f}{2} - S_{Me-1}}{f_{Me}}$$

Также будет строиться гистограмма распределения возрастов, который позволит наглядно увидеть, количество пользователей каких возрастов преобладает на той или иной социальной медиа-платформе (рис. 10).

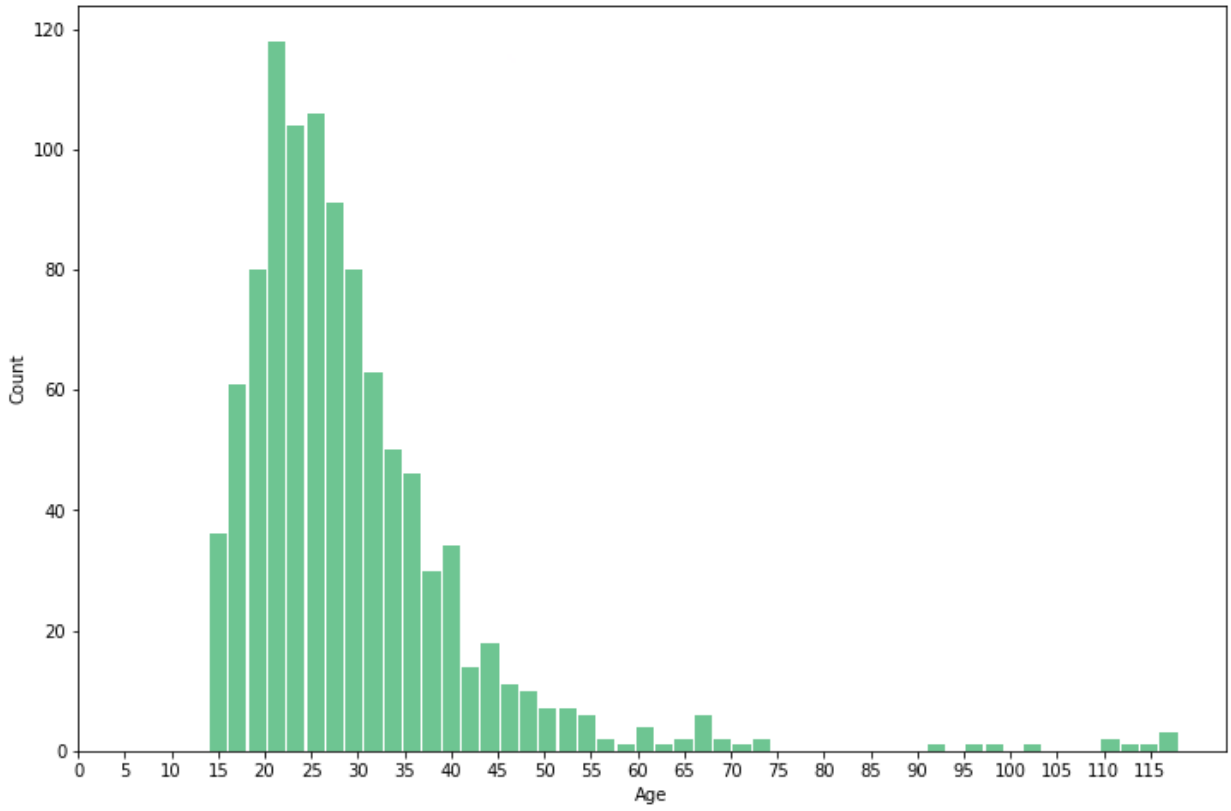

Рисунок 10 – Пример гистограммы распределения возрастов

3.3 Определение наиболее обсуждаемых тем на основании публикаций пользователей

Определение актуальности некоторых тем осуществляется на основе содержания публикаций пользователей. Одним из основополагающих факторов здесь является использование хэштегов.

Хэштег — это метка, которая используется для распределения сообщений по темам в социальных сетях и блогах. Помечая свои сообщения хэштегом, пользователи сети маркируют их и дают возможность другим пользователям найти тематическую информацию с помощью поиска.

Такой способ маркировки пришел из Твиттера и быстро распространился по другим социальным сетям. Если знать, как пользоваться хэштегами, они помогут структурировать информацию по конкретному запросу и потенциально увеличить посещаемость ваших страниц.

Свойства хэштега:

- Выделение главной мысли сообщения, использую ключевые слова;
- Группировка информации по темам;
- Обеспечение быстрого поиска по интересующим темам.

Внешне хэштеги выглядят как слово или несколько слов (рис. 11), перед которыми стоит символ # (пример: #хэштег, #коронавирус, #музыка). Решетка превращает слово или фразу в ссылку. На большинстве платформ, если кликнуть по этой ссылке, можно просмотреть все сообщения, помеченные данным хэштегом.

Рисунок 11 – Пример использования хэштегов

Таким образом, после процесса сбора и предобработки текстов постов, с помощью регулярных выражений извлекается набор указанных хештегов и добавляется в коллекцию. Самые популярные хэштеги и будут отражать события или обсуждаемые в соцсети темы.

4 РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ

4.1 Выбор средств разработки

Для разработки информационной системы должны быть учтены современные тенденции и технологии при проектировании программного обеспечения.

Первым делом нужно определиться с языком программирования для разработки информационной системы сбора и анализа данных. Для этого были выделены следующие критерии оценки:

- кроссплатформенный язык, позволяющий функционировать разработанному приложению на различных операционных системах (Windows, MacOS, Linux);
- легкость и быстрота освоения;
- отсутствие громоздких конструкций;
- наличие качественной документации;
- объектно-ориентированный;
- позволяет легко создавать парсеры для сбора информации с внешних сайтов;
- надежность.

В качестве рассматриваемых языков программирования были выбраны самые популярные на сегодняшний день: C#, TypeScript, PHP, Go, Python.

Представим критерии и список языков в таблице 2.

Таблица 2 – Сравнение языков программирования

Критерий	**C#**	**TypeScript**	**PHP**	**Go**	**Python**
Кроссплатформенность	+	+	+	+	+
Быстрота освоения	+/-	-	-	+	+
Отсутствие громоздких конструкций	+	+	-	+	+
Документация	+	+/-	+	-	+

Продолжение таблицы 2

Объектно-ориентированный	+	+	+	+/-	+
Легкое создание парсеров	+/-	+	-	-	+
Надежность	+	+	+	+	+

Отдельно стоит выделить, что TypeScript и PHP – языки, в основном предназначенные для проектирования web-приложений, а язык Go нашел свое применение при написании системного ПО для кластерных серверов и мэйнфреймов. Язык Python имеет самую обширную и хорошо проработанную документацию, а также стандартная библиотека включает большой объём полезных функций.

Таким образом, в качестве языка для разработки был выбран Python.

Python — высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. Язык создан Гвидо ван Россумом в 1989 году и с тех пор непрерывно совершенствуется.

Преимущества Python:

– язык довольно прост в изучении, особенно на начальном этапе;

– особенности синтаксиса стимулируют писать удобочитаемый код;

– имеются средства оперативного прототипирования и динамической семантики;

– множество полезных модулей и расширений можно достаточно просто использовать в своих разработках благодаря стандартизированному механизму импорта и программным интерфейсам;

– почти всё в Python является объектами в смысле объектно-ориентированного программирования, но при этом ООП не навязывается;

Python портирован и работает почти на всех известных платформах — от КПК до мейнфреймов. Существуют порты под Microsoft Windows, практически

все варианты UNIX(включая FreeBSD и Linux), Mac OS и Mac OS X, iPhone OS 2.0 и выше, Symbian, Android и т.д.

Python поддерживает динамическую типизацию, т.е. тип переменной определяется только при исполнении програмым. Поэтому вместо «присваивания значения переменной» лучше говорить о «связывании значения с некоторым именем». В Python имеются встроенные типы: bool, string, Unicode-string, int, float, complex и некоторые другие. Из коллекций в Python встроены: list, set (неизменяемый list), dict (словарь ключ-значение) и другие. Все значения являются объектами, в том числе функции, модули и классы.

Богатая стандартная библиотека является одной из привлекательных сторон Python. Здесь имеются средства для работы со многими сетевыми протоколами и форматами Интернета, например, модули для написания HTTP-запросов, для разбора и создания потоковых сообщений, для работы с XML и т. п. Набор модулей для работы с операционной системой позволяет писать кросс-платформенные приложения. Существуют модули для работы с регулярными выражениями, текстовыми кодировками, хэш-функциями, сериализаторами данных. Все это пригодится при разработке программного продукта.

Исходя из вышеуказанных преимуществ, язык Python подходит для решения поставленной задачи, т.е. разработки парсеров и анализа больших объемов данных.

В качестве инструмента для разработки экранных форм был выбран Qt Designer. Qt Designer – инструмент для проектирования и создания графических пользовательских интерфейсов (GUI) из компонентов Qt. Принцип работы работы в среде реализуется по принципу «What you see is what you get», WYSIWYG, «что вы видите, то и получаете» (рис. 12).

Виджеты и формы, созданные с помощью Qt Designer, интегрированы с управляющим кодом, использующий механизм сигналов и слотов Qt, который позволяет легко установить обработчики событий для элементов графики. Все свойства в Qt Designer изменяются динамически внутри программного кода.

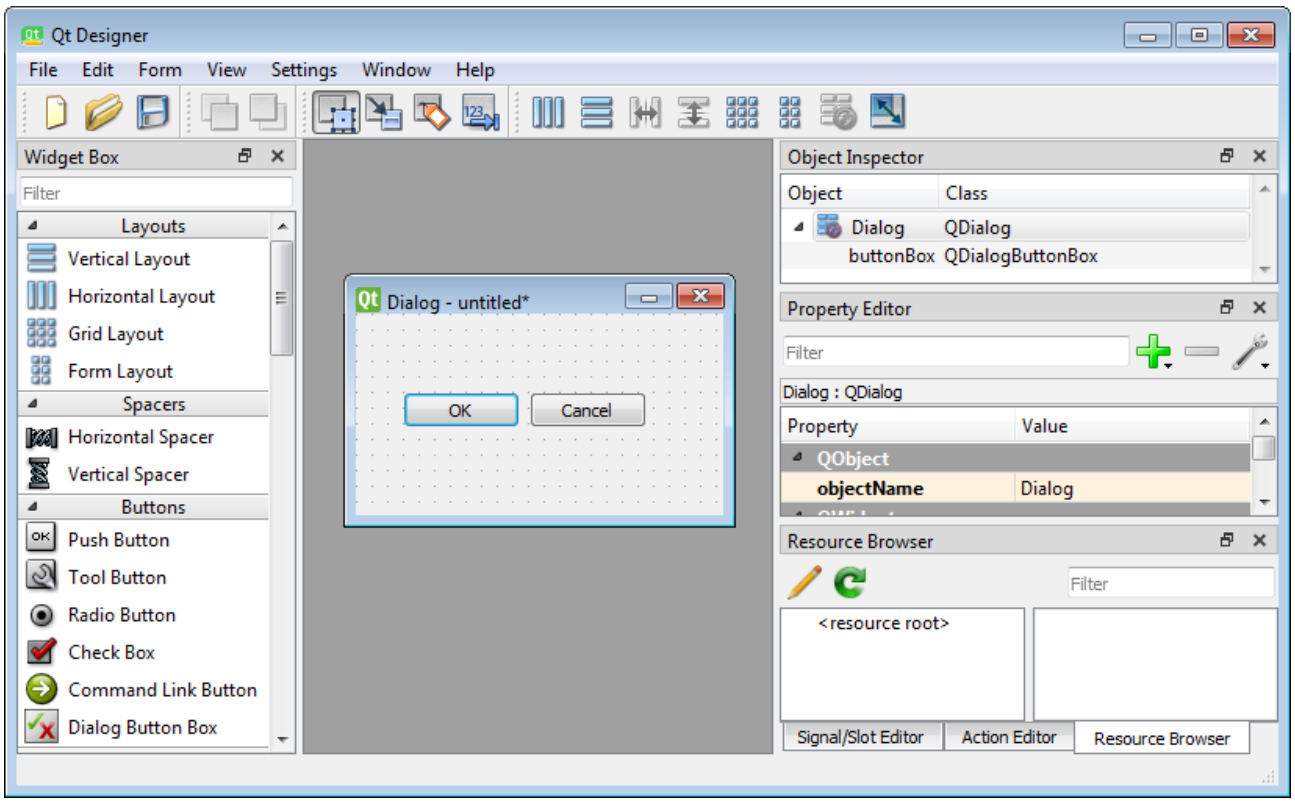

Рисунок 12 – Интерфейс Qt Designer

Есть два подхода, которые можно использовать при построении графического пользовательского интерфейса, используя виджеты Qt:

• создать, настроить компоненты и отобразить их на форме с помощью программного кода;

• воспользоваться визуальным редактором форм Qt Designer, который создаст файл формы (он будет описывать ее внешний вид, размещение, размеры, настройки, компонование и т.д.). В дальнейшем из файла формы на этапе компиляции будет создан файл с кодом программы.

Файлы формы имеют расширение .ui. Qt Designer позволяет редактировать файлы форм, содержащих настройки вида виджетов. Данную среду можно использовать как отдельную программу или воспользоваться интеграцией с оболочкой Qt Creator – редактором форм.

4.2 Проектирование схемы баз данных

На данный момент при проектировании приложений используется множество различных баз данных, таких как, например: MySQL, PostgreSQL, MS SQL Server, MS Access и Oracle.

Наиболее популярными решениями при выборе СУБД для проектов являются базы данных MySQL и PostgreSQL. Обе эти базы данных - бесплатные продукты с открытым исходным кодом. При этом по своим возможностям и характеристикам они удовлетворяют самым серьезным требованиям.

Также среди технологий СУБД набирает популярность SQLite.

SQLite - компактная встраиваемая реляционная БД. Термин «встраиваемая» указывает на то, что SQLite не использует распространенную парадигму клиент-сервер, то есть сервис SQLite не является отдельно работающим процессом, с которым взаимодействует приложение, а предоставляет библиотеку, с которой программа компонуется и ядро СУБД становится составной частью программы.

Также стоит отметить, что SQLite невероятно надежна. При выпуске версии она проходит через ряд серьезнейших автоматических тестов (проводится ~ 2 млн тестов), покрытие кода тестами 100%.

Все БД SQLite хранятся в файлах, по одному файлу на базу. Количество баз данных, а так же таблиц в них, ограниченно только свободным местом на носителе. Максимально возможный объём одной БД составляет 2 Тб. Чего предельно достаточно для разрабатываемого модуля.

Библиотека sqlite3 служит для встраивания поддержки БД SQLite в Python. В том числе эта библиотека содержит класс SQLite3.

Так как ядро базы и ее интерфейс выполнены как единое целое, большим плюсом SQLite является высокая производительность – для большинства типичных задач приложение, построенное на SQLite, работает быстрее, чем при использовании MySQL, в 2-3 раза и быстрее PostgreSQL в 10-20 раз. Прежде всего, SQLite предназначена для небольших и средних по объему приложений. Особенно актуально использование SQLite в случае, когда в основном проводятся

операции записи и считывания данных (как раз то, что нужно при парсинге сайтов).

Для выполнения поставленных задач требуется спроектировать схему базы данных, которая позволила бы оперативно записывать и считывать требуемую информацию. Была построена следующая схема (рис. 13):

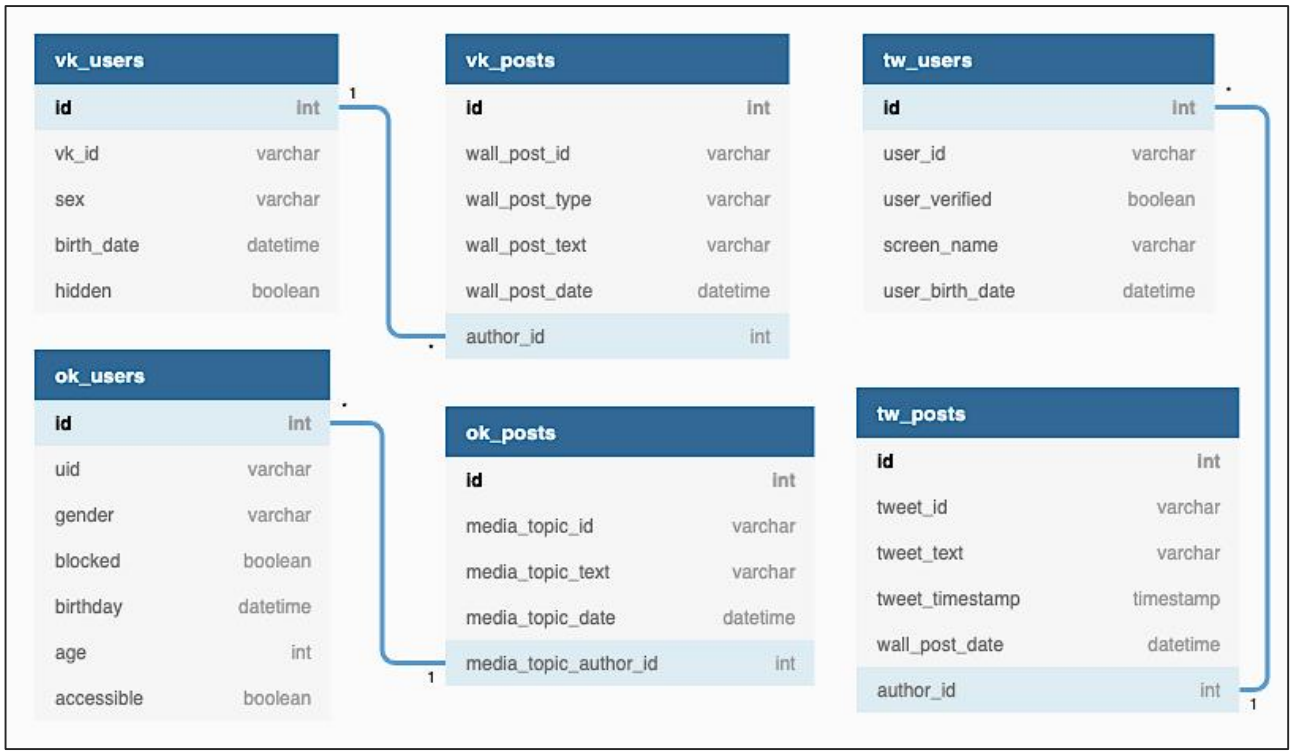

Рисунок 13 – Схема БД

База данных состоит из шести таблиц: **vk_users**, **tw_users**, **ok_users**, **vk_posts**, **tw_posts** и **ok_posts**. Соответственно по 2 таблицы для каждой социальной медиа, 1 таблица с данными о пользователях, другая с информацией о публикациях. Разделение пользователей и постов предусмотрено с целью повышения производительности обработки больших объемов данных. Описание полей каждой из сущностей представлено в табл. 3-8.

Таблица 3 – Описание атрибутов сущности vk_users

Атрибут	***Тип***	***Описание***
id	INT	первичный ключ
vk_id	VARCHAR	идентификатор пользователя ВКонтакте
sex	VARCHAR	пол
birth_date	DATE	дата рождения
hidden	BOOLEAN	скрыт ли профиль пользователя

Таблица 4 – Описание атрибутов сущности vk_posts

Атрибут	***Тип***	***Описание***
id	INT	первичный ключ
wall_post_id	VARCHAR	идентификатор записи ВКонтакте
wall_post_type	VARCHAR	тип записи
wall_post_text	VARCHAR	текст записи
wall_post_date	DATETIME	дата публикации
author_id	INT	FK vk_users.id

Таблица 5 – Описание атрибутов сущности tw_users

Атрибут	***Тип***	***Описание***
id	INT	первичный ключ
user_id	VARCHAR	идентификатор пользователя Twitter
screen_name	VARCHAR	ник
user_birth_date	DATE	дата рождения
user_verified	BOOLEAN	подтвержден ли пользователь

Таблица 6 – Описание атрибутов сущности tw_posts

Атрибут	***Тип***	***Описание***
id	INT	первичный ключ
tweet_id	VARCHAR	идентификатор записи Twitter
tweet_text	VARCHAR	текст твита
tweet_timestamp	TIMESTAMP	временная метка твита
tweet_date	DATETIME	дата публикации твита
author_id	INT	FK tw_users.id

Таблица 7 – Описание атрибутов сущности ok_users

Атрибут	***Тип***	***Описание***
id	INT	первичный ключ
uid	VARCHAR	идентификатор пользователя Одноклассники
gender	VARCHAR	пол
birthday	DATE	дата рождения
blocked	BOOLEAN	заблокирован ли пользователь
accessible	BOOLEAN	скрыт ли профиль пользователя от неавторизованных посетителей

Таблица 8 – Описание атрибутов сущности ok_posts

Атрибут	***Тип***	***Описание***
id	INT	первичный ключ
media_topic_id	VARCHAR	идентификатор медиатопика
media_topic_text	VARCHAR	текст медиатопика
media_topic_date	DATE	дата публикации
media_topic_author_id	INT	FK ok_users.id

4.3 Разработка экранных форм

Интерфейс пользователя приложения разрабатывался с учётом требований простоты, удобства и комфорта. В главном окне программы расположены 4 основные вкладки для работы (рис. 14). Перемещаясь по этим вкладкам пользователь имеет возможность работы с каждой из 3-х соцсетей, а также может установить требуемые параметры на вкладке “Settings”.

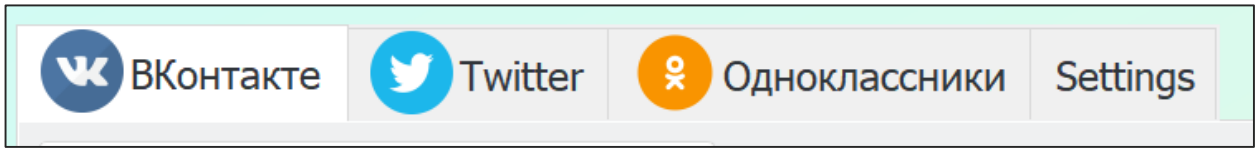

Рисунок 14 – Основные вкладки

Каждая из вкладок социальных сетей имеет схожий интерфейс и функционал. Их экранные формы представлены на рисунках 15-17.

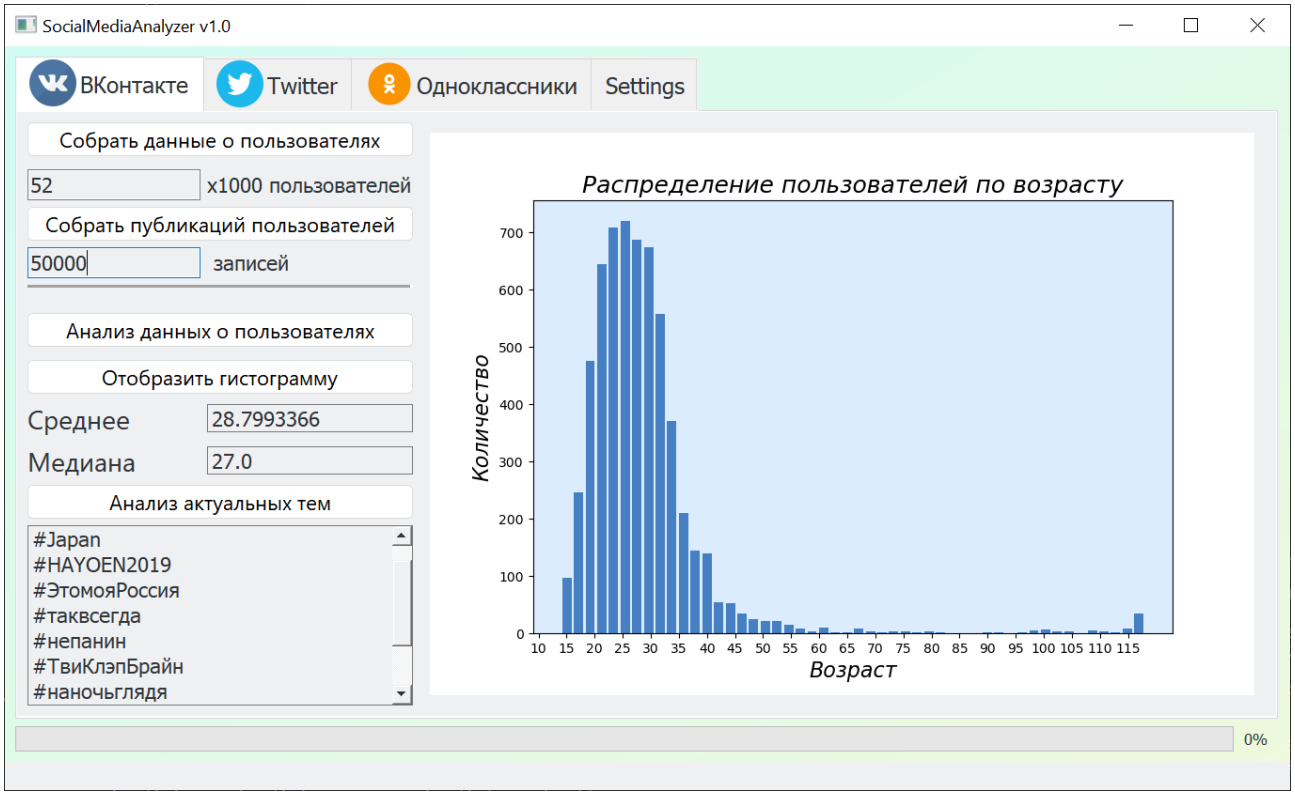

Рисунок 15 – Содержание вкладки «ВКонтакте»

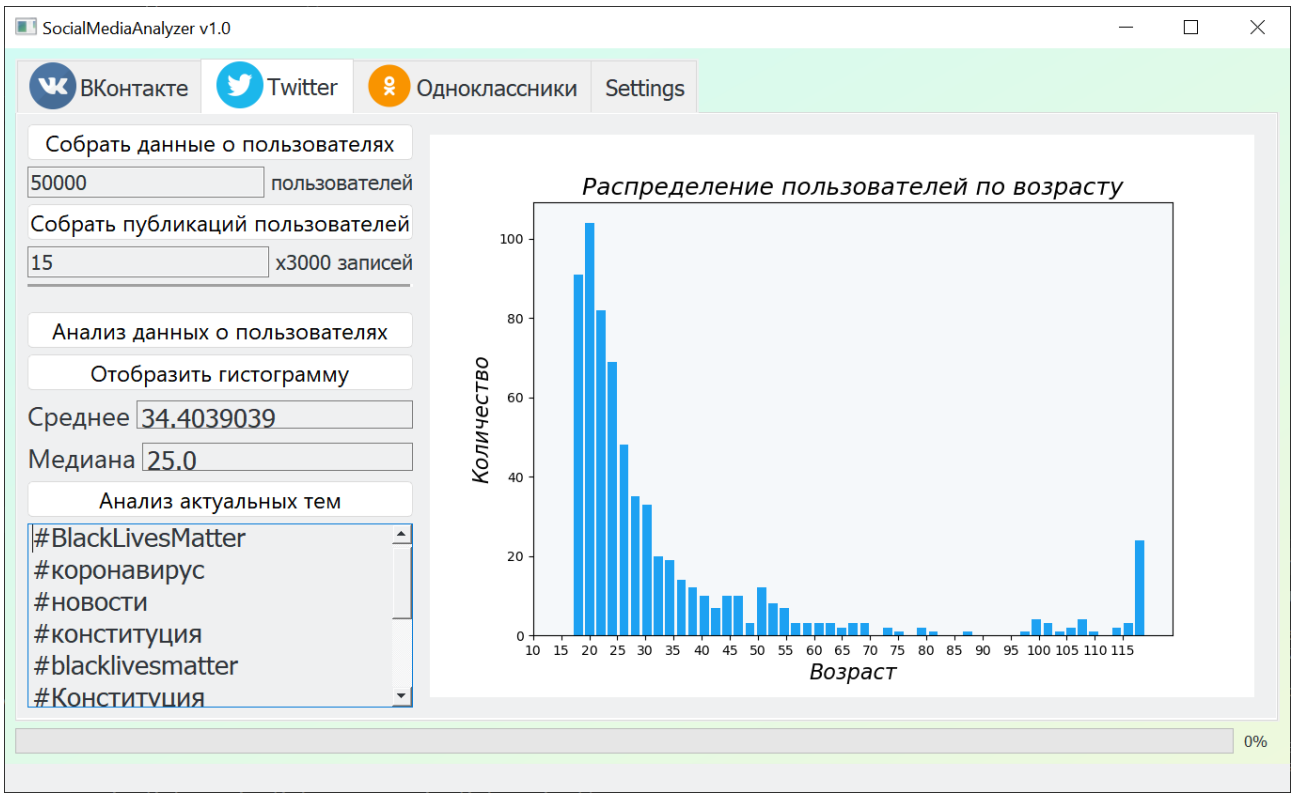

Рисунок 16 – Содержание вкладки «Twitter»

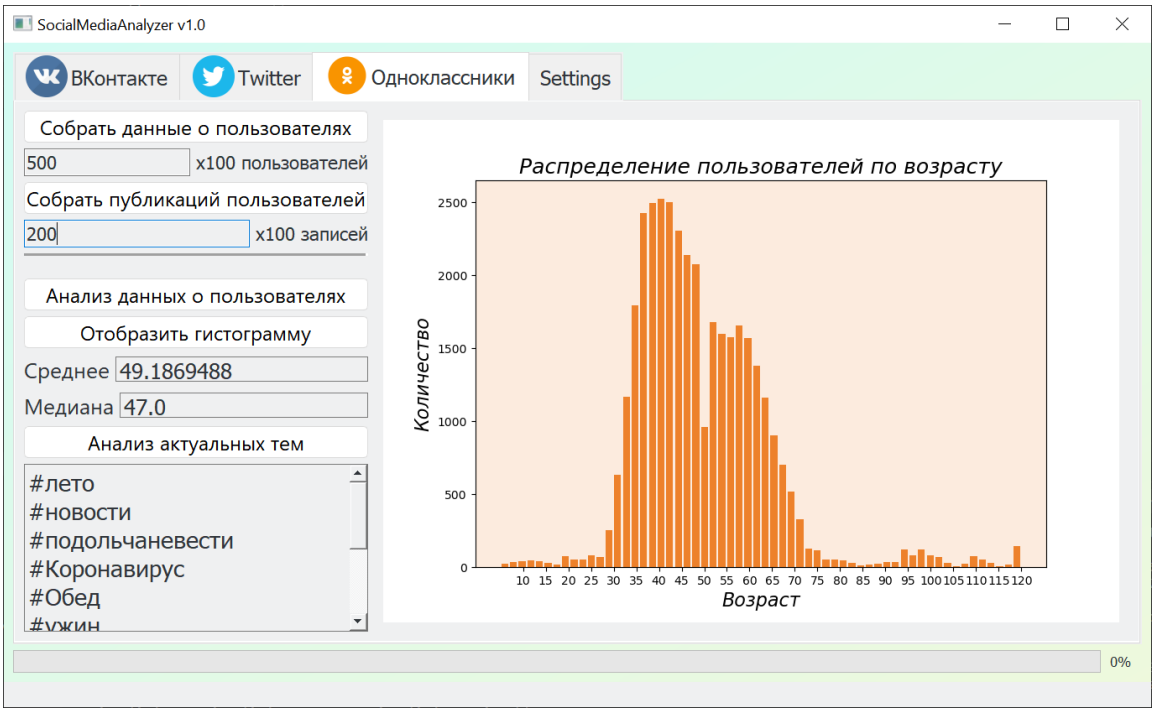

Рисунок 17 – Содержание вкладки «Одноклассники»

Четвертая вкладка «Settings» (рис. 18) позволяет просмотреть информацию о файлах CSV, куда была собраны данные из последних запросов. Поля в верху

формы предназначены для указания, откуда стоит считывать и анализировать информацию о пользователях или постах.

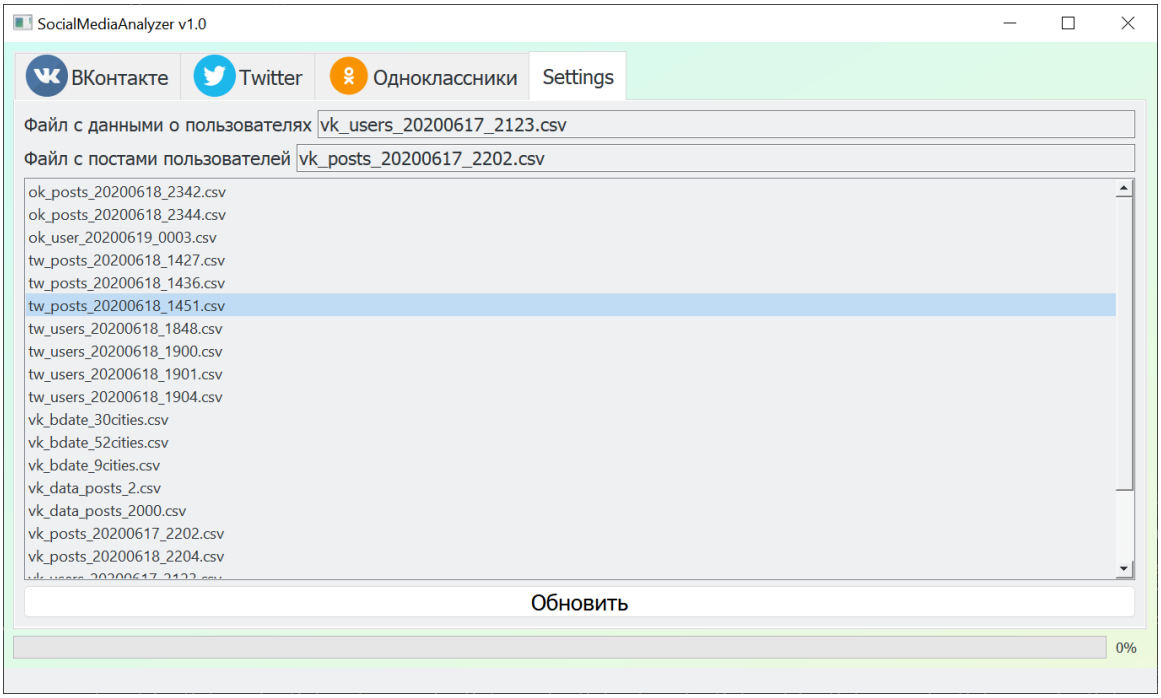

Рисунок 18 – Содержание вкладки «Settings»

Важно заметить, что файлы сохраняются с наименованиями в следующем формате:

{соцсеть}_{польз./посты}_{дата сбора: ГГГГММДД}_{время: ЧЧММ}.csv

Например, наименование файла «vk_users_20201015_1530» означает, что он содержит информацию о пользователях, собранную с соцсети «ВКонтакте» в 15:30 15 октября 2020 г.

4.4 Использование прокси и многопоточности

В настоящей работе для сбора данных в основном используются API. Но там, где возможностей API недостаточно или владелец сервиса установил слишком строгие ограничения, было решено использовать парсинг.

Как отмечалось выше, веб-парсинг (или краулинг) — это извлечение данных со стороннего веб-сайта путем загрузки HTML-кода сайта и его анализа для получения необходимых данных.

Но в чем тогда проблема? Главная проблема заключается в том, что практически все сайты не хотят, чтобы их парсили. Они желают только того, чтобы

их просматривали настоящие пользователи (кроме ботов-индексаторов Google, Yandex и других поисковых систем, естественно).

По этой причине при парсинге следует соблюдать осторожность, чтобы в разработанной программе сайт не распознал бота. Есть две основных вещи, о которых стоит помнить: важно имитировать человеческое поведение и использовать пользовательские инструменты. Ниже будут рассмотрено, какие инструменты используют сайты для определения и блокировки ботов и описаны все инструменты, которые скроют факт парсинга.

4.4.1 Эмуляторы пользовательских инструментов

Когда пользователь запускает браузер и заходит на веб-страницу, это практически всегда означает запрос у HTTP-сервера каких-либо данных. И один из самых простых способов получить контент с HTTP-сервера — применить классический инструмент командной строки, например, **cURL**.

Но важно помнить, что даже если просто выполнить: curl www.google.com, Google имеет массу методов, чтобы определить, что вы - бот, например, просто взглянув на заголовки HTTP. Заголовки — это маленькие фрагменты информации, которые поступают с каждым HTTP-запросом, передающимся на сервер, и один из этих фрагментов точно описывает клиента, выполняющего запрос. Речь идет о заголовке «User-Agent». Лишь посмотрев на заголовок «User-Agent», сервер поймет, что используется cURL.

Заголовки легко поменять с помощью cURL, и копирование заголовка User-Agent легального браузера может скрыть бота от наблюдения сайта. В реальной ситуации нужно установить более одного заголовка, но в общем случае не очень сложно искусственно создать HTTP-запрос с помощью cURL или любой другой библиотеки, которая сделает этот запрос похожим на браузерный.
Поэтому при разработке будет использоваться драйвер selenium, позволяющая подставлять заголовки “User-Agent” и будет вести себя аналогично настоящему браузеру. Подстановка заголовков выполняется случайном образом при каждом новом запуске функции (рис. 19). Использование headless будет осуществляться в браузере Firefox (рис. 20).

```
HEADERS_LIST = [
    'Mozilla/5.0 (Windows; U; Windows NT 6.1; x64; fr; rv:1.9.2.13) Gecko/20101203 Firebird/3.6.13',
    'Mozilla/5.0 (compatible, MSIE 11, Windows NT 6.3; Trident/7.0; rv:11.0) like Gecko',
    'Mozilla/5.0 (Windows; U; Windows NT 6.1; rv:2.2) Gecko/20110201',
    'Opera/9.80 (X11; Linux i686; Ubuntu/14.10) Presto/2.12.388 Version/12.16',
    'Mozilla/5.0 (Windows NT 5.2; RW; rv:7.0a1) Gecko/20091211 SeaMonkey/9.23a1pre'
]

HEADER = {'User-Agent': random.choice(HEADERS_LIST), 'X-Requested-With': 'XMLHttpRequest'}
```

Рисунок 19 – Подстановка заголовков реальных браузеров

```
options = webdriver.FirefoxOptions()
options.add_argument('-headless')
driver = webdriver.Firefox(executable_path=r'C:\geckodriver\geckodriver.exe', options=options)
driver.get(oauth_url)
```

Рисунок 20 – Запуск браузера в режиме headless

Однако и этого может быть недостаточно, поскольку на веб-сайтах теперь есть инструменты, позволяющие обнаруживать headless-браузер.

4.4.2 Прокси для эмуляции поведения человека

Человек, использующий браузер, вряд ли будет запрашивать с одного сайта 20 страниц в секунду. Поэтому, если предусматривается запрашивать с одного сайта большое количество страниц, то надо сделать так, чтобы сайт думал, что запросы идут от разных пользователей т.е. с разных IP-адресов. Другими словами, необходимо использовать прокси. Прокси в процессе работы меняются динамически таким образом, что с одного IP-адреса будет выполняться по 1-2 запроса в секунду, что позволить снять с бота подозрения. Функция для получения прокси представлена на рисунке 21.

```
PROXY_URL = 'https://free-proxy-list.net/'

def get_proxies():
    response = requests.get(PROXY_URL)
    soup = BeautifulSoup(response.text, 'lxml')
    table = soup.find('table',id='proxylisttable')
    list_tr = table.find_all('tr')
    list_td = [elem.find_all('td') for elem in list_tr]
    list_td = list(filter(None, list_td))
    list_ip = [elem[0].text for elem in list_td]
    list_ports = [elem[1].text for elem in list_td]
    list_proxies = [':'.join(elem) for elem in list(zip(list_ip, list_ports))]
    return list_proxies
```

Рисунок 21 – Получение списка бесплатных прокси

4.4.3 Многопоточность

Задача имитации поведения бота как реального пользователя решена. Осталось решить проблему скорости. Процесс парсинга усложняется существенными затратами времени на обработку данных. Распараллеливание процессов поможет в разы увеличить скорость обработки данных. Для этих целей используется библиотека multiprocessing. Создание пула рабочих процессов реализуется при помощи класса Pool. Он включает в себя методы, которые позволяют вам разгружать задачи к рабочим процессам. Ниже представлен пример использования (рис. 22):

```
pool = Pool(8)
for user in pool.map(get_tw_user_info,users):
    twitter_user_info.append(user)
```

Рисунок 22 – Использование Pool

Здесь сперва создается экземпляр **Pool** и ему указывается создать 8 рабочих процессов. Далее используется метод **map** для отображения функции **get_tw_user_info** (функция получения информации о пользователе) для каждого процесса. Наконец, в результате вся полученная инфомация с каждого процесса записывается в список **twitter_user_info.**

Применение методов распараллеливания процессов позволяет ускорить сбор и обработку информации в разы. Например, в данном случае, имея 4-х ядерный процессор с 8-ю потоками, сбор информации о 8000 пользователях сокращается с 32 минут до 4.

4.5 Применение ИС

На сегодняшний день использование Big Data в рекламных целях во многом определяет рост рынка интернет-рекламы и задает технологические тренды для всех представителей отрасли. Поэтому главная область применения данного приложения – использование для SMM и таргетированной рекламы. Для эффективного использования таргетированной рекламы необходимо полное понимание аудитории соц.сетей. Это важно, чтобы определить, подойдет ли конкретной

компании таргетированная реклама в социальных сетях и на какой именно социальной медиа-платформе стоит работать, на какой аудитории необходимо акцентировать своё внимание. Разработанное приложение поможет понять, в каких соц.сетях какая аудитория «обитает» и проанализирует тренды среди пользователей различных соц.сетей.

Для демонстрации возможностей приложения были произведены тестовые сбор и анализ данных. Они показали результаты представленные на рис. 15-18. Гистограммы количества пользователей относительно возраста представлены на рис. 23-25. Объемы собранных данных представлены в табл. 9.

Таблица 9 – Объем собранных тестовых данных

Соцсеть	Данные о пользователях	Данные о публикациях
ВКонтакте	52 000 записей	50 000 записей
Твиттер	16 000 записей	31 000 записей
Одноклассники	100 000 записей	20 000 записей

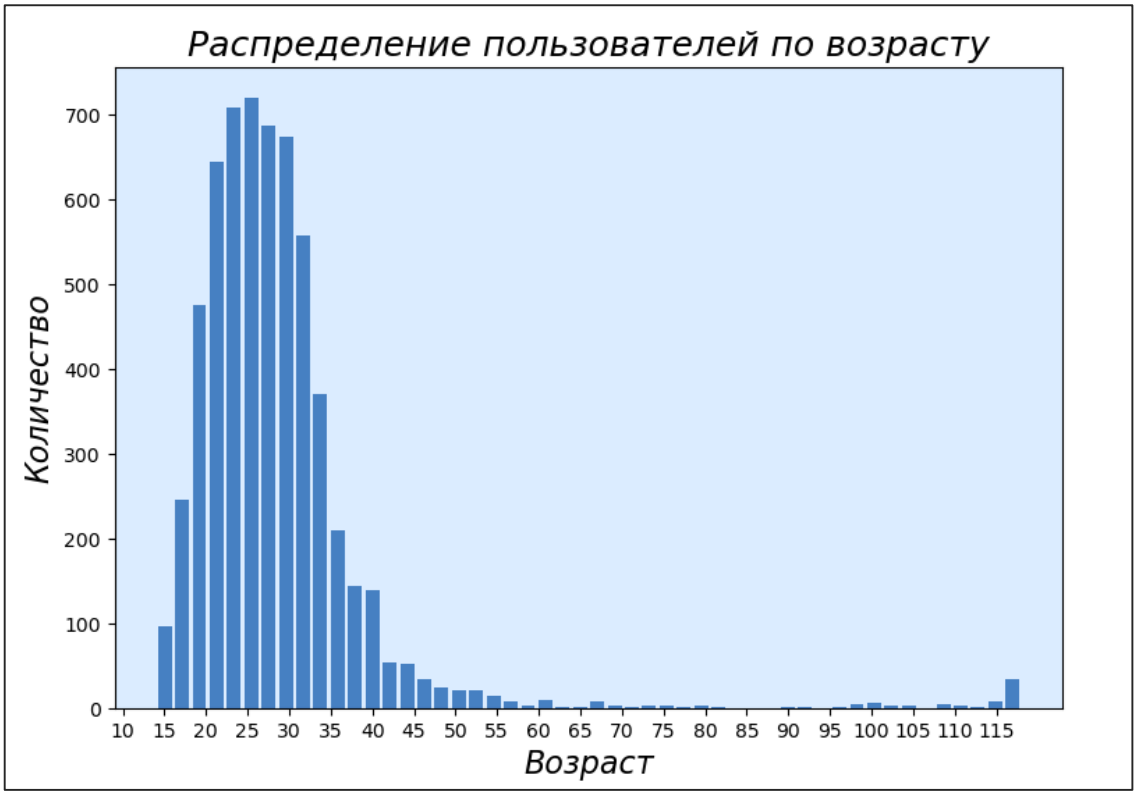

Рисунок 23 – Гистограмма «ВКонтакте»

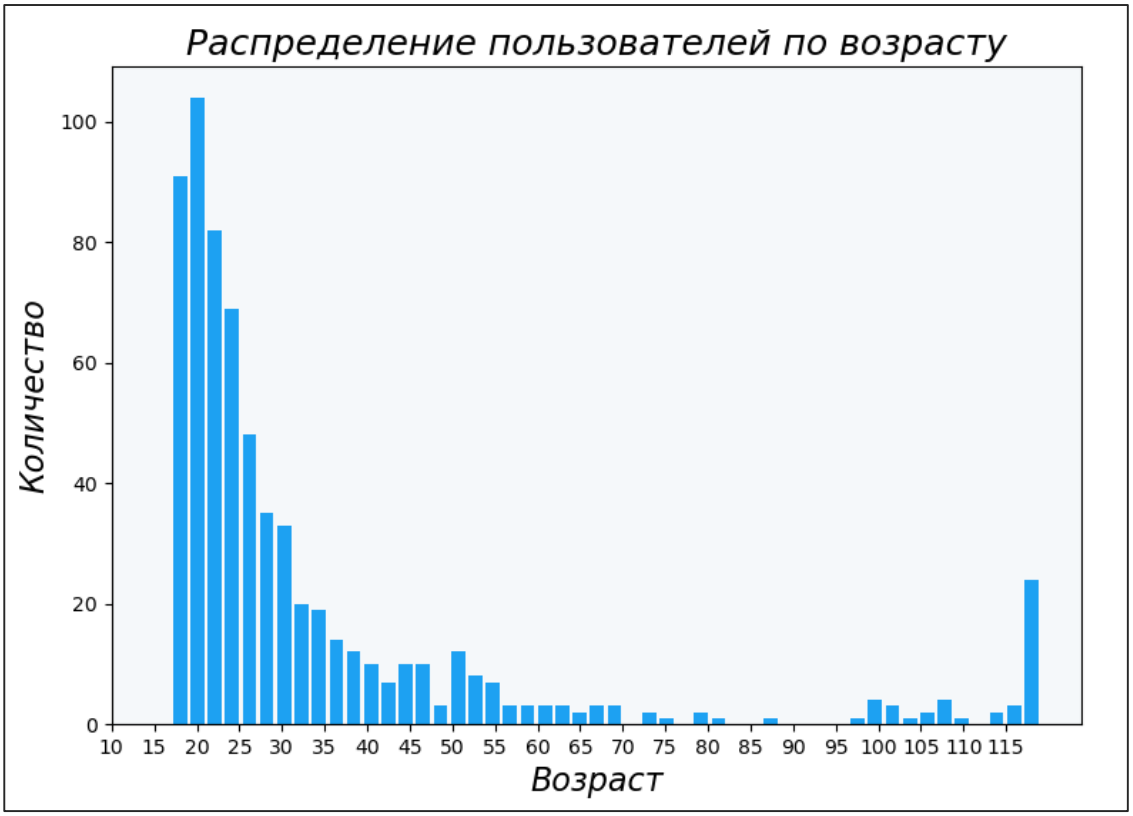

Рисунок 24 – Гистограмма «Твиттер»

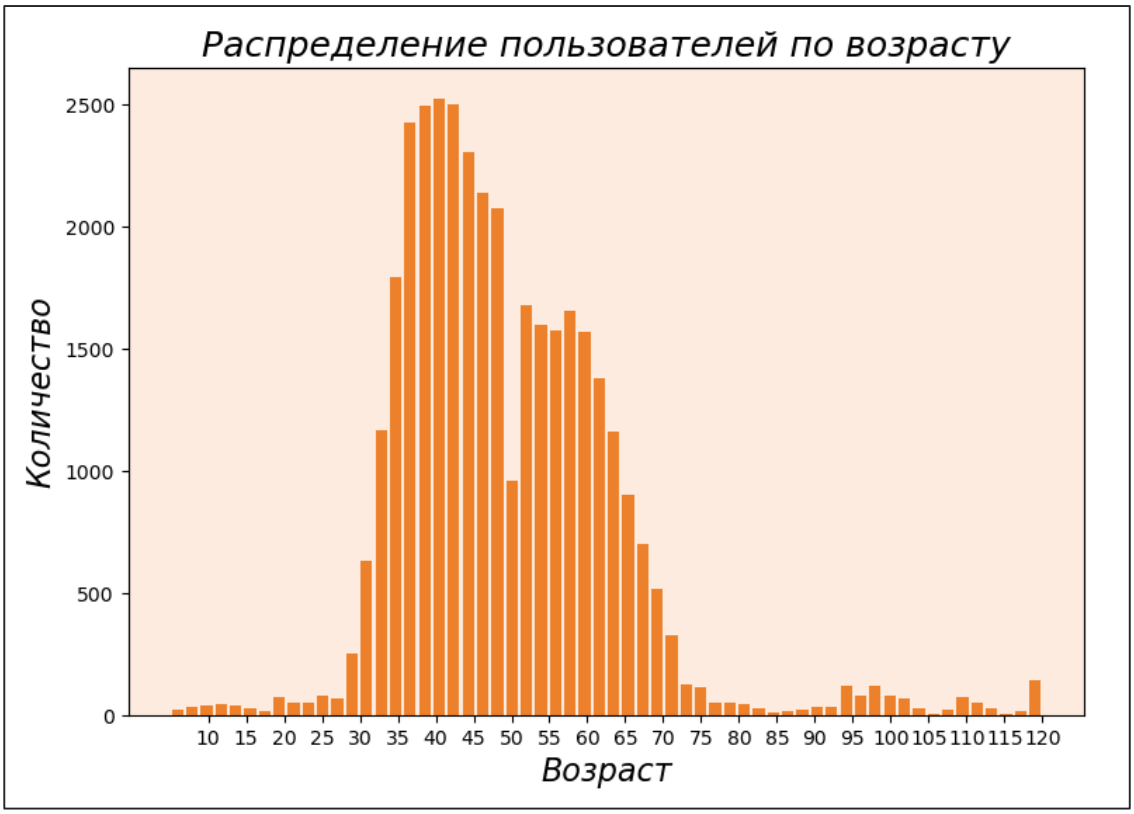

Рисунок 25 – Гистограмма «Одноклассники»

Средние значения (mean) возрастов и их медианы (median):

- ВКонтакте – mean=28.8, median=27 (лет);
- Твиттер – mean=34.4, median=25 (лет);
- Одноклассники – mean=49.18, median=47 (лет).

Предлагается для описания выводов использовать значение медианы, так как она позволяет избежать влияния выбросов (таких как количество пользователей в Твиттер возраста 120 лет = ~3% выборки, что мало похоже на правду).

Представленная информация позволяет сделать следующие выводы:

- Возраст основной аудитории пользователей ВКонтакте находится в диапазоне – 16-35 лет, Твиттера – 18-32 года, Одноклассников – 35-70 лет;
- Люди более старшего поколения предпочитают Одноклассники, вероятно это обусловлено тем, что платформа предоставляет удобный поиск людей, с которыми вы учились долгие годы назад;
- Молодые люди предпочитают Твиттер и Вконтакте для взаимодействия;
- Анализ актуальных тем показал, что в Твиттере сейчас популярны такие топики, как BlackLivesMatter (связано с беспорядками в США), Sony, PlayStation5 (из-за недавней презентации компанией Sony новой приставки PS5);
- Одной из популярнейших тем во всех трех социальных сетях оказался Коронавирус.

5 БЕЗОПАСНОСТЬ И ЭКОЛОГИЧНОСТЬ

Полноценная работа с информационной системой и ее техническая поддержка подразумевает наличие рабочих мест, а те в свою очередь – наличие помещения, где они размещены. Поэтому необходимо организовать данные места в соответствии нормативными документами и стандартами (СанПин) а также позаботиться о сохранении здоровья сотрудников при работе с ПЭВМ, разработав рекомендации и комплекс физических упражнений.

Безопасность жизнедеятельности (БЖД) – совокупность мероприятий, направленных на обеспечение безопасности человека в среде обитания, сохранение его здоровья, разработку методов и средств защиты, посредством уменьшения вредоносных воздействий до допустимых значений, выработку мер по ограничению ущерба в ликвидации последствий чрезвычайных ситуаций мирного и военного времени.

Изучение и решение проблем, связанных с обеспечением здоровых и безопасных условий, в которых протекает труд человека - одна из наиболее важных задач в разработке новых технологий и систем производства. Изучение и выявление возможных причин производственных несчастных случаев, профессиональных заболеваний, аварий, взрывов, пожаров, и разработка мероприятий и требований, направленных на устранение этих причин позволяют создать безопасные и благоприятные условия для труда человека. Комфортные и безопасные условия труда - один из основных факторов влияющих на производительность сотрудников, поддерживающих работу информационных систем. Работа сотрудников непосредственно связана с компьютером, а соответственно с вредным дополнительным воздействием целой группы факторов, что существенно снижает производительность их труда.

В следующих подразделах определены правила работы на персональном компьютере, способы безопасной утилизации носителей информации и элементов ПЭВМ, а также меры, позволяющие предотвратить чрезвычайные ситуации и их последствия.

5.1 Безопасность

5.1.1 Вредоносные и неблагоприятные факторы на рабочем месте пользователя ПК

Взаимодействуя с персональным компьютером необходимо следовать требованиям, которые предусмотрены стандартом СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».

По ГОСТ 12.0.003-2015 при работе с ПК опасными и несущими вред здоровью факторами являются:

- эл/статич. поля;
- э/м излучение;
- угроза удара током;
- низкая или высокая температура среды работ;
- ненормированная увлажненность воздуха;
- плохое качество освещения или недостаток естественного освещения;
- офтальмо-дистрофия;
- монотонность труда;
- стрессовые перегрузки;
- высокий уровень звукового давления.

Для снижения и недопущения воздействия всевозможных вредных факторов на пользователя ПК сформулированы требования, предъявляемые к помещениям, искусственному и естественному освещению, уровню акустического шума, к организации рабочего места, а также были разработаны советы пользователю ПК.

5.1.2 Организация автоматизированного рабочего места

Автоматизированное рабочее место пользователя – это место нахождения работника и средств и инструментов его труда, которая определяется на основе технических и эргономических регламентов и оснащается всевозможными средствами, необходимыми для исполнения сотрудником поставленной перед ним

поставленной задачи. АРМ представляет собой совокупность факторов окружающей среды, в том числе вредных. Вредный фактор – это фактор, воздействие которого на человека в некоторых условиях, может привести к болезням и ухудшению здоровья, а тем самым и к понижению работоспособности.

В соответствии с СанПиН 2.2.2/2.4.1340–03, к АРМ выдвигаются следующие требования:

- высота рабочей поверхности стола для взрослых пользователей должна регулироваться в пределах 680 – 800 мм; при отсутствии такой возможности высота рабочей поверхности должна составлять 725 мм;
- рабочий стол должен иметь пространство для ног высотой не менее 600 мм, шириной – не менее 500 мм, глубиной на уровне колен – не менее 450 мм и на уровне вытянутых ног – не менее 650 мм;
- поверхность сиденья должна иметь ширину и глубину не менее 400 мм, иметь с закруглённый передний край, регулироваться в пределах 400 – 550 мм и углами наклона вперед до 15 град. и назад до 5 град. угол наклона спинки в вертикальной плоскости должен обеспечивать ±30 градусов;
- стационарные или съемные подлокотники сиденья должны иметь длину не менее 250 мм и ширину 50 – 70 мм, регулироваться над сиденьем в пределах 230±30 мм и внутреннего расстояния между подлокотниками в пределах 350 – 500 мм;
- рабочее место пользователя ПК должно быть оборудовано подставкой для ног, имеющей ширину не менее 300 мм, глубину не менее 400 мм, регулировку по высоте в пределах 150 мм и по углу наклона опорной поверхности подставки до 20 град;
- клавиатура должна располагаться на поверхности стола на расстоянии 100 – 300 мм от края, обращенного к пользователю или на специальной, регулируемой по высоте рабочей поверхности, отделенной от основной столешницы.

На рисунке 26 представлено рекомендуемое размещение пользователя ПК.

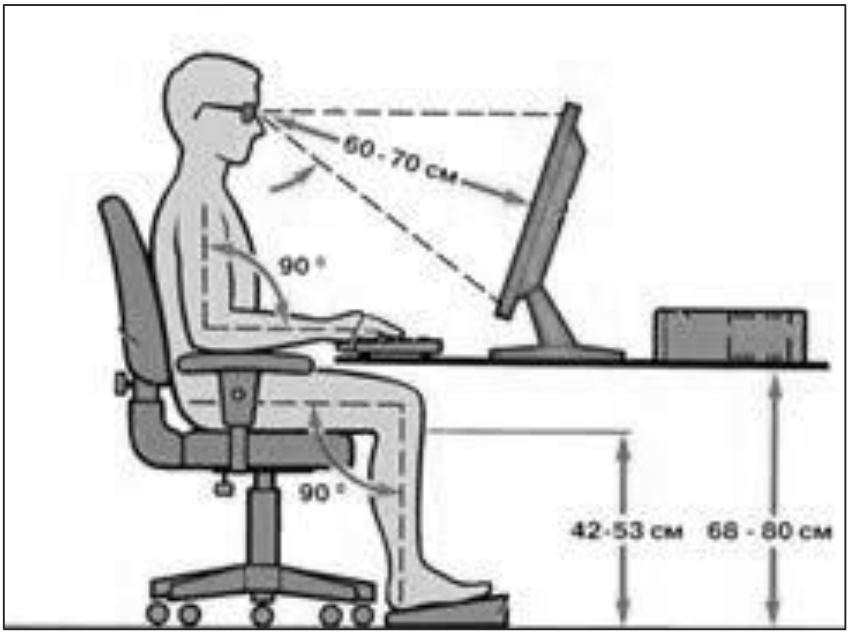

Рисунок 26 – Рекомендуемое размещение пользователя ПК

5.1.3 Освещение

Искусственное и естественное освещение является одним из важных требований, предъявляемых к помещениям с ПК. Правильное освещение повышает продуктивность, поскольку снижается нагрузка на зрительный орган. Плохое освещение, наоборот, приводит к быстрой утомляемости, ослаблении внимания при работе за ПК, ослеплении и раздраженности при чрезмерной яркости.

Виды освещения бывают следующие:

- естественное;
- искусственное;
- совмещенное;
- аварийное.

Естественное освещение непременно должно присутствовать в каждом помещении, где трудится рабочий персонал. В зависимости от расположения, оно может быть боковым, верхним или комбинированным. При совмещенном освещении недостаточное естественное дополняется искусственным.

Существует искусственное освещение двух типов: общее (равномерное и локализованное) и комбинированное. Помещения оборудуют системами общего

искусственного освещения – когда светильники располагаются в верхней зоне. Если расстояние между светильниками принимается одинаковым, то освещение считают равномерным, если светильники располагают ближе к оборудованию, то освещение называют локализованным. Комбинированным называют такое искусственное освещение, когда к общему добавляется местное.

5.1.4 Шум

На АРМ источниками возникновения шума являются тех. средства (компьютер, МФУ, вентиляционное оборудование), а также внешний шум. Уровни акустических шумов на рабочих местах при работе аппаратуры должны удовлетворять всем требованиям СанПиН 2.2.2/2.4.1340-03. Допустимые значения уровней звукового давления представлены в таблице 10.

Таблица 10 – Допустимые значения уровней звукового давления

Уровни звукового давления со среднегеометрическими частотами									Уровни звука, дБ
31,5 Гц	63 Гц	125 Гц	250 Гц	500 Гц	1000 Гц	2000 Гц	4000 Гц	8000 Гц	
86 дБ	71 дБ	61 дБ	54 дБ	49 дБ	45 дБ	42 дБ	40 дБ	38 дБ	50 дБ

5.1.5 Микроклимат

Микроклимат производственных помещений – это совокупность нормированных показателей, таких как температура, влажность и т.д., которые оказывают влияние на теплоотдачу человека и определяют самочувствие, работоспособность и производительность труда. Отсюда и одна из важнейших задача охраны труда – поддержание микроклимата АРМ в пределах установленных норм.

На рабочих местах источником существенных выделений является ПК, который повышает температуру человека, что влечет за собой снижение работоспособности и производительности, также ПК незначительно повышает температуру всего помещения в целом. Ввиду этого, поддержание температуры на необходимом уровне позволит обеспечить безопасность и комфортность при работе за ПК.

Для поддержания микроклимата в помещении используются системы вентиляции. Система вентиляции – система смены воздуха в помещении, которая предназначена для поддержания синоптических параметров помещения и подачи чистого воздуха. Для сохранения комфортных условий применяют систему естественной вентиляции, а в весеннее и летнее время года дополнительно устанавливают систему кондиционирования для полного нормирования метеорологических параметров в рабочем помещении для создания комфортабельных условий труда.

5.1.6 Исследование помещения с ПК

Работа с ПК производится в помещении площадью 18м^2 (рис. 27) В помещении находится 2 АРМ, содержащих жидкокристаллический-монитор, системный блок, мышь и клавиатуру. Это помещение полностью соответствует требованиям СанПиН 2.2.2/2.4 1340–03, поскольку на одно рабочее место приходится 9м^2. Размеры рабочей поверхности и сидений также соответствуют всем необходимым требованиям. АРМ размещены слева и справа относительно оконных проемов (рис. 28), что удовлетворяет требованиям к естественному освещению. В соответствии с техническими требованиями данное помещение оборудовано защитным заземлением. Температура помещения поддерживается в диапазоне от 23 °С до 26 °С, присутствует кондиционер для контроля температуры воздуха.

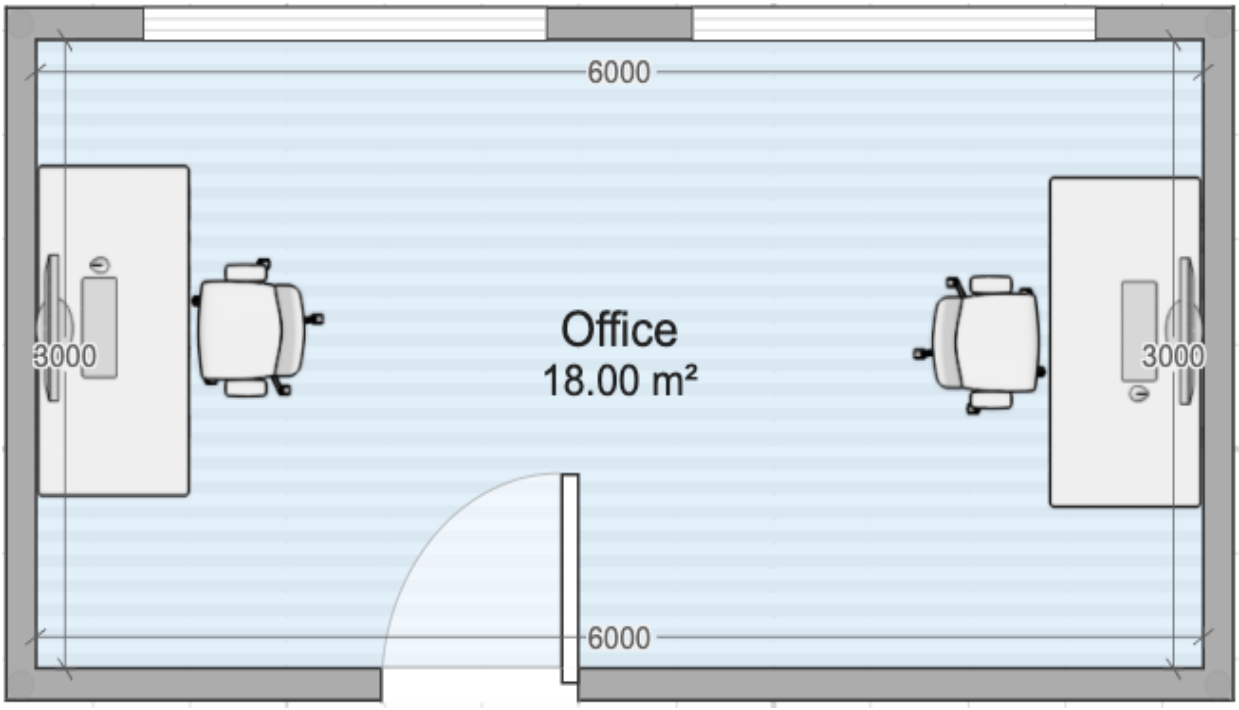

Рисунок 27 – Планировка помещения

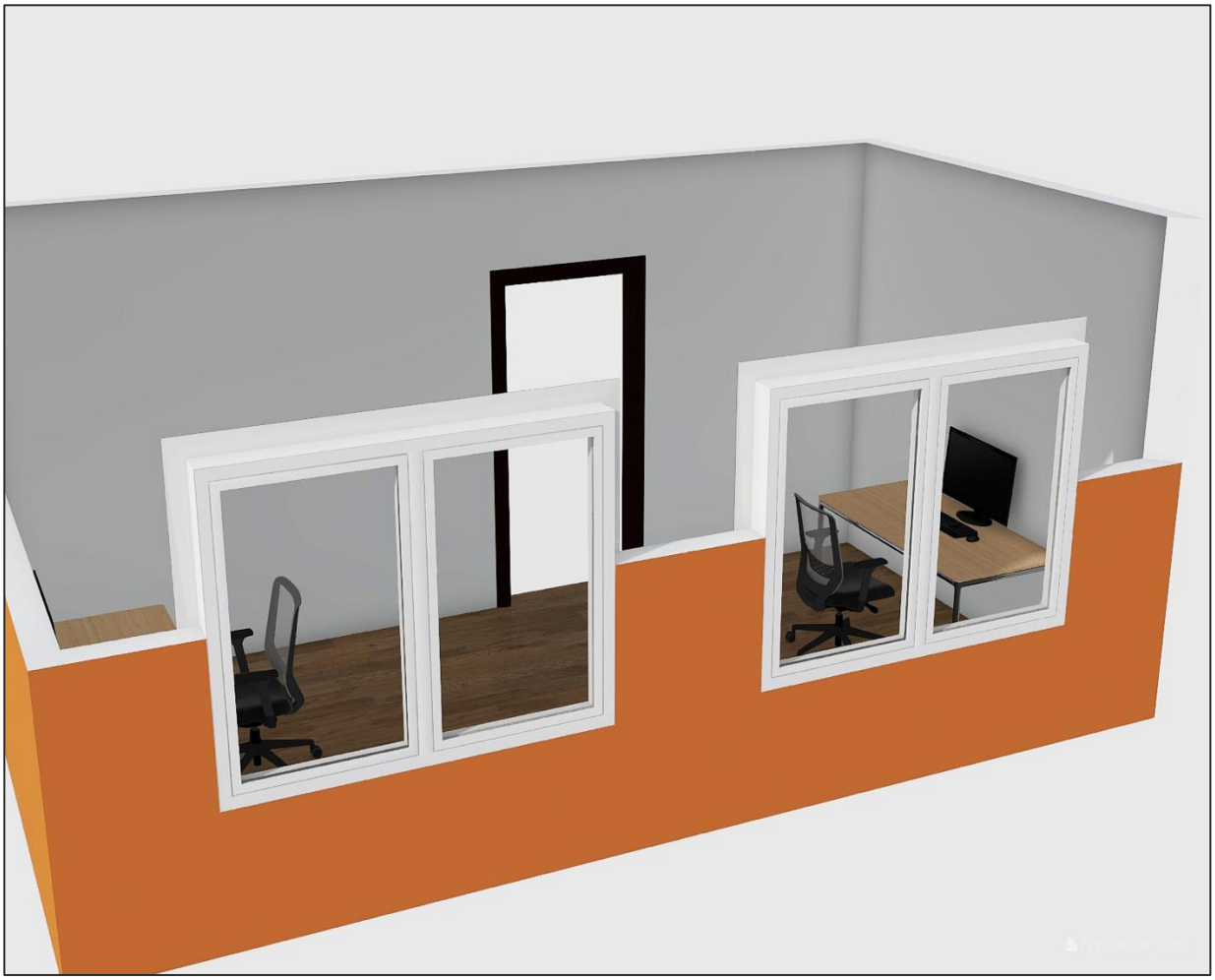

Рисунок 28 – Расположение рабочих мест

5.2 Экологичность

ПК состоит из большого количества элементов, содержащих токсичные вещества и представляющих угрозу для окружающей среды и людей. К таким веществам относятся:

- ртуть (поражает нервную систему), находится в подсветке ЖК-мониторов;
- щелочи (прожигают слизистые и эпидермис), находятся в щелочных аккумуляторах источников питания;
- никель и цинк (могут вызывать токсемию и дерматит), находится в материнской плате и аккумуляторных батареях питания для ноутбуков;
- поливинилхлорид (вызывает раковые заболевания), находится в кабелях, подключаемых к устройствам.

Поэтому ПК требует специальных совокупных методов утилизации и переработки. Этот комплекс мероприятий включает в себя разделение металлических и неметаллических частей. Затем металлические части отправляются на переплавку для последующего производства, а неметаллические части компьютера утилизируются специальным способом.

На текущий момент проектируется и интегрируется низкоотходная технология в множестве отраслей промышленности, однако полный перевод всех ведущих отраслей промышленности на безотходную технологию потребует решения весьма большого количества очень сложных технических, проектных и организационных задач.

5.3 Чрезвычайные ситуации

5.3.1 Аварийные ситуации

При работе могут возникнуть следующие аварийные ситуации:

- обрыв проводов питания электрического тока;
- поломка средста заземления;
- повреждение оборудования;
- повреждение конструкторских коммуникаций.

Во всех случаях детектирования аварийных ситуаций или выявления резких ухудшений самочувствия, а также в любых других ситуациях, которые влекут за собой прямую угрозу здоровью людей, необходимо:

- остановить выполнение работ;
- при наличии раненых, произвести оказание первой помощи;
- при надобности, организовать отключение электроэнергии;
- обеспечить эвакуацию работников и открытие запасных и аварийных выходов;
- сообщить о принятых мерах руководителю и действовать в соответствии с полученными распоряжениями.

Сотрудник, находящийся непосредственно вблизи места происшествия, обязан оказать доврачебную помощь пострадавшему, сообщить об этом оперативному дежурному или начальнику отдела. При обнаружении человека, попавшего под напряжение, незамедлительно отключить электропитание и освободить его от действия электрического тока.

5.3.2 Меры пожарной безопасности на рабочих местах

При расстановке оборудования должно быть обеспечено наличие проходов к путям эвакуации и эвакуационным выходам.

ПК должен быть установлен на крепкую и надежную опору (подставку, кронштейн и т. п.), не допускающую начала его падения. Запрещается устанавливать ПК:

- ближе 100 сантиметров от нагревательных приборов и от горючих предметов (тюлей, занавесок и т. п.);
- ближе 70 сантиметров от проходов, путей передвижения и аварийной эвакуации людей.

Перед началом эксплуатации ПК требуется произвести следующий ряд действий:

- осуществить внешний обзор места установки ПК и монитора и убедиться в выполнении требований безопасности, описанных выше;

- провести внешний осмотр персонального компьютера, шнуров и убедиться в их пригодности к использованию, а если же данные элементы повреждены, то персональный компьютер эксплуатировать запрещено;
- при наличии на, над и около ПК и ЖК-монитора легковоспламеняемых предметов (салфеток, книг, учебников, украшений и т.д.) и ёмкостей с жидкостью (вазы с живыми цветами) – убрать их на расстояние 1 метр;
- убедиться в том, что кулеры в задней крышке корпуса ПК не закрыты какими-либо предметами;
- убедиться в наличии возле ПК противопожарной ткани или огнетушителя.

Данные меры безопасности при работе с ПК позволят сократить риск ситуаций, приводящих к возникновению и распространению пожара.

5.4 Комплексы физических упражнений для сохранения и укрепления индивидуального здоровья и обеспечения полноценной профессиональной деятельности

При длительной и/или напряжённой работе с ПК, как и при его неверной эксплуатации очень часто возникают различные проблемы со здоровьем. В основном эти проблемы связаны со зрением и опорно-двигательным аппаратом. Для предотвращения этого, необходимо выполнять рекомендациии при работе с ПК. Например, 15-минутный отдых после 1,5-2-часовой работы. Во время данного перерыва необходимо встать со своего рабочего места и проделать небольшой комплекс упражнений, для снятия затекания и напряженности в мышцах.

В целом, рекомендуются следующие формы самостоятельных занятий:

- утренняя лечебная гимнастика;
- гимнастика для глаз;
- занятия физкультурой по некоторой программе;
- 15-минутная пауза во время работы за персональным компьютером;
- элементы массажа;
- закаливание организма.

Для людей (сотрудников, студентов), страдающих близорукостью, разработаны специальные упражнения типа лечебной физкультуры.

Работники с близорукостью высокой степени (6.0 дптр и более) должны выполнять следующие общие правила:

- следовать рекомендациям офтальмолога и терапевта;
- учитывать состояние здоровья;
- физическую нагрузку соразмерять с возрастом и тренированностью организма;
- помнить об ограничениях, связанных с состоянием органа зрения при выполнении некоторых видов упражнений. Так с близорукостью более 6,0 диоптрий, а также с хроническими изменениями на глазном дне противопоказаны упражнения с продолжительными и напряженными переходами из положения сидя в положение лежа и обратно;
- противопоказаны упражнения, связанные с резкими телодвижениями (прыжки, подскоки).

Так как рабочие места с ПК в преобладающем большинстве случаев – сидячие, у многих людей, работающих за персональным компьютером, наблюдается искривление осанки, что говорит о некой слабости мышц задней части тела человека, которая может привести к появлению и развитию близорукости или астигматизма. Поэтому наряду с упражнениями для глаз необходимо выполнять упражнения для укрепления мышц плеч и спины.

Примеры общих упражнений:

1) Принять положение лежа на задней части туловища, выставить руки вверх. Выполнять перекрестные движения выпрямленными руками в течение 30 секунд. Дыхание может быть произвольным.

2) Принять положение лежа на задней части туловища, руки вверх. Махи левой ноги к правой руке, правой ноги – к левой. Выполнять упражнение в течение 45 секунд. Отдых 15 секунд и второй подход 30 секунд.

3) Стоя, руки по швам, выполнение поочередного подъема колен к груди. Каждый 3-й подъем – выдох. Повторить по 20 раз каждой ногой.

4) Сядьте на пол и упритесь руками сзади. Старайтесь держать ноги в выпрямленном положении. По очереди поднимайте ноги. Сделать 25 раз на каждую ногу. Смотреть прямо.

5) Встаньте, выполните медленные вращения головы сначала в левую сторону, затем в правую и продолжайте, меняя направление каждые 5 секунд. Продолжайте повторять в течение 1 минуты

6) Встаньте, сделайте заминку. Поднимите руки вверх, насколько это возможно, потянитесь, начинайте опускать руки и медленно выдыхать. Повторите данное упражнение в течение 2-3 минут и возвращайтесь к работе.

ЗАКЛЮЧЕНИЕ

В результате выполнения выпускной квалификационной работы была разработана система поиска, сбора и анализа данных. Полученные данные были обработаны, приведены к более удобной для дальнейшего анализа форме и сохранены в формате CSV. В ходе работы был произведен анализ современных методик и инструментов анализа данных, способов сбора и оценки данных социальных сетей, задачи исследования больших данных. Рассмотрены такие вопросы как:

- определение Data Mining;
- парсинг данных с веб-страниц;
- использование прокси и многопоточности;
- программные средства разработки.

Так же были проанализированы угрозы безопасности жизнедеятельности, разработаны рекомендации по работе с системой (с ЭВМ), а также составлен комплекс физических упражнений для сохранения и укрепления здоровья людей, работающих с данной системой.

Для дальнейшего улучшения рассмотренных в работе алгоритмов необходимо провести работу в следующих направлениях:

- применение методов машинного обучения для более эффективного анализа;
- определение возраста пользователей, не указавших свою дату рождения на основании их социальных связей (например, на основании возраста их друзей).

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1 Коннолли, Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика / Т. Коннолли. – М.: Издательский дом «Вильямс», 2008. – 1120 с.

2 Димов, Э.М. Проектирование информационных систем: учебное пособие / Э.М. Димов, А.Р. Диязитдинова. – Самара: Издательство Поволжского гос. Академии, 2008. – 112 с.

3 Халилов Д. Маркетинг в социальных сетях / Д. Халилов. – М.: Манн, Иванов и Фербер, 2013. – 240 с.

4 Зоткин А. С., Ворожцов А. С. Большие данные: современные технологии обработки информации // Информационные технологии. – 2016. – 150 с.

5 Шлюйкова Д. П. Большие данные: современные подходы к хранению и обработке // Наука, техника и образование. – 2016. – №. 1. – с. 81.

6 Назаренко Ю. Л. Обзор технологии «большие данные»(big data) и программно-аппаратных средств, применяемых для их анализа и обработки //European Science. – 2017. – №. 9. – с. 30.

7 Ковалевский А. Е., Ефремов Е. А. Большие данные //Новая наука: Стратегии и векторы развития. – 2016. – №. 6-1. – с. 28.

8 Макконнелл С. Совершенный код. Мастер-класс / С. Макконнелл. – М.: Издательство «Русская Редакция»; СПб.: Питер, 2008. – 896 стр.

9 Дронов, Прохоренок. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web. - мастера. - СПБ: BXB-Петербург, 2015. - 768 с.

10 Емельянова Н., Партыка Т., Попов И. Устройство и функционирование информационных систем. – М.: Форум, Инфра, - М, 2016. - 448 с.

11 Петин В. API Яндекс, Google и других популярных веб. - сервисов. Готовые решения для вашего сайта. - СПБ: BXB-Петербург, 2017. - 480 с.

12 Полынская Г. Информационные системы маркетинга. Учебник и практикум. - М.: Юрайт, 2014. - 370 с.

13 Data Mining. Извлечение информации из Facebook, Twitter, LinkedIn, Instagram, GitHub. — СПб.: Питер, 2020. — 464 с.

14 Райгородский А. Модели Интернета. - М.: Интеллект, 2013. - 64 с.

15 Data Science. Наука о данных с нуля: пер. с англ. СПб.: БХВ-Петербург, 2017. – 335 с.

16 Анализ социальных медиа на Python: пер. с англ. – М.: ДМК Пресс, 2018. – 288 с.

17 Википедия. API. [Электронный ресурс]. Режим доступа: https://ru.wikipedia.org/wiki/API/ (дата обращения: 05.06.2020).

18 Веб-сайт ВКонтакте. Работа с API. [Электронный ресурс]. Режим доступа: https://vk.com/dev/apiusage/ (дата обращения: 10.06.2020).

19 Википедия. Selenium. [Электронный ресурс]. Режим доступа: https://ru.wikipedia.org/wiki/Selenium/ (дата обращения: 06.05.2020)

20 ГОСТ 12.0.003-2015. Система стандартов по безопасности труда. Опасные и вредные производственные факторы. Классификация. – Введ. 2017-03-01. – М.: Стандартинформ, 2016. – 10 с.

21 СанПиН 2.2.2/2.4.1340-03. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы : утв. постановлением гл. гос. санитар. врача Рос. Федерации от 30.05.2003 No118. – М.: Рид Групп, 2011. – 32 с.