

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 МАТЕМАТИЧЕСКОГО И ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ**

**РАЗРАБОТКА СЕРВИСА ДЛЯ ПОИСКА ЦЕЛЕВОЙ
АУДИТОРИИ В СОЦИАЛЬНОЙ СЕТИ «Вконтакте»**

Выпускная квалификационная работа
обучающегося по направлению подготовки
02.03.01 Математика и компьютерные науки
очной формы обучения, группы 07001403
Ганжела Егора Андреевича

Научный руководитель
доцент Румбешт В.В.

БЕЛГОРОД 2018

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
1. ПОСТАНОВКА ЗАДАЧИ НА РАЗРАБОТКУ СЕРВИСА	7
1.1 Проблема создания рекламного контента на основе информации, извлекаемой из социальных сетей.....	7
1.2 Обзор и анализ существующих сервисов для поиска целевой аудитории в социальных сетях	14
1.3 Требования к разработке веб-сервиса.....	20
2. ПРОЕКТИРОВАНИЕ И ПРОГРАММНАЯ РЕАЛИЗАЦИЯ СЕРВИСА	29
2.1 Разработка объектно-ориентированной модели программного обеспечения	29
2.2 Технология получения доступа к информации из социальной сети Вконтакте	42
2.3 Проектирование структуры и интерфейса сайта.....	48
3. ИСПЫТАНИЯ СЕРВИСА.....	54
3.1 Программа и методика испытаний	54
3.2 Результаты испытаний	61
ЗАКЛЮЧЕНИЕ	65
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ.....	67
ПРИЛОЖЕНИЕ 1	71
ПРИЛОЖЕНИЕ 2	71
ПРИЛОЖЕНИЕ 3	71
ПРИЛОЖЕНИЕ 4.....	71

ВВЕДЕНИЕ

Уже достаточно давно существует поговорка, что реклама является двигателем торговли. Со все большим развитием предпринимательской деятельности подтверждается ее справедливость. И это не просто пустые слова. Рекламный бизнес постоянно присутствует неподалеку с торговлей и предпринимательской деятельностью, чего просто невозможно не заметить. Сейчас люди все чаще и с удовольствием тратят деньги в интернете, поэтому именно здесь нужно искать своих будущих клиентов. Время обычной рекламы прошло – люди от нее устали и больше ей не доверяют, а интернет-маркетинг даёт новые способы рассказать миру о нашем товаре или услуге.

Как уже было сказано, интернет давно перестал быть просто средством передачи информации (media), превратившись из места для размещения рекламы в полноценный канал продаж, что позволило добиться максимально быстрой конверсии, или перехода обычных посетителей сайта в потенциальных, а затем и реальных покупателей. Каждый современный предприниматель стремится открыть виртуальные двери перед своими покупателями, и все больше людей по всему миру заинтересованы в таком виде торговли.

Интернет-маркетинг довольно обширное понятие, включающее в себя и SEO продвижение сайтов, и контекстную рекламу, и контент маркетинг и продвижение в социальных сетях (SMM). аббревиатура SMM расшифровывается как social media marketing. Это комплекс мероприятий по использованию социальных медиа в качестве каналов для продвижения компании и решения других бизнес-задач. Эксперты в области коммуникации считают, что социальные сети (так называемый Web 2.0) произвели некоторую революцию. Раньше коммуникация была односторонней – на рекламную информацию брендов пользователи чаще всего не могли ответить или делали это слишком часто из-за технических

сложностей. Сегодня каждый пользователь может зайти на страничку бренда в социальной сети и задать вопрос, оставить жалобу, поблагодарить за быструю доставку и т.д. Одной из ключевых особенностей социальных сетей является максимальная возможность для таргетинга целевой аудитории (ЦА).

Мы можем настраивать таргетинг по гендерному признаку, географическому положению, интересам. Таким образом, мы можем делать фокусировку и показывать рекламу только тем людям, которые нам действительно нужны и которые с большей вероятностью заинтересуются нашим продуктом. Понимание целевой аудитории безусловно важно и для размещения рекламы в том или ином информационном канале. Необходимо определить, что смотрит, читает, слушает ваша целевая аудитория. При разработке рекламного сообщения необходимо учитывать различные особенности целевой аудитории (психологию, традиции, ментальность, религиозные, этические, политические и прочие убеждения, привычные способы получения информации и т.д.). Каким образом мы можем проанализировать большое количество людей из социальных медиа и понять, кто же из них подходит под критерии нашей целевой аудитории. Именно для этого создали парсеры для социальных сетей. Парсер — это программа/скрипт, которая по заданному алгоритму собирает нужную информацию на сайте. Так как мы с вами говорим о социальной сети ВКонтакте, то в нашем случае парсер ВК — это скрипт, обращающийся к API ВК и получающий от него нужную информацию о пользователях, сообществах и контенте.

Актуальность темы данной работы обуславливает всё вышесказанное, а также возможность выхода с данным сервисом на рынок таргетинговых услуг.

Разрабатываемый сервис ориентирован в первую очередь на маркетологов, которые в свою очередь собирают целевую аудиторию для рекламы каких либо товаров. Например если какой либо аккаунт присутствует в 3х группах по тематике рыбалка, то его можно включить в

список потенциальных целевых клиентов, которым в последствии можно разослать рекламу по тематике рыбалка.

Степень изученности вопроса взаимосвязи социальных сетей и бизнеса достаточно высока. Во-первых, это различные публикации в специализированных журналах и на интернет-порталах и ресурсах, посвященных бизнесу. Во-вторых, это различные книги, посвященные исследованиям на данную тему. Это, например, книги А. Парабеллума, Н. Мрочковского, В. Калаева «Социальные сети. Источники новых клиентов для бизнеса» и Д. Румянцева «Продвижение бизнеса в ВКонтакте. Быстро и с минимальными затратами».

В основном, все исследования по данному вопросу, проведенные специалистами, сводятся к анализу влияния социальных сетей на бизнес активность компании и результатов их использования на прибыль, повышения узнаваемости, лояльности и увеличение продаж и упорядочиванию инструментов использования социальных сетей.

Исследователи приходят к выводу, что популярность социальных сетей среди бизнес-сообщества растет в геометрической прогрессии, а их эффективность как нового маркетингового инструмента во много раз превосходит другие.

Цель данной работы — разработать сервис для поиска целевой аудитории в социальной сети ВКонтакте.

Задачи:

1. Разобрать проблему создания рекламного контента на основе информации, извлекаемой из социальных сетей
2. Провести обзор и анализ существующих сервисов для поиска целевой аудитории в социальных сетях
3. Подобрать требования к сервису
4. Провести разработку объектно-ориентированной модели программного обеспечения веб-сервиса

5. Описать технологию получения доступа к информации из социальной сети Вконтакте при разработке веб-сервиса

6. Спроектировать структуру и интерфейс веб-сервиса

7. Разработать программу и выработать методику испытаний

8. Провести тестовые испытания на данных

9. Получить результаты испытаний

Предмет данного исследования — сервис для поиска целевой аудитории в социальной сети ВКонтакте.

Объект — группы в социальной сети ВКонтакте

Структура работы состоит из введения, трёх глав, заключения, списка литературы и приложений. Во введении обозначена актуальность темы дипломной работы, определены цели задачи. Первая глава посвящена постановке задачи на разработку сервиса. Вторая глава представляет собой проектирование и программную реализацию веб-сервиса. В третьей главе проведены испытания веб-сервиса. В заключении сделаны общие выводы по работе. В работе имеются 29 рисунков, 3 таблицы и 3 листинга кода.

1. ПОСТАНОВКА ЗАДАЧИ НА РАЗРАБОТКУ СЕРВИСА

1.1 Проблема создания рекламного контента на основе информации, извлекаемой из социальных сетей

Типичная PR-кампания в социальных сетях включает в себя несколько последовательных этапов. Первоначально требуется разработать стратегию коммуникаций на основе данных, полученных в ходе мониторинга. Стратегия коммуникаций должна включать в себя перечень работ, необходимых для достижения цели PR-кампании, желаемые результаты, планируемый бюджет, правила привлечения и взаимодействия с аудиторией.

На следующем этапе создается так называемая бренд-платформа — совокупность сообществ в социальных сетях, обеспечивающих взаимодействие бренда с целевой аудиторией. В случае если бренд работает сразу на несколько целевых аудиторий, для каждой из них потребуется создать отдельное сообщество.

После того как сообщества будут созданы, придется позаботиться о наполнении их контентом. Новости, видео, фотографии — все это необходимо, чтобы сохранить интерес аудитории к группе и поддержать общение. При этом на брендированный контент должно приходиться не более трети общего объема содержимого сообщества, еще треть должна быть отведена под fun-контент и треть — под полезный контент.

Успех любой стратегии в социальных сетях в конечном счете определен ее способностью соединяться с людьми на непосредственном, человеческом уровне. В лингвистике есть явление, известное как «переключение кода». То есть, когда человек переходит с одного языка на другой и обратно в течение одной беседы. Значение этого явления, примененного в социальных сетях масштабно. Социальные медиа медленно, но верно, стирают грань между личным и профессиональным. Связав это

с «переключением кода», мы получаем новый подход в поведении бренда в социальных сетях [88].

Дело в том, что в SMM не используется рекламный формат. Основной механизм взаимодействия — это общение на актуальные для пользователя темы и распространение интересного для него контента. Естественно, и общение, и контент содержат обязательную промопривязку, но при этом в отличие от рекламы представляют ценность для пользователя [93, 39].

Среди бесплатных методов раскрутки в социальных сетях можно отметить следующие:

Создание сообщества или группы — доступно в большинстве социальных сетей, чаще всего такая функция бесплатна. В созданные группы можно пригласить друзей и других заинтересованных в продукции или товарах компании лиц. Далее следует позаботиться об организации общения между пользователями в группе: создавать интересные темы, через создание искусственных профилей имитировать дискуссию между реальными пользователями. Заинтересованных пользователей можно приводить с помощью поиска по интересам в аналогичных группах других сервисов. Давать по возможности полноценные ответы на вопросы пользователей.

Создание публичных страниц (fan page). Методы раскрутки в таких страницах напоминают продвижение в сообществах и группах.

Платные методы социальной рекламы:

Реклама в приложениях для социальных сетей: программы для создания картинок, аватаров, различные игры. Самые популярные из них используются миллионами пользователей.

Таргетированная реклама — направленная на определенную категорию пользователей. Наиболее распространена среди способов платной раскрутки в социальных сетях. Сюда можно отнести рекламу для конкретного города, страны, для людей с определенным возрастом, интересами.

Проведение конкурсов, акций, скидок для целевой аудитории с призами для победителей.

«ВКонтакте» (<http://vk.com>) — это социальная сеть для быстрой и удобной коммуникации между людьми по всему миру. Задача ВКонтакте — в каждый отдельно взятый момент оставаться наиболее современным, быстрым и эстетичным способом общения в сети [63]. На сайте зарегистрировано более 290 миллионов пользователей. Более 69 000 000 посетителей заходят на сайт каждый день. Более 3 000 000 000 страниц открываются ежедневно. Примерно 65% посетителей проживают в России. Более 56% пользователей сайта старше 25 лет [14].

ВКонтакте у своих посетителей вызывает положительные эмоции, потому что чаще всего они используют ресурс для общения и развлечения. Это формирует высокую лояльность пользователей.

Есть два основных формата работы в ВКонтакте.

Первый — работа исключительно внутри социальной сети. В этом случае все привлеченные посетители аккумулируются в сообществе, там же осуществляются продажи, коммуникация с клиентами и их информирование.

Второй вариант — «перевод» пользователей на внешний Интернет-ресурс. Такой вариант подходит компаниям, которые осуществляют основные транзакции (заказ товаров или услуг) на своем сайте, а не через социальные сети. Если же сайт информационный (блог или тематический портал), там посетители из социальных сетей читают материалы о компании и товарах. И в таком случае сообщество в социальной сети — дополнительный источник трафика для основного проекта.

Какой вариант выбрать, зависит от изначальных условий. Оба формата допустимы и могут дать результат. Второй вариант особенно актуален для тех, кто зарабатывает на своих сайтах, продавая рекламу, в частности для всевозможных СМИ. Иногда форматы смешивают: например, все информационные материалы размещают в сообществе, а продажи осуществляют только на сайте [83, 29].

Любой проект во ВКонтакте решает две основные задачи: увеличение прибыли и обратная связь с клиентами. Так или иначе, все сводится к этому.

С другой стороны, задача увеличения прибыли может решаться как напрямую — непосредственные продажи товаров или услуг, так и косвенно — с помощью повышения узнаваемости бренда, роста его репутации и, соответственно, общей лояльности к компании.

Главный интерес для бизнеса в социальной сети «ВКонтакте» представляют «группы» и «паблики» — это сообщества людей объединенных схожими интересами. Количество участников группы может измеряться миллионами.

Функционал группы в принципе не большой. Через раздел «Обсуждения» можно создать небольшой форум по заданной теме для участников/подписчиков сообщества, а «Стена» группы служит информационной площадкой, где администраторы и модераторы группы могут выложить информационный и развлекательный контент. Еще одна возможность — рассылка приглашений подписчикам и другим пользователям социальной сети на мероприятия, которые проходят в условиях реального времени и организованы компанией, которой принадлежит группа.

Помимо того, что ВКонтакте позволяет осуществлять прямые продажи, у любого сообщества есть возможность делать повторные продажи. В какой-то степени работа с группой в современных условиях напоминает e-mail-маркетинг. Только в случае с последним люди получают новости на свою электронную почту, а в социальных сетях контакт с подписчиками в ленте новостей постоянный. Но суть та же: формируется сообщество (база потенциальных клиентов) и поддерживается постоянное с ними взаимодействие. В отличие от аудитории, которая приходит с поисковых систем и чаще всего не возвращается, люди в социальных сетях постоянно на связи, так как становятся членами сообщества и получают новости о всех новых публикациях на его стене.

Кроме того, сообщество может выполнять функции «технической поддержки». Как показывает практика, многим пользователям удобнее задать

вопрос через сообщество, чем позвонить по номеру, который еще нужно узнать, или «достучаться» до онлайн-консультанта на сайте, если таковой имеется. С помощью правильно выстроенной коммуникации с клиентами можно существенно снизить негатив и повысить общую лояльность к компании.

Благодаря разделу «статистика сообщества» всегда можно узнать об аудитории сообщества и ее активности. В этом разделе есть несколько вкладок. Во вкладке «Посещаемость» можно увидеть количество уникальных посетителей и просмотров. Есть данные за сутки и за месяц плюс масса дополнительных факторов: пол и возраст посетителей; география (страна и город); источники переходов (откуда посетители попали в ваше сообщество); количество посетителей, вышедших из группы или вступивших в нее; какие разделы посетители просматривали: видео, фотоальбомы, обсуждения, аудиозаписи.

Следующая вкладка «Охват» — это пользователи, просмотревшие записи сообщества на «стене» или в разделе «Мои Новости». Здесь также есть дополнительные данные по полу, возрасту и географии.

А во вкладке «Активность» представлены следующие данные: сколько пользователей «лайкнули» запись, то есть нажали кнопку «Мне нравится»; сколько пользователей сделали репост, то есть нажали кнопку «Рассказать друзьям»; сколько пользователей упомянули страницу группы; сколько пользователей скрыли группу из новостей; общее количество оставленных комментариев; кроме того, в этой вкладке есть информация о количестве комментариев к конкретным разделам: обсуждения, фото или видеозаписи. Здесь же представлен график обратной связи. На нем учитываются не только действия пользователей, произведенные непосредственно на стене группы или страницы, но и весь дальнейший путь сообщения. Если пользователь размещает ссылку на сообщение у себя на странице при помощи функции «рассказать друзьям», реакция других пользователей на такую копию также учитывается.

Любой проект может присутствовать в ВКонтакте в трех основных форматах: публичная страница (паблик), группа или профиль. У каждого формата есть свои особенности.

Основные различия между публичной страницей и группой — это их функционал. В пабликах на первом плане находится контент. Сразу под описанием сообщества стоит микроблог. И в отличие от групп в пабликах над «стеной» нет блока с фотографиями и блока новостей.

На публичных страницах пользователи могут «предлагать» свои посты, используя функцию «Предложить новость». В некоторых случаях это избавляет от необходимости вести свое сообщество, так как пользователи сами генерируют контент.

Крайне важный нюанс, публичная страница есть в блоке интересных страниц на странице пользователя. По сути, это пять рекламных баннеров на самом видном месте профиля. Если владелец аккаунта подписан на публичную страницу и регулярно ее посещает или делает репост записей паблика из ленты новостей, паблик будет находиться в этом блоке постоянно.

В отличие от публичных страниц в группах основной акцент делается на коммуникацию между пользователями. Этому способствуют:

1. Блок «Обсуждения», который можно закрепить над «стеной» группы, сразу после блока новостей. Это аналог форума.

2. В группе посетители могут общаться прямо на «стене». Например, задавать свои вопросы. Это увеличивает коммуникативный эффект, но и количество спама тоже.

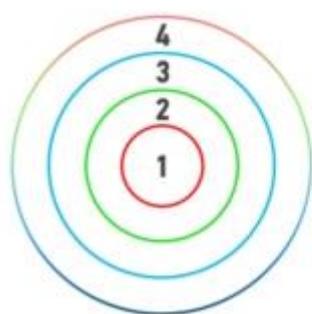
3. В группе, в блоке «Новости», есть вики-разметка, что позволяет создать красивое дополнительное меню.

Основная проблема создания рекламного контента на основе информации из соц. Сети ВКонтакте заключается в поиске целевой аудитории ВКонтакте для эффективной настройки таргетированной рекламы, и разработке матрицы целевой аудитории.

1-ый этап - Понять, кто ваш клиент.

2-ой этап - понять, как найти вашего клиента ВКонтакте.

Для эффективной настройки таргетированной рекламы ВКонтакте используется так называемая матрица поиска целевой аудитории. Матрица поиска целевой аудитории представлена на Рисунке 1.1.



1. **Горячая** (друзья, база, ретаргетинг).
2. **Теплая** (конкуренты, смежные сообщества, сервисы, сайты, приложения, look-alike, ретаргетинг на активность в сообществах, профессиональные сообщества, закрытые группы).
3. **Холодная** (категории интересов, большие паблики).
4. **Смежная/переходная** (дни рождения, путешественники, образование).

Рис. 1.1 – Матрица целевой аудитории

Первым настраивается “горячая аудитория”. Это люди, которые знают вас или ваш продукт (друзья, ваша база и аудитория ретаргетинга по вашей базе и/или посетителям вашего сайта).

Чтобы показывать рекламу только вашим друзьям и подписчикам, вы сначала должны загрузить их в отдельный файл и добавить в базу ретаргетинга ВКонтакте.

Друзей и подписчиков можно собрать через парсер Segmento-Target [1].

Файл со списком друзей и подписчиков загружается в базу ретаргетинга ВКонтакте. Подробная инструкция по загрузке базы представлена по ссылке [2].

По этой базе видно отдельную рекламную кампанию, в которой используются знакомые вашей базе изображения (ваши фотографии + фотографии ваших товаров/услуг с логотипом компании).

Далее к горячей аудитории относятся также подписчики ваших личных сообществ ВКонтакте. На них легко настраивается реклама через “Сообщества” в пункте “Интересы”. Вставляются туда адреса сообществ и реклама будет показываться только участникам этих сообществ.

После “горячей аудитории” настраивается “теплая аудитория”. Это люди, у которых есть определенная проблема в жизни, о которой они знают.

Остальные группы настраиваются аналогично.

Из всего вышесказанного появляется основная проблема, как сформировать целевую аудиторию, т.е. как создать файл со списком друзей и подписчиков по конкретной тематике для дальнейшей рекламной компании.

Приведём пример создания рекламного контента, например если мы рекламируем детские развивающие квесты, нам нужно спарсить женщин у которых есть дети. Причем сегментировать детей по возрасту от 5 до 8 и от 8 до 12 и под каждую писать свой текст.

Таким образом, в данном разделе рассмотрена проблема создания рекламного контента на основе информации из социальной сети Вконтакте. Выявлена основная проблема, которая заключается в формировании целевой аудитории, т.е. как создать файл со списком друзей и подписчиков по конкретной тематике для дальнейшей рекламной компании.

1.2 Обзор и анализ существующих сервисов для поиска целевой аудитории в социальных сетях

Таргетированная реклама — один из самых быстрорастущих видов рекламы. Тем более, когда вопрос касается рекламы в социальных сетях. Ведь она позволяет найти не только людей, горячо заинтересованных в покупке (как например, в контекстной рекламе), но и огромное количество потенциальных покупателей, которые выдают свой базовый интерес к определенному товару/услуге своим поведением в социальных сетях. Так человек, состоящий сразу в нескольких сообществах одной тематики, скорее всего в ней реально заинтересован. Собрав таких людей, подписанных одновременно на 2-3 паблика одной тематики, можно получить довольно

конверсионную аудиторию, которая будет гораздо чаще реагировать на показываемое объявление, чем просто аудитория, состоящая во всех тематических пабликах (отсекаются накрученные и случайные подписчики). На подобных принципах и базируется таргетированная реклама в социальных сетях.

А данном разделе будет рассматриваться работа с таргетированной рекламой в самых популярных социальных сетях рунета. Также будут приведены существенные различия ключевых показателей при использовании разных сервисов сбора целевой аудитории.

Обзор сервисов сбора целевой аудитории во ВКонтакте

Ранее исследование подобных вопросов было проведено очень мало, видимо, полагая, что все парсеры работают по одному принципу и отличаться могут только количеством инструментов и, может быть, скоростью обработки заданий. Однако исследовав опыт в таргетированной рекламе, нами был составлен более подробный анализ существующих сервисов для поиска целевой аудитории в социальных сетях.

За последнее время было разработано и предоставлено большое количество парсеров, и в основном внимание исследователей было обращено только на то, какие возможности парсинга сервис предоставляет и сколько стоит его использование. Последним парсером, на котором мы обоснованно и остановили свой выбор, как на самом популярном, был ЦереброТаргет.

Но существует не только один сервис сбора целевой аудитории ВКонтакте и, «про некоторые уникальные инструменты».

Ознакомившись с сервисом, мы открыли один из инструментов «Поиск сообществ по ключевой фразе»), который дал задание поиска сообществ по ключевой фразе «Платья». Сервис довольно быстро справляется с задачами.

На выходе получено 57 000 групп, когда по этой же ключевой фразе Церебро выдаёт 7912.

Результат был сверен с ВКонтакте, который выдал 70 000 групп. Данный сервис выдал практически все группы по платьям, которые

существуют ВКонтакте. Данный сервис отсекает заблокированные группы — потому их меньше.

Повторив выборку с фильтром групп от 500 человек, можно получить чуть более 7 400 пабликов, по которым можно собрать аудиторию, подписанную одновременно минимум на 3 паблика. Сервис выдал нам аудиторию, численностью более чем 3,5 млн человек, против 1,3 млн человек, собранных с почти 3400 пабликов, найденных по таким же параметрам парсером Церебро.

В итоге на тех же самых объявлениях нам удалось добиться пропорционального понижения цены клика, и как результат, стоимость лида тоже упала почти в 3 раза.

Было принято решение исследовать несколько популярных парсеров с целью сравнения количества выдаваемых данных и сбора этих данных в небольшую сводную Таблицу 1.1 и 1.2. Целью являлось получить объективную и наглядную сводку.

Таблица 1.1 и 1.2 содержат данные обо всех доступных в этих сервисах инструментах, цене пользования и скорости выполнения одинаковых задач. Некоторые задачи были усложнены, так некоторые парсеры перестали с ними справляться. На наш взгляд, это важный критерий — насколько парсер способен удовлетворить «капризы», которые периодически у каждого случаются.

Поиск сообществ — это поиск сообществ ВКонтакте, по соответствующей тематике.

Поиск сообществ без точного вхождения — это поиск сообществ ВКонтакте по схожим словам, по соответствующей тематике.

Поиск людей по демографии — это поиск аккаунтов, по заданным возрастным критериям.

Поиск людей по семейному статусу — это поиск аккаунтов, по заданному семейному статусу.

Поиск людей по городу, имени, работе, должности интуитивно

ПОНЯТНЫ.

Обзор и анализ существующих сервисов для поиска целевой аудитории в социальной сети Вконтакте представлен в Таблице 1.1.

Таблица 1.1

Обзор и анализ существующих сервисов для поиска целевой аудитории в социальной сети Вконтакте.

Функции	ЦереброТаргет(стац.)	ЦереброТаргет(серв.)	TargetHunter(серв.)
Поиск сообществ	ключ.фраза Приколы от 50 чел - 1.798 шт	ключ.фраза Приколы от 50 чел 10.733 шт / 30 сек = 358 в сек	ключ.фраза Приколы от 50 чел 24.769 шт / 76 сек = 326 в сек
Поиск сообществ без точного вхождения	ключ.фраза Приколы 3.000 шт	ключ.фраза Приколы 23.641 шт / 30 сек = 788 в сек	ключ.фраза Приколы 603.075 шт / 190 сек = 3.174 в сек
Поиск людей по демографии	есть	нет	есть
Поиск людей по семейному статусу	есть	нет	есть
Поиск людей по городу	город Москва 122.439 шт	нет	город Москва 18.920.877 шт / 305 сек = 62.036 в сек
Поиск людей по Имени	имя Иван 74.128 шт	нет	имя Иван 2.439.639 / 665 сек = 3.669 в сек
Поиск людей по Работе	работа Салон 8.398 шт	нет	работа Салон 118.337 / 45 сек = 2.630 в сек
Поиск людей по Должности	нет	нет	должность Тренер 49.675 / 61 сек = 814 в сек
Поиск новостей	есть	есть	есть
Поиск людей по дню рождения	есть	нет	есть
Поиск пар именинников	есть	нет	есть
Сбор постов сообществ	есть	есть	есть
Сбор опросов	есть	нет	есть
Сбор контактов сообществ	есть	есть	есть
Сбор друзей	в друзьях у 2 и более ЦА SMM 1.276.950 шт	в друзьях у 2 и более ЦА SMM 8.706.465 шт / 3.897 сек = 2.234 в сек	в друзьях у 2 и более ЦА SMM 8.731.610 шт / 519 сек = 16.824 в сек
Сбор родственников	братья и сестры 18.453 шт	нет	братья и сестры 30.344 шт / 20 сек = 1.517 в сек
Сбор подарков	нет	нет	есть
Сбор участников	состоит в 3 и более группах 2.738.519 шт	не справился	состоит в 3 и более группах 2.752.441 шт / 471 сек = 5.844 в сек
Сбор участников встреч	есть	есть	есть
Сбор дней рождения	есть	есть	есть
Сбор пар	вторых половинок 3.354 шт	вторых половинок 3.435 шт / 78 сек = 44 в сек	вторых половинок 5.009 шт / 6 сек = 835 в сек
Сбор почт	нет	нет	есть
Сбор телефонов	нет	нет	есть
Сбор Skype	нет	нет	есть
Сбор Instagram	есть	есть	есть
Сбор Twitter	есть	есть	есть
Сбор Facebook	есть	есть	есть
Сбор LiveJournal	нет	нет	есть
Сбор лайков с сайта	нет	нет	есть
Сбор комментариев	нет	нет	есть
Сбор комментаторов с сайта	есть	есть	есть
Сбор активностей в сообществах	есть	есть	есть
Сбор активностей в фотоальбомах	есть	есть	есть
Сбор активностей в обсуждениях	есть	есть	есть
Сбор активностей в видеоальбомах	нет	нет	есть

Поиск новостей – это поиск новостей Вконтакте по ключевым словам.

Поиск людей по дате рождения логично понятно, поиск пар именинников осуществляет поиск второй половины именинников.

Сбор постов сообществ – осуществляется сбор постов по схожей

тематике по ключевым словам.

Сбор опросов осуществляется по ключевым словам в заголовке опроса.

Сбор контактов сообществ осуществляет сбор по контактными данным сообществ Вконтакте.

Сбор друзей – осуществляется сбор друзей определённой страницы по заданным критериям. Сбор родственников осуществляется аналогично.

Сбор подарков – это сбор одинаковых подарков у пользователей Вконтакте.

Сбор участников осуществляет сбор участников опроса по заданным критериям. Сбор участников встреч – производит сборку по определённой встрече по заданным критериям.

Сбор дней рождений парсит например всех с одной датой рождения и фильтрует по критериям.

Сбор пар – это сбор пар (влюблённых, семейных и др.) по заданным критериям, например возрасту.

Сбор почт, телефонов, скайп, инстаграмм, твиттер , фейсбук, Лайвжурнал производит сбор перечисленных у отобранных контактов по критериям.

Сбор лайков с сайта – это сбор тех кто лайкал, выборка отобранных по критериям.

Сбор комментариев с сайта – это сбор тех кто комментировал, выборка отобранных по критериям.

Сбор активности в сообществах – это сбор тех кто проявлял любую активность, лайкал, комментировал, писал посты в сообществах, выборка отобранных по критериям.

Сбор активностей в фотоальбомах – это сбор тех кто проявлял любую активность, лайкал, комментировал фотографии в определённых альбомах, выборка отобранных по критериям.

Сбор активностей в обсуждениях – это сбор тех кто проявлял любую активность, лайкал, комментировал в комментариях определённого поста,

выборка отобранных по критериям.

Сбор активностей в видеоальбомах – это сбор тех кто проявлял любую активность, лайкал, комментировал видео в определённых видеоальбомах, выборка отобранных по критериям. Обзор и анализ существующих сервисов для поиска целевой аудитории в социальной сети Вконтакте представлен в таблице 1.1.

Таблица 1.2

Обзор и анализ существующих сервисов для поиска целевой аудитории в социальной сети Вконтакте

Функции	SegmentoTarget(серв.)	RetargetSexy(серв.)	PepperNinja(серв.)
Поиск сообществ	ключ.фраза Приколы от 50 чел 10.000 шт / 4 сек = 2,500 в сек	ключ.фраза Приколы от 50 чел 3.000 шт / 3 сек = 1,000 в сек	ключ.фраза Приколы от 50 чел 1.000 шт / 54 сек = 19 в сек
Поиск сообществ без точного вхождения	нет http://joxi.ru/YmEk4nZhZY8k3A	ключ.фраза Приколы 3.000 шт / 3 сек = 1.000 в сек	нет
Поиск людей по демографии	есть	есть	есть
Поиск людей по семейному статусу	есть	есть	есть
Поиск людей по городу	город Москва 10.000.000 шт / 917 сек = 10.905 в сек	нет	город Москва 356.254 шт / 9,216 сек = 39 в сек
Поиск людей по Имени	нет	имя Иван 9.962 / 3 сек = 3.320 в сек	нет
Поиск людей по Работе	нет	работа Салон 7.896 / 5 сек = 1.579 в сек	нет
Поиск людей по Должности	должность Тренер 8.887 / 7 сек = 1.270 в сек	нет	нет
Поиск новостей	нет	есть	нет
Поиск людей по дню рождения	есть	нет	есть
Поиск пар именинников	есть	нет	нет
Сбор постов сообществ	нет	есть	нет
Сбор опросов	нет	нет	нет
Сбор контактов сообществ	есть	есть	есть
Сбор друзей	не справился http://joxi.ru/zAN5KD1slne0y2	в друзьях у 2 и более ЦА SMM 1.000.000 шт / 510 сек = 1.961 в сек	не справился
Сбор родственников	братья и сестры 22.439 шт / 12 сек = 1.870 в сек	братья и сестры 19.070 шт / 40 сек = 477 в сек	нет
Сбор подарков	нет	нет	нет
Сбор участников	не справился http://joxi.ru/vAWYk3GskZ9o3m	состоят в 3 и более группах 1.000.000 шт / 490 сек = 2.041 в сек	не справился
Сбор участников встреч	есть	есть	есть
Сбор дней рождения	есть	есть	нет
Сбор пар	вторых половинок 3.418 шт / 7 сек = 488 в сек	вторых половинок 3.515 шт / 20 сек = 176 в сек	нет
Сбор почт	нет	нет	нет
Сбор телефонов	нет	нет	нет
Сбор Skype	нет	нет	нет
Сбор Instagram	есть	есть	нет
Сбор Twitter	нет	есть	нет
Сбор Facebook	нет	есть	нет
Сбор LiveJournal	нет	нет	нет
Сбор лайков с сайта	нет	нет	есть
Сбор комментариев	нет	нет	нет
Сбор комментаторов с сайта	нет	нет	есть
Сбор активностей в сообществах	есть	есть	есть
Сбор активностей в фотоальбомах	есть	есть	есть
Сбор активностей в обсуждениях	есть	есть	есть
Сбор активностей в видеоальбомах	нет	есть	есть

Полная таблица исследования и по бесплатным парсерам, и по парсерам в стадии beta-теста представлена в Приложении 1. Данная таблица содержит строки с вычислением средних баллов по результативности и

скорости работы парсеров, а также общий балл выгоды покупки по соотношению цены к количеству доступных функций.

Таблица из Приложения 1 содержит 15 сервисов с полным разбором.

Анализ вышеуказанных сервисов не позволяет решить одну простую проблему, а именно не даёт возможности начинающему рекламщику или старперу собрать целевую аудиторию, так как стоимость услуг у данных сервисов не подъемна для большинства начинающих предпринимателей в интернете. В связи с этим, мы предлагаем разработать веб-сервис, который будет выполнять основные функции парсинга целевой аудитории по группам Вконтакте, фильтровать результаты и формировать список аккаунтов для дальнейшей настройки персонализированной таргетированной рекламы.

Это привлечёт в свою очередь большой поток посетителей, что в свою очередь позволит разместить контекстную рекламу и получить доходы за счёт неё.

Таким образом, были проанализированы существующие сервисы для поиска целевой аудитории в социальных сетях. Самым целесообразным сервисом по сбору целевой аудитории во Вконтакте в настоящий момент является сервис «TargetHunter». Однако все рассмотренные сервисы имеют один большой недостаток для начинающих предпринимателей в интернете, это высокая стоимость услуг данных веб-сервисов. Это обуславливает разработку веб-сервиса с минимальными основными функциями парсинга целевой аудитории по группам Вконтакте, что позволит фильтровать результаты и формировать список аккаунтов.

1.3 Требования к разработке веб-сервиса

Любой сервис в сети интернет относится к веб-сервисам, в том числе и парсеры и сборщики целевой аудитории из групп социальных сетей.

Требования к разработке и использованию интерфейсов важнейший аспект разработки любого веб-сервиса.

Интерфейс информационной системы, подключаемой к системе взаимодействия, должен быть реализован в виде веб-сервиса.

Программно-аппаратные средства обеспечения защищённой интеграции информационных систем с системой взаимодействия должны обеспечивать выполнение настоящих требований.

Применяемые при разработке и использовании интерфейсов технологии, стандарты и спецификации должны соответствовать государственным стандартам Российской Федерации, иным нормативно установленным и общепризнанным стандартам и требованиям в области информационных технологий и программного обеспечения.

Требования к протоколам сетевого взаимодействия являются частью требований к веб-сервису.

При использовании сетевых протоколов передачи данных необходимо придерживаться следующих спецификаций:

- протокол передачи гипертекста HTTP v. 1.1 – стандарт RFC 2616 инженерной группы проектировщиков информационно-телекоммуникационной сети Интернет (Internet Engineering Task Force, IETF);

- модернизированный протокол http v. 1.1 с обеспечением безопасности транспортного уровня (TLS) для существующего протокола управления передачей (TCP);

- протокол защищённых соединений SSL v. 3 / TLS – стандарт RFC 5246 инженерной группы проектировщиков информационно-телекоммуникационной сети Интернет (Internet Engineering Task Force, IETF);

- набор протоколов IP Security для обеспечения защиты данных – стандарты RFC 4301, 4302, 4835, 2403, 2404, 2405, 4303, 4835, 5996, 2410, 2411, 2412 инженерной группы проектировщиков информационно-телекоммуникационной сети Интернет (Internet Engineering Task Force, IETF);

- сервисы поддержки пространства имен DNS (Domain Name Services) – стандарт RFC 1035 инженерной группы проектировщиков информационно-телекоммуникационной сети Интернет (Internet Engineering Task Force, IETF).

Требования к протоколам веб-сервисов являются одним из составляющих общих требований к сервису.

При разработке веб-сервисов необходимо придерживаться следующих спецификаций:

- базовый профиль интероперабельности v. 1.1 – стандарт Организации по интероперабельности веб-сервисов – спецификация носит обязательный характер[3];

- профиль передачи веб-сервисами бинарных приложений (WS-I Attachments Profile 1.0) – стандарт Организации по интероперабельности веб-сервисов WS-I[4] – спецификация носит рекомендательный характер;

- профиль передачи веб-сервисами бинарных приложений (SOAP Message Transmission Optimization Mechanism) – стандарт консорциума W3C[5] – спецификация носит рекомендательный характер;

- профиль связывания веб-сервисов (WS-I Simple SOAP Binding Profile 1.0) – стандарт организации по интероперабельности веб-сервисов[6] WS-I – спецификация носит рекомендательный характер;

- протокол SOAP 1.1 – стандарт консорциума W3C[7] – спецификация носит обязательный характер;

- язык описания веб-сервисов (Web Services Description Language, WSDL 1.1) – стандарт консорциума W3C[8] – спецификация носит обязательный характер;

- политика использования веб-сервисов (Web Services Policy 1.2) – проект рекомендации консорциума W3C[9] – спецификация носит рекомендательный характер;

- спецификация универсального описания, обнаружения и интеграции веб-сервисов UDDI v. 3.0 (Universal Description Discovery and Integration) –

стандарт Организации по развитию стандартов структурированной информации[10] – спецификация носит рекомендательный характер;

- спецификация универсального описания, обнаружения и интеграции веб-сервисов UDDI v. 2.0 (Universal Description Discovery and Integration) –

стандарт Организации по развитию стандартов структурированной информации[11] – спецификация носит рекомендательный характер.

При описании данных, метаданных и используемых наборов символов, применяемых в процессе информационного обмена, необходимо придерживаться следующих спецификаций:

- расширяемый язык разметки XML (Extensible Markup Language) – в соответствии с набором стандартов консорциума W3C[12];

Существуют несколько типов требований:

1) функциональные требования – описаны в С-требованиях (См. 4.1 С-требования).

2) Нефункциональные требования:

– производительность – программа должна мгновенно реагировать на действия пользователя.

– надежность и безопасность – программа должна обеспечивать надежность и секретность хранения и передачи информации.

– обработка ошибок – в случае ошибки программа должна предложить пользователю отправить отчет.

– интерфейсные требования – для связи с БД используется SQL контроллер. SQL выбран как уникальное средство запросов к БД.

– ограничения – программа должна быть написана на PHP. Это позволит создать приложение, удовлетворяющее требованиям заказчика и временным ограничениям.

Разработку веб-сервиса следует выполнять в соответствии с действующими стандартами по действующим актуальным ГОСТам.

ГОСТ 19.201-78[13]. Единая система программной документации (ЕСПД) устанавливает порядок построения и оформления технического

задания на разработку программы или программного изделия для вычислительных машин, комплексов и систем независимо от их назначения и области применения. От него следует отталкиваться при проектировании нашего программного продукта.

Техническое задание оформляют в соответствии с ГОСТ 19.106-78[14] на листах формата 11 и 12 по ГОСТ 2.301-68[15].

Для внесения изменений или дополнений в техническое задание на последующих стадиях разработки программы или программного изделия выпускают дополнение к нему. Согласование и утверждение дополнения к техническому заданию проводят в том же порядке, который установлен для технического задания.

Перейдём к составлению технического задания веб-сервиса по парсингу клиентов ВК.

1. Введение

1.1 Наименование программы: "VK Hunter"

2. Основания для разработки

Основанием для разработки является учебный план БелГУ факультета ИТ кафедры ИТ утвержденный заведующим кафедрой.

3. Назначение разработки;

Программный продукт предназначен для получения в социальной сети ВК данные о пользователях. Программа представляет собой веб-сервис и GUI-интерфейс. Применяя различные фильтры и критерии для выборки данных можно получить список потенциальных клиентов с различными интересами и качествами с целью дальнейшего использования этой информации для эффективной настройки таргетированной рекламы.

4. Требования к программе или программному изделию

4.1. Требования к функциональным характеристикам

- Возможность выбора страны для парсинга (фильтрация)
- Возможность выбора города для парсинга (фильтрация)
- Возможность выбора пола для парсинга (фильтрация)

- Возможность выбора семейного статуса для парсинга (фильтрация)
- Возможность выбора возраста для парсинга (фильтрация)
- Возможность выгрузки списка спарсенных аккаунтов из соцсети

Вконтакте в формате csv

4.1.1 Что должен делать сервис

1. Искать пользователей социальной сети VK.com с помощью формы поиска. На выходе можно посмотреть найденных пользователей в 2-х видах, просмотр как “Карточка” и “Таблица”. В таблице отображается чуть больше полей чем в карточке.

2. Производить поиск пользователей применяя дополнительные фильтры.

3. Выгружать найденных пользователей в файл формата CSV.

4. Искать группы социальной сети VK.com с помощью формы поиска с применением различных фильтров. На выходе можно посмотреть найденные группы в 2-х видах, просмотр как “Карточка” и “Таблица”. В таблице отображается чуть больше полей чем в карточке.

5. Производить поиск групп применяя дополнительные фильтры.

6. Выгружать найденные группы в файл формата CSV.

7. С помощью ID группы или с помощью короткого имени группы должен получать подробные параметры группы. В параметрах, например, будет такое поле как количество подписчиков.

8. Выгружать участников указанной группы в файл формата CSV.

4.2.1. Требования к обеспечению надежного функционирования программы

Надежное (устойчивое) функционирование программы должно быть обеспечено выполнением Заказчиком совокупности организационно-технических мероприятий, перечень которых приведен ниже:

- 1) организацией бесперебойного питания технических средств:
- 2) использованием лицензионного программного обеспечения:
- 3) регулярным выполнением рекомендаций Министерства труда и социального развития РФ. изложенных в Постановлении от 23 июля 1998 г.

Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств»;

4) регулярным выполнением требований ГОСТ 51188-98. Защита информации. Испытания программных средств на наличие компьютерных вирусов.

4.2.2. Время восстановления после отказа

Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбоем (не крахом) операционной системы, не должно превышать 10-ти минут при условии соблюдения условий эксплуатации технических и программных средств.

Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем (крахом) операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановки программных средств.

4.2.3. Отказы из-за некорректных действий пользователей программы

Отказы программы вследствие некорректных действий пользователя при взаимодействии с программой через веб-интерфейс недопустимы.

4.3. Условия эксплуатации

Стандартные условия эксплуатации программных продуктов. Минимальное количество пользователей, требуемых для работы программы, должно составлять не менее 1 штатной единицы - пользователь программы (клиент).

4.4. Требования к составу и параметрам технических средств

Для нормальной работы программы необходимо:

4.4.1. Процессор Intel(R) Xeon(R) CPU E5620 @ 2.40GHz (1 core)

4.4.2. ОЗУ не менее 150 Мб

4.4.3. Не менее 100 Мб на жёстком диске

4.4.4. Интернет канал со скоростью от 1 мб/с для удобной работы

4.5. Требования к информационной и программной совместимости

Для нормальной работы программы необходимо:

4.5.1. Программа должна работать под управлением UNIX-совместимой ОС серверов.

4.5.2. Apache 2.4.6-67.el7.centos.6

4.5.3. PHP 5.4.16-43.el7_4.1

4.5.4. Perl 5.16.3-292.el7

4.6. Специальные требования

Простота интерфейса пользователя.

5. Требования к программной документации;

Документация должна быть написана в общедоступном формате, на языках пользователей сервиса и социальной сети Вконтакте с соблюдением ГОСТов.

5.1. Предварительный состав программной документации

Состав программной документации должен включать в себя:

5.1.1. Техническое задание

5.1.2. Листинг программного кода

5.1.3. Руководство пользователя

5.1.4. Инструкция по установке программы

6. Порядок контроля и приемки;

Испытания представленной программы и контроль качества ее работы провести на сервере предоставленного компанией 7-host.ru. Во время испытаний проверить работу системы по следующим позициям:

1) Веб-сервис должен корректно проходить весь набор функциональных тестов;

2) корректное функционирование заданных в ТЗ функций;

3) возможность функционирования на сервере с указанными минимальными системными требованиями;

4) демонстрация возможности установки на указанную систему.

Таким образом, в данном разделе были рассмотрены требования к

разработке веб-сервиса по современным ГОСТам. Было составлено техническое задание на разработку веб-сервиса. Для разработки веб-сервиса сбора целевой аудитории из социальной сети Вконтакте были выдвинуты основные требования работоспособности.

2. ПРОЕКТИРОВАНИЕ И ПРОГРАММНАЯ РЕАЛИЗАЦИЯ СЕРВИСА

2.1 Разработка объектно-ориентированной модели программного обеспечения

Для получения качественного Web-сайта, сначала необходимо спроектировать все необходимые действия. Это обязательное условие для создания сайта с понятной структурой и хорошей навигацией, но даже при этом цель создания сайта может быть не достигнутой. Если же действовать не по плану, то вероятность создания никому не понятного проекта очень велика.

Если не разрабатывать подробный план действий, то, скорее всего, при создании будут наблюдаться “скачки” между различными этапами: от написания кода – к разработке архитектуры, от разработки архитектуры – к описанию концепции, от описания концепции – назад к написанию кода. В результате появляется риск потери логики сайта во время этих “скачков”. То есть посетители в будущем могут не понять, для чего вообще создавался этот Web-проект. Кроме того, постоянные переходы между этапами способны значительно ухудшить качество результата работы, особенно это утверждение верно в отношении навигации сайта. “Прыжки” между этапами будут приводить к тому, что разработчикам придется по несколько раз переделывать одну и ту же работу. А это негативно скажется как на сроки изготовления проекта, так и на его стоимости.

Для избегания этой проблемы, достаточно заранее спроектировать все действия и четко следовать по полученному плану. Сегодня существует немало стандартных моделей проектирования, которые позволяют поэтапно, шаг за шагом, реализовать любой проект от идеи до ее воплощения.

Их выбор зависит только от разработчиков и тех целей, которые они преследуют. Примерами таких моделями являются: «Каскадная модель», модель «Спираль» и одной из примечательных таких моделей для проектирования сайта является модель Уолта Диснея, она состоит из трех этапов: концептуальное проектирование; логическое проектирование; физическое проектирование.

Этапы следуют последовательно один за другим, но в некоторых случаях возможен переход к следующей стадии без окончания предыдущей. Это может происходить, например, когда есть несколько разработчиков, каждый из которых работает со своей частью сайта. В любом случае, после окончания этапа физического проектирования следует вернуться к началу и внести соответствующие коррективы.

Концептуальное проектирование. Порой бывает сложно оценить эффективность сайта. Необходимо знать и понимать критерии оценки для того, чтобы определить хорошим или плохим является разработанный ресурс.

Есть универсальный критерий, который довольно точно характеризует эффективность сайта – это достижение разработчиками сайта поставленных перед ними целей. В этом случае сайт превращается в качественный инструмент, который выполняет возложенные на него функции.

Концептуальное проектирование служит для указания целей, задач сайта и определения аудитории, на которую он рассчитан.

На этом этапе проектирования опишем следующее:

1. Основные и второстепенные цели разработки веб-сервиса .

Основная цель - Разработка веб-сервиса для для получения в социальной сети VK данных о пользователях. Программа представляет собой веб-сервис и GUI-интерфейс. Применяя различные фильтры и критерии для выборки данных можно получить список потенциальных клиентов с различными интересами и качествами с целью дальнейшего использования этой информации для эффективной настройки таргетированной рекламы.

Второстепенные цели.

- 1) Контакты групп. Например, вставляя ссылку на группы, на выходе получать список администраторов групп
 - 2) Участники групп. Вставляя ссылку на группы, на выходе получать список всех участников
 - 3) Поиск групп по ключевой фразе. Например "бизнес" и должно выдавать все группы Вконтакте, где есть эта фраза.
 - 4) Фильтрация. Когда имеется список групп – возможность отфильтровать по критериям.
 - 5) Фильтрация пользователей.
2. Действия, которые необходимо предпринять для достижения поставленных целей.
- Разработать веб-сервис.
3. Аудиторию сайта.
- Маркетинговые агенства и интернет-предприниматели
4. Интересы групп пользователей.
- Основной интерес это привлечение клиентов к товарам и услугам, путём сбора целевой аудитории и последующей рекламной компанией.
5. Разделы сайта.
- Разделов нет, сайт одностраничник.
6. Критерии достижения цели.
- Разработать удобный и понятный интерфейс веб-сервиса, работоспособность которого будет проверена тестами.

После определения поставленных целей и интересов пользователей, можно составить список функций сервиса, которые будут располагаться на сайте, что представлено на рисунке 2.1.



Рис. 2.1 – Функции веб-сервиса

Поведенческое моделирование.

На рисунке 2.2 представлена схема общего принципа работы клиентской части, серверной части и API VK.



Рис. 2.2 - Схема общего принципа работы клиентской части, серверной части и API VK.

Интерфейс пользователя получает данные от пользователя и отражает данные с помощью клиентской части javascript.

Клиентский javascript получает данные от аjax запросов к серверу и отражает данные в интерфейсе пользователя.

AJAX запросы получают данные для запроса от клиентского javascript и отправляет их на сервер, после получения ответа от сервера AJAX запрос отправляет ответ от сервера клиентскому javascript

Сервер получает данные для запроса к API VK от аjax запроса и делает запрос к API VK, после получения ответа от API VK сервер отправляет ответ

AJAX запросу. API VK получает данные для поиска от сервера и отправляет найденные данные серверу.

На рисунке 2.3 представлена схема поведенческой работы интерфейса с пользователем и сервисом.

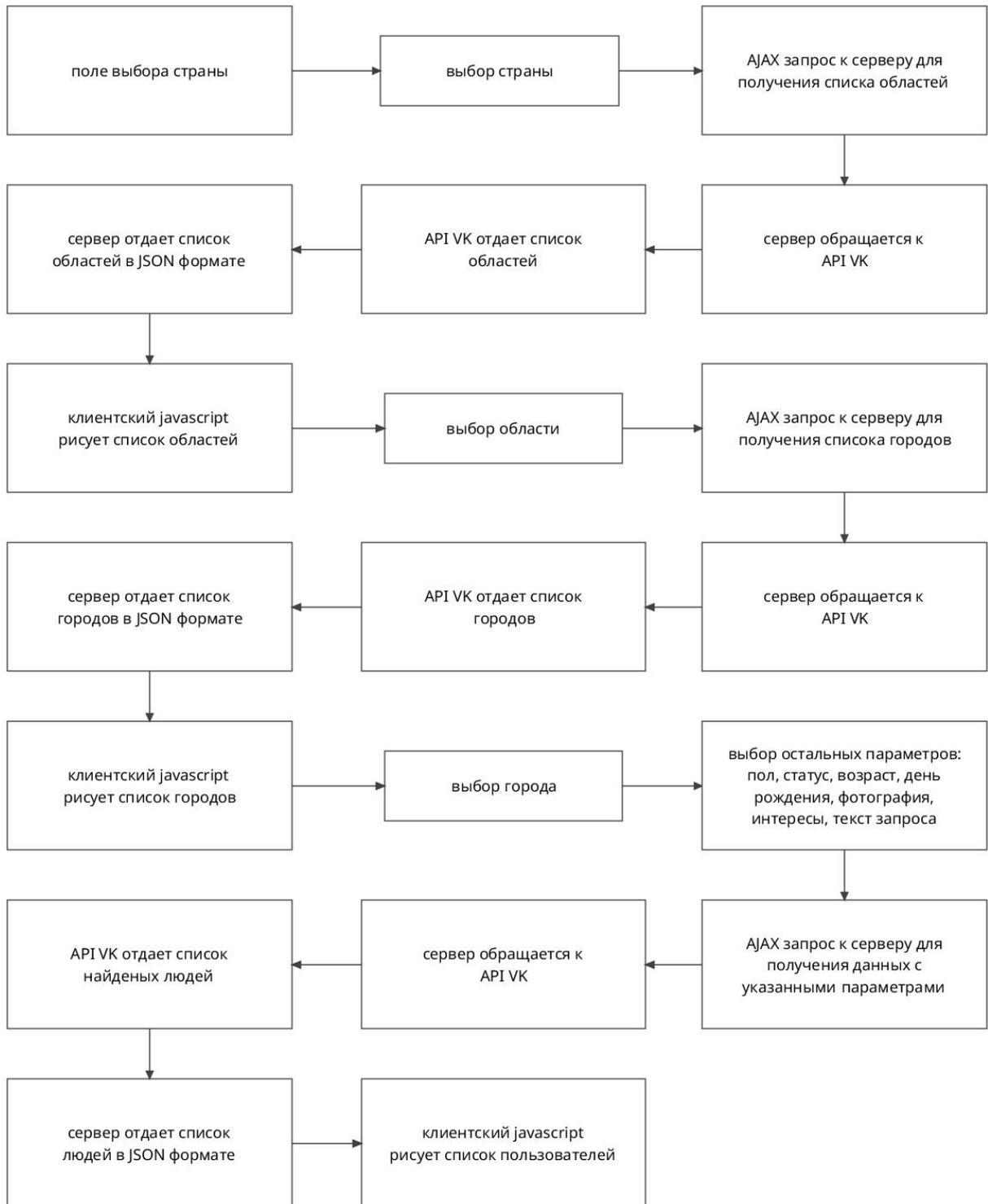


Рис. 2.3 - Схема поведенческой работы интерфейса с пользователем и сервисом

Логическое проектирование. Определенные разделы сайта, на предварительном этапе, пока не упорядочены и не структурированы, поэтому их нужно привести к удобному и понятному виду.

Логическое проектирование включает организацию информации на сайте, построения ее структуры и навигации по разделам.

На данном этапе следует задаться вопросом, каким образом будет упорядочена информация. Одновременное использование различных способов охватывает большую аудиторию и позволяет быстрее найти нужную информацию на сайте.

Логическая схема регистрации сервиса в API vk.com представлена на Рисунке 2.4.

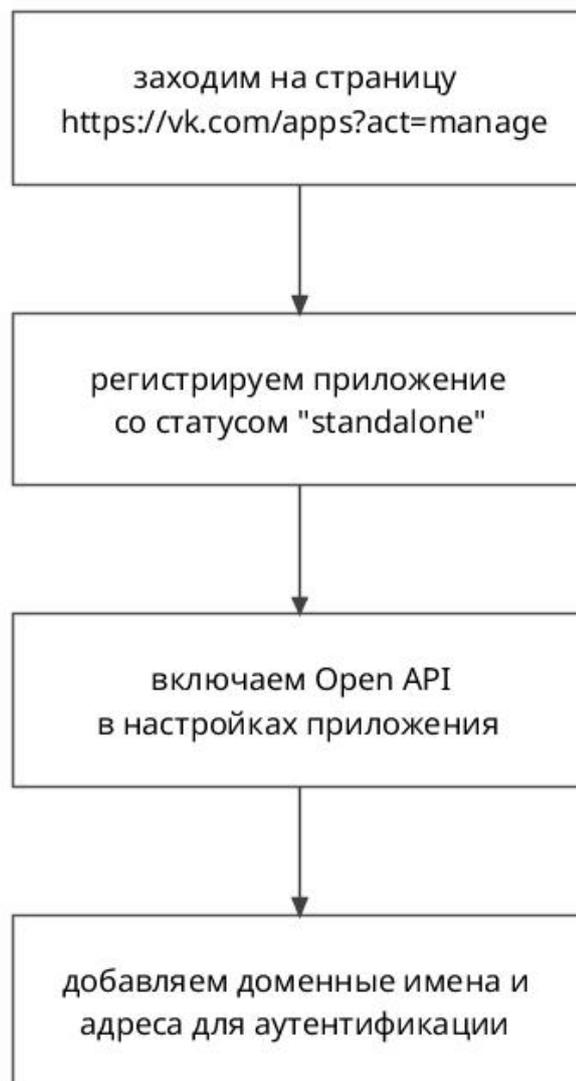


Рис. 2.4 - Логическая схема регистрации сервиса в API vk.com

На этом этапе опишем следующее:

1. Тип структуры сайта.

Тип структуры веб-сервиса линейный, сайт не использует базу данных и является одностраничником.

2. Названия разделов.

Разделы отсутствуют.

3. Какая информация будет размещена на определенных страницах сайта.

На главной и единственной странице будет краткое описание основной функции сервиса и минимальный дизайн.

Конечным результатом логического проектирования является блок-схема или структурная диаграмма, показывающие взаимосвязь различных частей сайта, которая не целесообразна в отображении, так как взаимосвязи веб-сервиса отсутствуют.

Схема логической работы программной части службы на стороне сервера представлена на рисунке

Физическое проектирование. Данный этап связан с поиском проблем, а не их решений, связанных в основном с технической реализацией сайта.

На этом этапе следует описать следующее: технологии, которые будут применяться на сайте; программное обеспечение, используемое; возможные проблемы и способы их устранения; как будет обновляться информация.

После завершения этого этапа следует вернуться к концептуальному проектированию и проверить, не нужно внести изменения, в связи с переосмыслением проекта на других стадиях.

Схема работы сервиса в отношении “Сервис” – “API” представлена на рисунке 2.5

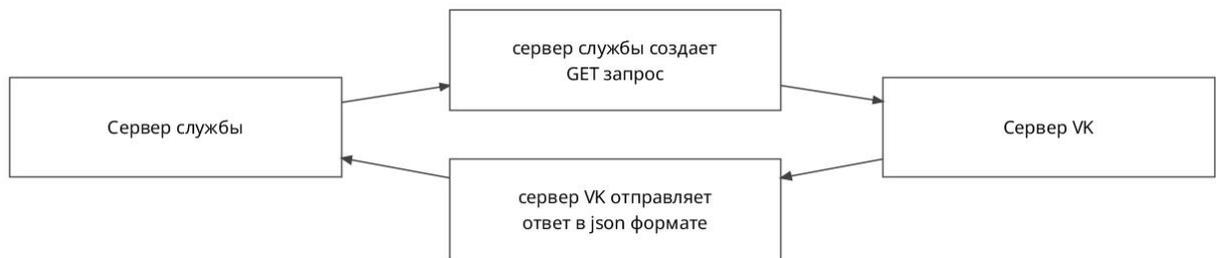


Рис. 2.5 - Диаграмма работы сервиса в отношении “Сервис” – “API”

Схема работы программной части службы на стороне сервера представлена на Рисунке 2.6

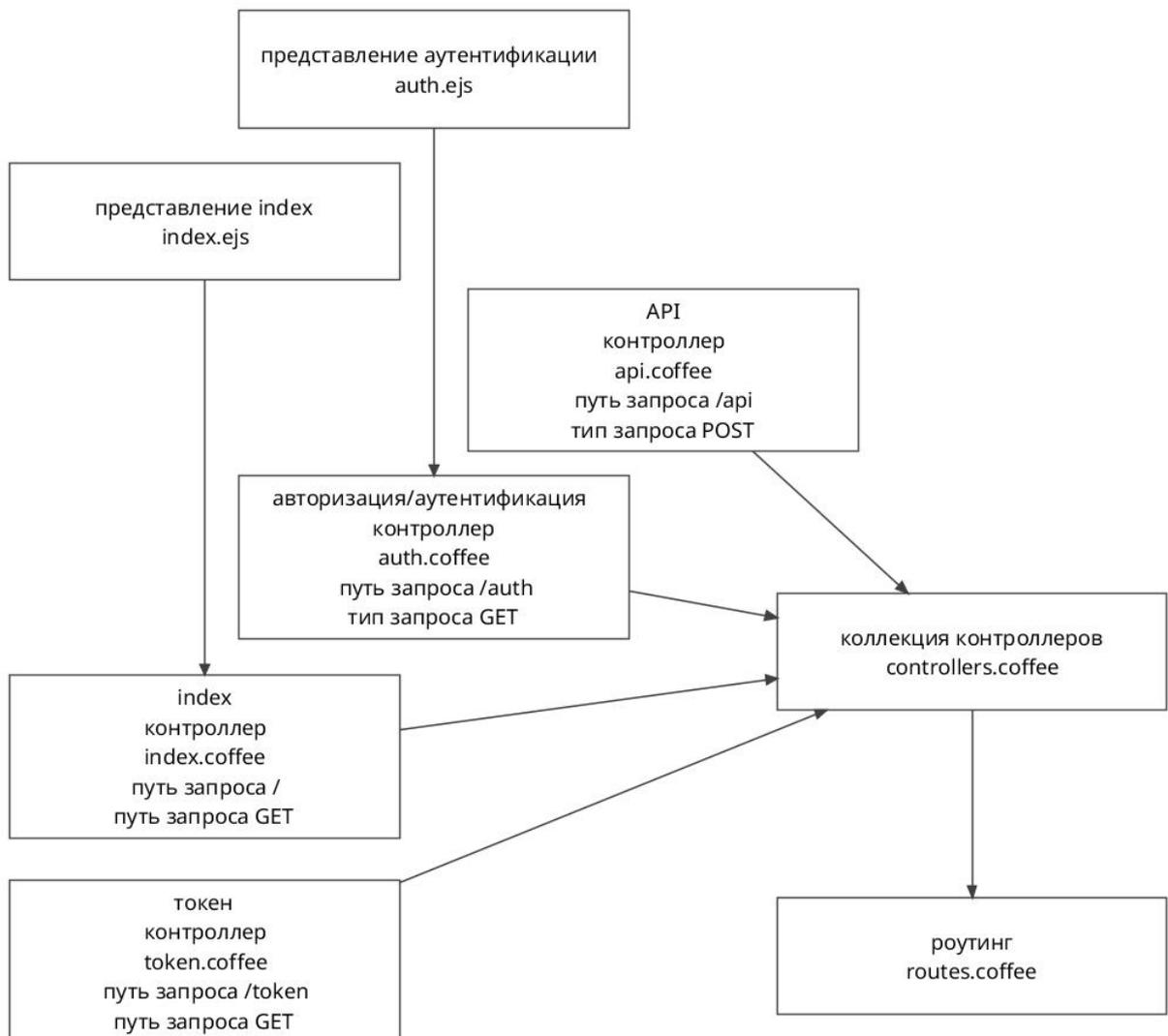


Рис. 2.6 - Схема физической работы программной части службы на стороне сервера

В результате проектирования физических требований показана программная часть, представлена в виде диаграммы use-case на Рисунке 2.7.

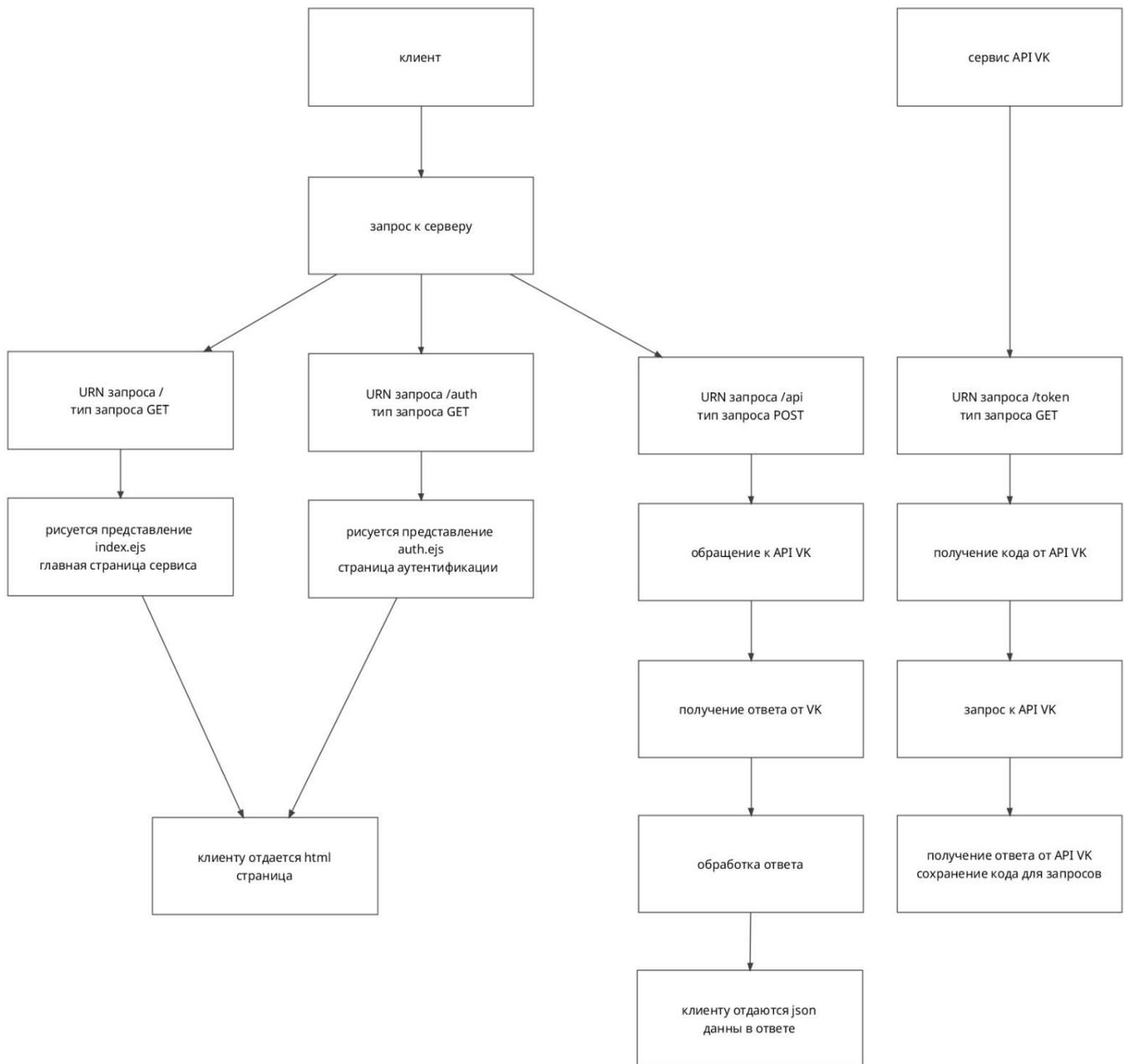


Рис. 2.7 – Активити диаграмма веб-сервиса

- 1.1 Представление index index. js
- 1.2 Index контроллер index.coffee путь запроса / путь запроса GET
- 1.3 Токен контроллер token.coffee путь запроса /token путь запроса GET
- 2.1 Представление аутентификации auth.js
- 2.2 Авторизация/аутентификация контроллер auth.coffee путь запроса /auth тип запроса GET
3. API контроллер a pi.coffee путь запроса /api тип запроса POST
4. Коллекция контроллеров control lers.coffee

5. Роутинг routes.coffee

Интерфейс приложения интеллектуально понятен пользователю. При входе в систему появится форма ввода, на которой в текстовых полях, предназначенных для ввода, вводятся параметры для парсинга групп и пользователей из соцсети Вконтакте. После того как посетитель ввёл соответствующую информацию в поля и нажал кнопку поиска, веб-сервис обрабатывает запрос и обращается к соцсети Вконтакте через API, данные полученные через API выводятся на сайт.

Всё программное обеспечение функционирует и работает на основе одного виртуального сервера (VPS) на стороне провайдера услуг аренды VPS серверов.

Доступ к серверу осуществляется по средствам сетевого доступа через сети предоставляемые провайдером услуг аренды VPS серверов.

Описание серверной части

Серверная часть программы написана на основе платформы Node.js и фреймворка Express.js.

Node.js это программная платформа, основанная на движке V8 (транслирующем JavaScript в машинный код), превращающая JavaScript из узкоспециализированного языка в язык общего назначения. Node.js добавляет возможность JavaScript взаимодействовать с устройствами ввода-вывода через свой API (написанный на C++), подключать другие внешние библиотеки, написанные на разных языках, обеспечивая вызовы к ним из JavaScript-кода.

Node.js применяется преимущественно на сервере, выполняя роль веб-сервера, но есть возможность разрабатывать на Node.js и десктопные оконные приложения (при помощи NW.js, AppJS или Electron для Linux, Windows и macOS) и даже программировать микроконтроллеры (например, tessel и espruino).

В основе Node.js лежит событийно-ориентированное и асинхронное (или реактивное) программирование с неблокирующим вводом/выводом.

Express.js или просто Express, каркас web-приложений для Node.js, реализованный как свободное и открытое программное обеспечение под лицензией MIT. Он спроектирован для создания веб-приложений и API. Де-факто является стандартным каркасом для Node.js при создании программного обеспечения на стороне веб сервера.

Вся файлы и папки размещаются в одной корневой папке проекта.

В качестве языка программирования проекта применяется Coffeescript. Coffeescript - язык программирования, транслируемый в JavaScript. CoffeeScript добавляет синтаксический сахар для того, чтобы улучшить читаемость кода и уменьшить его размер. CoffeeScript позволяет писать более компактный код по сравнению с JavaScript. JavaScript-код, получаемый трансляцией из CoffeeScript, полностью проходит проверку JavaScript Lint.

То есть код серверной части проекта написан с помощью языка программирования Coffeescript.

Так же вместе с node.js применяются дополнительные модули. Список модулей и их назначение указаны ниже.

node-vkapi - простой вызов всех существующих методов API ВКонтакте. По сути это просто обёртка для создания запросов к api.

coffee-script - модуль-компилятор coffeescript в javascript. Модуль позволяет запускать coffeescript код без предварительной компиляции.

gulp - модуль-сборщик. Gulp с помощью созданных предварительных инструкций собирает файлы. Например файлы в формате sass компилирует в css. Или например файлы в формате coffee компилирует в javascript. Обычно сборщик применяется для сборки кода и вёрстки для клиентской части приложения (frontend).

mocha - модуль для написания тестов.

chai - модуль для написания тестов.

Перечислены ключевые модули. Всего модулей применяется больше, и все они перечислены в специальном файле package.json.

Файл `Package.json` необходим для установки пакетов с помощью программы управления пакетами NPM.

Npm - менеджер пакетов, входящий в состав Node.js. Это самый удобный способ для установки дополнительных модулей.

Структура файлов и папок имеет такой вид, представленный в Приложении 2. Описание назначения файлов:

`index.coffee` - основной файл с веб сервера. Основная точка запуска

`config.coffee` - файл с настройками проекта

`controllers.coffee` - файл со списком контроллеров

`controllers` - папка с контроллерами

`gulpfile.coffee` - файл с инструкциями для сборщика `gulp`

`helpers` - папка с мини-функциями помощниками

`logs` - папка с логами обращений к серверу

`package.json` - файл для npm системы управления модулями

`prod` - публичная папка файлы, из которой отдаются в клиентскую часть

`routes.coffee` - файл с настройками роутинга. В нём описываются, какие запросы поддерживает сервер, и, какие функции будут на эти запросы реагировать.

`views` - папка с разметкой представлений. В данной папке содержатся шаблоны для представлений ответов сервером.

`vkapi.json` - файл с сохраненными ключами доступа к VK API. В данном проекте не применяется никакая база данных, а потому некоторые динамические настройки сохраняются просто в файл в формате JSON.

Описание клиентской части проекта

Основу клиентской части составляют файлы и папки в каталоге `dev`, структура которых представлена в Приложении 3.

Описание папок и файлов в каталоге `DEV`.

В каталоге `dev` лежат файлы и папки которые будут собраны с

помощью инструмента автоматизации Gulp. Целью сборки является компиляция и копирование файлов из среды разработки в публичную среду.

coffee/script.coffee - файл содержит клиентский код. Этот код делает проект живым, реагирует на нажатие кнопок, отрисовку результатов и так далее.

js/ - тут содержатся файлы библиотеки, которые применяются в клиентской части сайта. Например, bootstrap и jquery. JQuery применяется, потому что он требуется библиотеки bootstrap.

sass - файлы каскадных стилей в формате SASS. Эти файлы компилируются в CSS. Sass Sass это метаязык на основе CSS, предназначенный для увеличения уровня абстракции CSS кода и упрощения файлов каскадных таблиц стилей. SASS отличается отсутствием фигурных скобок, в нём вложенные элементы реализованы с помощью отступов.

Основной принцип сборки проекта.

В терминале запускается команда:

```
npm run c
```

В итоге с помощью модуля gulp и заранее подготовленных инструкций в файле gulpfile.coffee проект собирается в папку prod.

После можно запустить сервер с помощью команды:

```
npm run s
```

После запуска этой команды запускается сервер

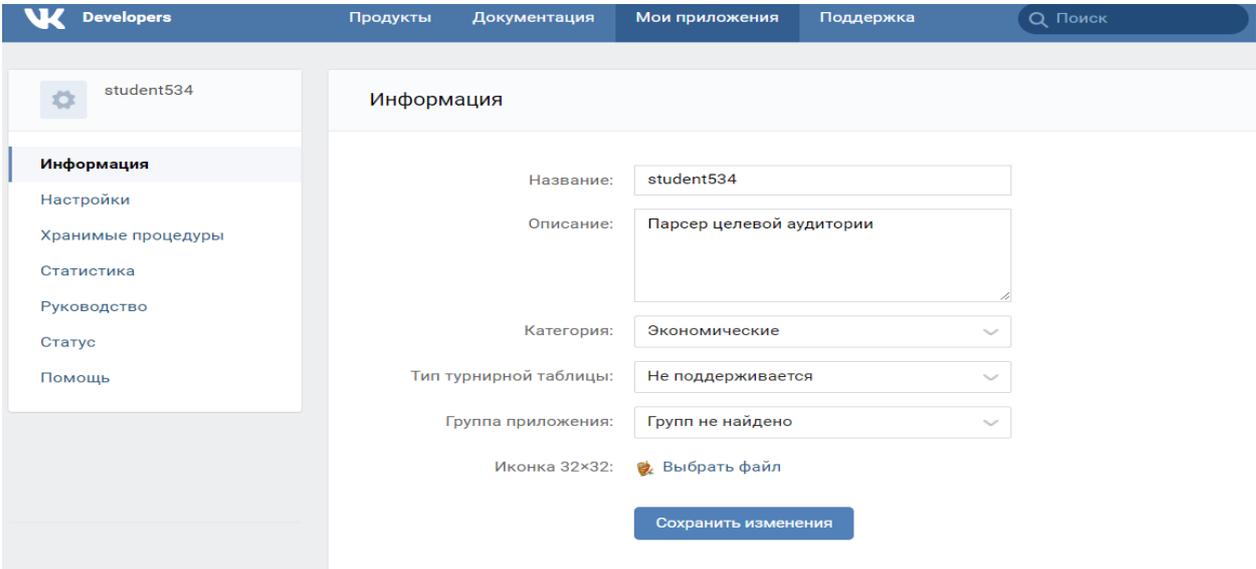
Таким образом, в данном разделе мы описали сервис с 3-х разных сторон. Провели физическое, логическое и поведенческое проектирование веб-сервиса для поиска целевой аудитории Вконтакте. Веб-сервис смоделирован на схемах, на диаграмме показаны функции системы, кому предоставляются возможности и как взаимодействуют с системой.

2.2. Технология получения доступа к информации из социальной сети «ВКонтакте»

Для начала следует перейти к созданию приложения в социальной сети ВКонтакте.

Для этого следует перейти в раздел Developers в раздел приложений. Далее следует обозначить название, описание и категорию приложения.

Также дополнительно можно заполнить остальные поля и выбрать иконку приложения, как это представлено на Рисунке 2.8.



The screenshot displays the 'Информация' (Information) settings for a VK application. The interface includes a top navigation bar with 'Developers' and a search bar. A left sidebar shows the user profile 'student534' and a menu with options like 'Информация', 'Настройки', and 'Статус'. The main content area contains the following fields:

- Название: student534
- Описание: Парсер целевой аудитории
- Категория: Экономические
- Тип турнирной таблицы: Не поддерживается
- Группа приложения: Групп не найдено
- Иконка 32x32: Выбрать файл

A 'Сохранить изменения' (Save changes) button is located at the bottom of the form.

Рис. 2.8 – Основные параметры приложения ВКонтакте

В настройках получить Защищённый ключ и Сервисный ключ доступа. Последним следует сменить состояние на включено, когда всё будет готово, как это представлено на Рисунке 2.9.

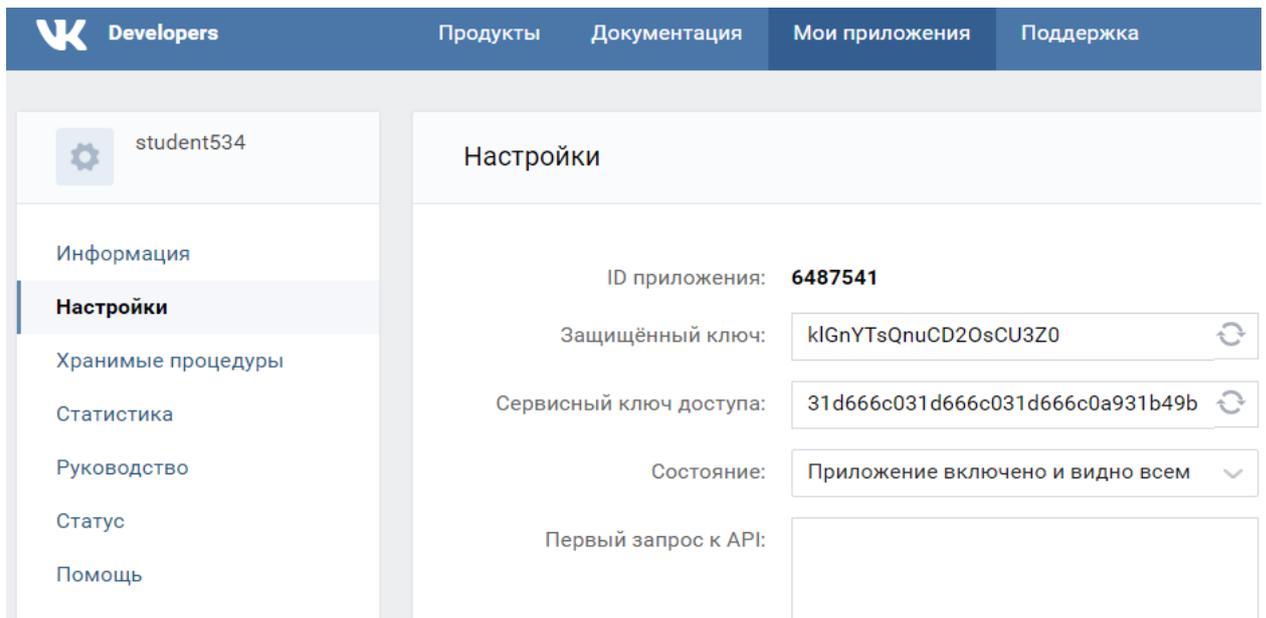


Рис. 2.9 – Настройки приложения ВКонтакте

API (программный интерфейс приложения, интерфейс прикладного программирования) (англ. application programming interface, API [эй-пи-ай]) — набор готовых классов, процедур, функций, структур и констант, предоставляемых приложением (библиотекой, сервисом) или операционной системой для использования во внешних программных продуктах. Используется программистами при написании всевозможных приложений.

В разработке веб-сервиса были использованы следующие методы работы с API ВКонтакте:

Полное описание всех методов API представлено по ссылке 2.

Ниже приводятся все методы для работы с данными ВКонтакте.

Как информационное обеспечение мы использовали «node-vkapi» .

Установку произвели командой в SSH сервера, на котором базируется наш веб-сервер Apache и веб-сервис.

```
npm install node-vkapi --only=prod
```

Возможности «node-vkapi»:

- Простой вызов всех существующих методов API ВКонтакте
- Авторизация пользователя и получение токена

- Прямая авторизация через официальное приложение (Android, iPhone)
- Авторизация через Web-версию сайта
- Загрузка файлов любого типа
- Разгадывание капчи с помощью стороннего сервиса

Пример использования «node-vkapi» представлен в листинге 2.1

Листинг 2.1

```
const vkapi = new (require('node-vkapi'))();
// Получение некоторых данных о пользователе id1
// и вывод их в консоль.
vkapi.call('users.get', {
  user_ids: '1',
  fields: 'verified,sex'
})
.then(users => console.dir(users[0]))
.catch(error => console.error(error));
```

Документация.

`new VkApi([options])`

`options<Object>` Опции экземпляра `VkApi`

Options

Свойства объекта `options` и их значения по умолчанию:

```
accessToken: null, // <String> Ключ доступа
apiVersion: '5.68', // <String> Версия API
appId: null, // <Number> ID приложения ВКонтакте
appSecret: null, // <String> Секретный ключ приложения ВКонтакте
captchaApiKey: null, // <String> API ключ сервиса по распознаванию капчи
captchaService: 'anti-captcha', // <String> Сервис по распознаванию капчи (anti-
captcha, antigate, rucaptcha)
userLogin: null, // <String> Логин пользователя
userPassword: null, // <String> Пароль пользователя
baseDelay: 334 // <Number> Базовая задержка между вызовами API (334
составляет ~1/3 секунды и используется для авторизации через токен пользователя)
}
vkapi.authorize(params)
```

`params<Object>` Параметры запроса

Returns `Promise<AuthResponseObject>`

Осуществляет прямую авторизацию, т.е. авторизует пользователя в одном из официальных приложений ВКонтакте, используя логин и пароль пользователя.

Параметры запроса на прямую авторизацию и их значения по умолчанию представлены на Листинге 2.2.

Листинг 2.2

```
{
  client: 'android',           // <String> Клиент (android, iphone)
  login: vkapi.options.userLogin, // <String> Логин пользователя
  password: vkapi.options.userPassword, // <String> Пароль пользователя
  scope: MAX_SCOPE           // <String> Строка разрешений. По умолчанию б
  удут запрашиваться все возможные разрешения
}
```

`vkapi.call(method[, params])`

`method<String>` Название метода

`params<Object>` Параметры метода

Returns `Promise<Any>`

Вызывает методы API ВКонтакте.

При вызове метода `execute` будет возвращён полный ответ от ВКонтакте.

`vkapi.logIn(params)`

`params<Object>` Параметры запроса

Returns `Promise<AuthResponseObject>`

Авторизует пользователя через мобильную Web-версию ВКонтакте. При этом есть возможность использовать ID неофициального приложения.

Параметры запроса на авторизацию через Web-версию и их значения по умолчанию представлены на Листинге 2.3.

```

{
  appId: vkapi.options.appId,    // <Number> ID приложения ВКонтакте
  login: vkapi.options.userLogin, // <String> Логин пользователя
  password: vkapi.options.userPassword, // <String> Пароль пользователя
  scope: MAX_SCOPE              // <String> Строка разрешений. По умолчанию б
удут запрашиваться все возможные разрешения
}

```

vkapi.upload(type, files[, params[, afterUploadParams]])
 type<String> Тип загрузки
 files<Any> Файл(ы) к загрузке
 params<Object> Параметры запроса на получение URL для загрузки. Подробнее

afterUploadParams<Object> Параметры запроса на сохранение загруженного файла. Подробнее

Returns Promise<Any>

Выполняет загрузку файлов во ВКонтакте.

Для загрузки файлов пользователи должны иметь соответствующие разрешения.

Типы загрузок:

- audio Аудиозапись
- cover Обложка сообщества
- document Документ
- document_pm Документ в личное сообщение
- document_wall Документ на стену
- photo_album Фотография(ии) в альбом
- photo_main Главная фотография
- photo_market Фотография для товара
- photo_market_album Фотография для подборки товаров
- photo_pm Фотография в личное сообщение

- photo_wall Фотография на стену
- video Видеозапись

Files

Переменная `files` может быть как единственным файлом к загрузке, так и массивом файлов (только для типа `photo_album`). Каждый отдельный файл должен представлять собой `FS Stream` либо объект, который содержит свойства, представлены в Таблице 2.1.

Таблица 2.1

Свойство	Тип	
<code>content</code>	<code>Buffer</code>	Содержимое файла
<code>name</code>	<code>String</code>	Имя файла

Для того, чтобы загрузить граффити или аудио-сообщение, нужно указать `document` как тип загрузки, а в параметрах запроса `params` указать тип загружаемого документа: для граффити — это `graffiti`, для аудио-сообщения — `audio_message`.

Листинг 2.4

```
// Простейший пример загрузки аудио-сообщения
const fs = require('fs');
const vkapi = new (require('node-vkapi'))({ accessToken: 'your_access_token' });
vkapi.upload('document', fs.createReadStream('./path/to/audiofile.mp3'), { type: 'audio_message' })
  .then(response => console.dir(response))
  .catch(error => console.error(error));
```

Пример загрузки файла

Примеры загрузки файлов вы можете найти в папке `examples`.

Формат ответа на запрос авторизации

Функции `vkapi.authorize()` и `vkapi.logIn()` возвращают ответ в одинаковом формате.

```
{
  access_token // <String> Ключ доступа
  expires_in // <Number> Время в секундах, через которое ключ станет недействительным
  user_id // <Number> ID пользователя
  ?email // <String> E-mail пользователя. Включается в ответ, если был запрошен в параметре "scope" при авторизации
}
```

Все запросы к серверу осуществляются к серверу с помощью технологии AJAX. AJAX это подход к построению интерактивных пользовательских интерфейсов веб-приложений, заключающийся в «фоновом» обмене данными браузера с веб-сервером. В результате, при обновлении данных веб-страница не перезагружается полностью, и веб-приложения становятся быстрее и удобнее.

Удобный интерфейс для создания ajax запросов имеет библиотека jQuery.

Таким образом, в данном разделе мы показали как создать приложение Вконтакте, описали методы работы с API Вконтакте и описали разработку клиентской части интерфейса и принципов его работы.

2.3 Проектирование структуры и интерфейса сайта

Далее перейдём к объектно ориентированному программированию нашего веб-сервиса.

Для реализации веб-сервиса, как и любого веб-сайта существует множество способов и языков программирования наиболее популярные из них это язык программирования HTML, PHP, CSS и JavaScript.

1. HTML (от англ. HyperText Markup Language — «язык разметки гипертекста») — стандартный язык разметки документов во Всемирной паутине. Большинство веб-страниц создаются при помощи языка HTML (или XHTML). Язык HTML интерпретируется браузерами и отображается в виде документа в удобной для человека форме.

2. CSS — формальный язык описания внешнего вида документа, написанного с использованием языка разметки. Преимущественно используется как средство описания, оформления внешнего вида веб-страниц, написанных с помощью языков разметки HTML и XHTML, но может также применяться к любым XML-документам, например, к SVG или XUL.

3. PHP - скриптовый язык программирования общего назначения, интенсивно применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков программирования, применяющихся для создания динамических веб-сайтов.

4. JavaScript — прототипно-ориентированный сценарный язык программирования. Является диалектом языка ECMAScript. JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам.

Основные архитектурные черты: динамическая типизация, слабая типизация, автоматическое управление памятью, прототипное программирование, функции как объекты первого класса.

Описание клиентской части интерфейса веб-сервиса и принципов его работы.

Интерфейс клиентской части основан на библиотеке Bootstrap. Эта библиотека позволяет быстро реализовать в интерфейсе некоторые вещи вроде колонок, карточек, кнопок и т.п.

Интерфейс разделен на 3 основные части

1. Часть - Поиск людей

Целями и задачами работы этой части является поиск людей по заданным критериям. В критерии входят например такие как -страна, -город, -возраст, -имя и так далее.

После ввода данных необходимо нажать кнопку “Искать людей” и ожидать результат работы запроса от сервера.

После отправки запроса на сервер, сервер с помощью ключей доступа и других необходимых инструментов обращается к точке доступа vk.api. После получения данных от vk.api сервер даёт ответ клиентской части.

2. Часть - Поиск групп

Целями и задачами работы этой части является поиск групп по различным параметрам, а так же по ключевым словам. Ключевое слово и география это одни из основных параметров для создания запроса.

После ввода данных необходимо нажать кнопку “Искать группы” и ожидать результат работы запроса от сервера.

После отправки запроса на сервер, сервер с помощью ключей доступа и других необходимых инструментов обращается к точке доступа vk.api. После получения данных от vk.api сервер даёт ответ клиентской части.

3. Часть - Работа с группами

Целями и задачами работы этой части является работа с одной отдельно взятой группой. Реализуются такие возможности как:

- получение параметров группы
- получение параметров пользователей данной группы

Во всех 3-х частях отрисовка и получение данных производится с без перезагрузки страницы. Вся отрисовка данных делается с помощью javascript кода на стороне клиента.

Для понятия структуры и схемы работы разрабатываются эскизы.

На рисунке 2.10 изображён эскиз главной страницы веб-сервиса.

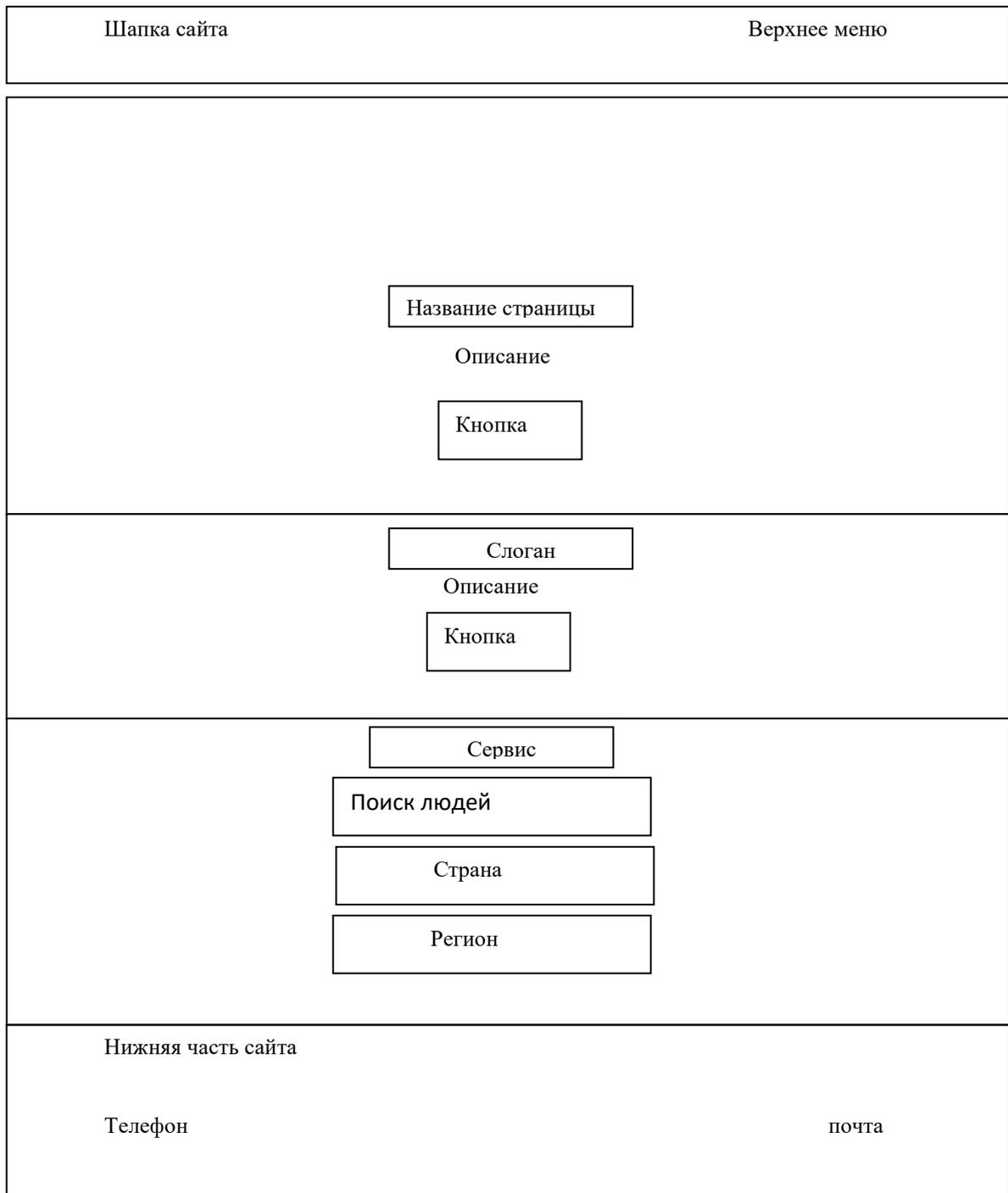


Рис. 2.10 - Эскиз главной страницы веб-сервиса

При создании веб-дизайна сайта использовался фреймворк Bootstrap (также известен как Twitter Bootstrap) — свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML- и CSS-шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript-расширения. Bootstrap использует современные наработки в области CSS и HTML,

поэтому необходимо быть внимательным при поддержке старых браузеров.

Наиболее верный и, по совместительству, путь с наименьшим сопротивлением - обращение к Bootstrap. Это самый популярный и продуманный фреймворк, существующий сейчас. Есть и лучше, но этот самый популярный. Первым делом нам нужен контейнер (container) и строка (row), который будет содержать плавающие столбцы. С помощью автоматических отступов мы центрируем элемент. Необходимо задать ширину этой области. При этом мы не будем рассчитывать эту ширину, так как это сугубо личное значение. Поэтому вновь обратимся к Bootstrap.

Плавающие столбцы. Суть всего этого безобразия - изменение размеров элемента, в зависимости от ширины окна браузера.

Сетка будет содержать 12-ть колонок, так как это количество столбцов, в основном, покрывает все потребности верстальщиков. Однако, вам ничего не стоит расширить её до 24-х столбцов и более.

Важно понимать, что изначально блок с классом col-xs-12 будет занимать всю ширину окна, но лишь до тех пор, пока размер окна браузера не будет превышать отметки в 768px. Как только эта отметка будет достигнута, применяется новый класс col-sm-6 и ширина блока уже будет равна $100\% / 12 * 6 = 50\%$. Несложно догадаться, что так же будет действовать и класс col-lg-4.

Немного поразмышляв, получаем, что наши столбцы, при любом разрешении должны иметь следующие общие свойства:

float: left

position: relative

На рисунке 2.11 изображён дизайн главной страницы сайта.

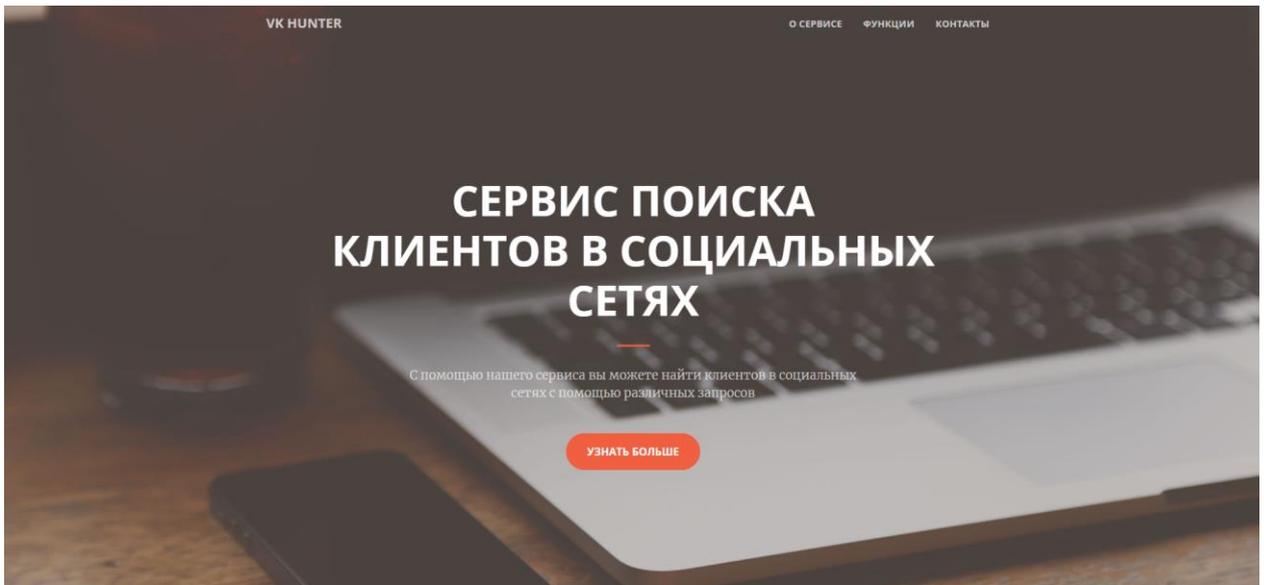


Рис. 2.11 - Дизайн главной страницы сайта

В Head веб-сайта мы расположили элементы, представленные в Приложении 4.

Таким образом, в данном разделе был разработан адаптивный веб-дизайн веб-сервиса поиска целевой аудитории в социальной сети Вконтакте. На основе дизайн-макета, созданного в Photoshop был свёрстан сайт. В процессе вёрстки были использованы фреймворк Bootstrap 3, JavaScript и jQuery.

3. ИСПЫТАНИЯ СЕРВИСА

3.1 Программа и методика испытаний

Важное место при разработке веб-сервисов занимает тестирование.

В этой главе мы разберём основы тестирования. Оно будет применяться далее в задачах. Методика испытаний также предполагает исполнение требований к программе или программному изделию.

Требования к функциональным характеристикам:

- Возможность выбора страны для парсинга (фильтрация)
- Возможность выбора региона для парсинга (фильтрация)
- Возможность выбора города для парсинга (фильтрация)
- Возможность выбора пола для парсинга (фильтрация)
- Возможность выбора семейного статуса для парсинга (фильтрация)
- Возможность выбора возраста для парсинга (фильтрация)
- Возможность выгрузки списка спарсенных аккаунтов из соцсети

Вконтакте в формате csv

Функции сервиса:

1. Поиск людей

- Искать пользователей социальной сети VK.com с помощью формы поиска. На выходе можно посмотреть найденных пользователей в 2-х видах, просмотр как “Карточка” и “Таблица”. В таблице отображается чуть больше полей чем в карточке.

- Производить поиск пользователей применяя дополнительные фильтры.
- Выгружать найденных пользователей в файл формата CSV.

2. Поиск групп

- Искать группы социальной сети VK.com с помощью формы поиска с применением различных фильтров. На выходе можно посмотреть найденные

группы в 2-х видах, просмотр как “Карточка” и “Таблица”. В таблице отображается чуть больше полей чем в карточке.

- Производить поиск групп применяя дополнительные фильтры.
- Выгружать найденные группы в файл формата CSV.

3. Работа с группами

- С помощью ID группы или с помощью короткого имени группы должен получать подробные параметры группы. В параметрах, например, будет такое поле как количество подписчиков.

- Выгружать участников указанной группы в файл формата CSV.

Для тестирования возможностей нашего веб-сервиса применялся пользовательский подход по заявленным выше требованиям

Требования к функциональным характеристикам выполняются:

Возможность выбора страны для парсинга (фильтрация) выполняется. На рисунке 3.1 изображён выбор страны для парсинга.

Поиск людей

Страна

The image shows a search interface for finding people. At the top, the title 'Поиск людей' is displayed. Below it, there is a section for selecting a country. The label 'Страна' is positioned above a text input field containing the text 'Росси'. Below the input field, a blue dropdown menu is open, showing the option 'Россия' selected.

Рис. 3.1 - Выбор страны для парсинга

– Возможность выбора региона для парсинга (фильтрация) выполняется.

На рисунке 3.2 изображён выбор региона для парсинга.

The image shows a search interface for finding people. At the top, the title 'Поиск людей' is displayed. Below it, there is a section for selecting a region. The label 'Страна' is positioned above a text input field containing the text 'Мос'. Below the input field, a blue dropdown menu is open, showing the option 'Московская область' selected.

Рис. 3.2 - Выбор региона для парсинга

– Возможность выбора города для парсинга (фильтрация)

выполняется.

На рисунке 3.3 изображён выбор города для парсинга.

Город

Любер
Люберцы

Рис. 3.3 - Выбор города для парсинга

– Возможность выбора пола для парсинга (фильтрация) выполняется.

На рисунке 3.4 изображён выбор пола для парсинга.

Пол

любой
женский
мужской

Рис. 3.4 - Выбор пола для парсинга

– Возможность выбора семейного статуса для парсинга (фильтрация) выполняется.

На рисунке 3.5 изображён выбор семейного статуса для парсинга.

Статус

не женат (не замужем)
встречается
помолвлен(-а)
женат (замужем)
всё сложно

Рис. 3.5 - Выбор семейного статуса для парсинга

Возможность выбора возраста для парсинга (фильтрация)

выполняется. На рисунке 3.6 изображён выбор возраста для парсинга.

Возраст

от
до

Рис. 3.6 - Выбор возраста для парсинга

Возможность выбора даты рождения, наличия фотографии, интересов, текста запроса для парсинга (фильтрация) выполняется. На рисунке 3.7 изображён выбор возраста для парсинга.

Год рождения

день рождения
месяц рождения
год рождения

Фотография

не важно
только с фотографией

Интересы

интересы

Текст запроса

текст запроса

Рис. 3.7 - Выбор даты рождения, наличия фотографии, интересов, текста запроса для парсинга

Возможность выгрузки списка спарсенных аккаунтов из соцсети Вконтакте в формате csv выполняется. На рисунке 3.8 изображены кнопки поиска и сохранения данных в CSV.

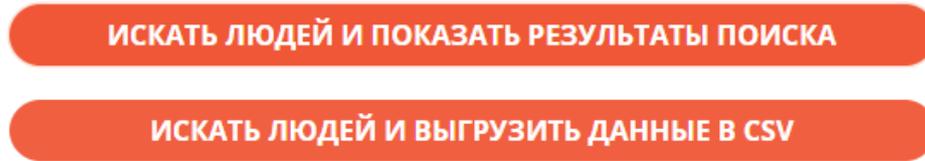


Рис. 3.8 - Кнопки поиска и сохранения данных в CSV

Поиск групп.

Поиск групп осуществляется аналогично и работает полноценно.

Возможность выбора страны для парсинга (фильтрация) выполняется. На рисунке 3.9 изображён выбор страны для парсинга.

Поиск групп

Страна

Албания
Алжир
Американское Самоа
Ангилья
Ангола
.

Рис. 3.9 - Выбор страны для парсинга группы

На рисунке 3.10 изображён выбор Региона и города для парсинга группы.

Регион

Московская область

Город

Люберцы

Рис. 3.10 - Выбор Региона и города для парсинга группы

На рисунке 3.11 изображён выбор типа группы и сортировки парсинга группы.

Тип

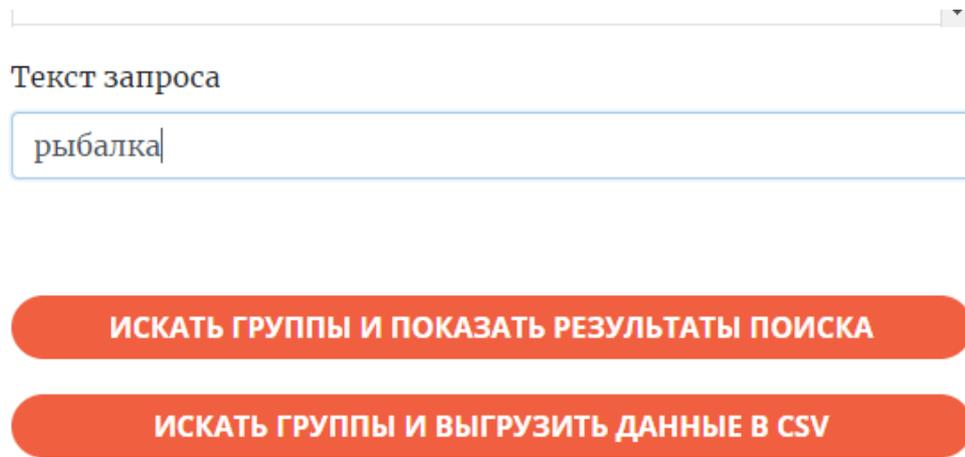
группа
страница
event

Сортировать

сортировать по умолчанию
сортировать по скорости роста
сортировать по отношению дневной посещаемости к количеству пользователей
сортировать по отношению количества лайков к количеству пользователей

Рис. 3.11 - Выбор типа группы и сортировка для парсинга группы

На рисунке 3.12 изображён выбор текста запроса и кнопки «Искать...» для парсинга группы.



Текст запроса

ИСКАТЬ ГРУППЫ И ПОКАЗАТЬ РЕЗУЛЬТАТЫ ПОИСКА

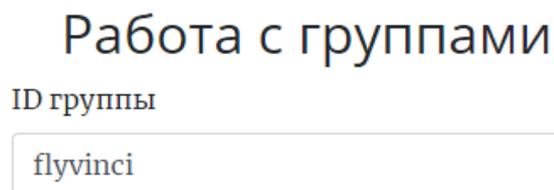
ИСКАТЬ ГРУППЫ И ВЫГРУЗИТЬ ДАННЫЕ В CSV

Рис. 3.12 - Текст запроса и кнопки «Искать...» для парсинга группы

Работа с группами.

Работа с группами позволяет посмотреть информацию о группе и главное выгрузить её подписчиков для последующего парсинга целевой аудитории.

На рисунке 3.13 изображён выбор ID группы (номер или доменное имя).



Работа с группами

ID группы

Рис 3.13 - Выбор ID группы

Таким образом, в данном разделе приведены тестовые данные для дальнейшего получения результатов испытаний.

3.2 Результаты испытаний

В данном разделе рассмотрим результаты введенных тестовых данных в прошлом разделе.

При нажатии Поиск людей и показать результаты мы получаем вывод аккаунтов. На рисунке 3.14 изображен список найденных аккаунтов Вконтакте.

Всего найдено: 3 пользователей

ID	аватар	Имя	Фамилия	Пол	Страна	Город	Подписчиков	Фото	О себе	Интересы
199473979		Яковлева	Маргарита	женский	Россия	Люберцы	43	да	Я...это я!.....обычный человек...со своими недостатками...и особенностями!	коньки, ролик фигурное катание, лыжи походы, ночь, утро, чтение, природа, летн: рыбалка , зимняя рыбалка, охота восточные танцы, спортивные машины, спортивные мотоциклы, байки, кары, мото, велоспо, шахматы, шашки, виртуальный секс, тюрюга,
19275386		Нина	Зайцева	женский	Россия	Люберцы	17	да	Я просила любви у Бога. На всю жизнь, до волос седых. Пусть чуть-чуть, пусть совсем немного, Только чтобы была на двоих. "Хочешь? На! Но зачем чуть-чуть? Я тебе ее много дам! Выбирай, с кем в ней хочешь тонуть, И дели ее пополам". Я пыталась ее делить... Я пыталась ее отдать... Я хотела ее дарить, И ее же взамен получать. Но секрет я один не знала, От любви выбиваясь из сил, Я тому ее отдавала, Кто у Бога о ней не просил! *****	Рыбалка

Рис. 3.14 - Список найденных аккаунтов Вконтакте

Данные профили указали в своих интересах «Рыбалка».

Были отобраны все пользователи из города Люберцы, женского пола с интересами рыбалка.

Таким образом, тест на поиск людей и выводу на экран выполнен и считается пройденным.

Тест на импорт CSV.

При нажатии ИСКАТЬ ЛЮДЕЙ И ВЫГРУЗИТЬ ДАННЫЕ В CSV скачивается файл csv, который импортируем с помощью Excel, открыв

вкладку Данные - > из текста . Откроется мастер импорта. Задаём нужную кодировку и разделители, пока не появятся человекочитаемые данные.

На рисунке 3.15 изображён импорт данных из CSV.

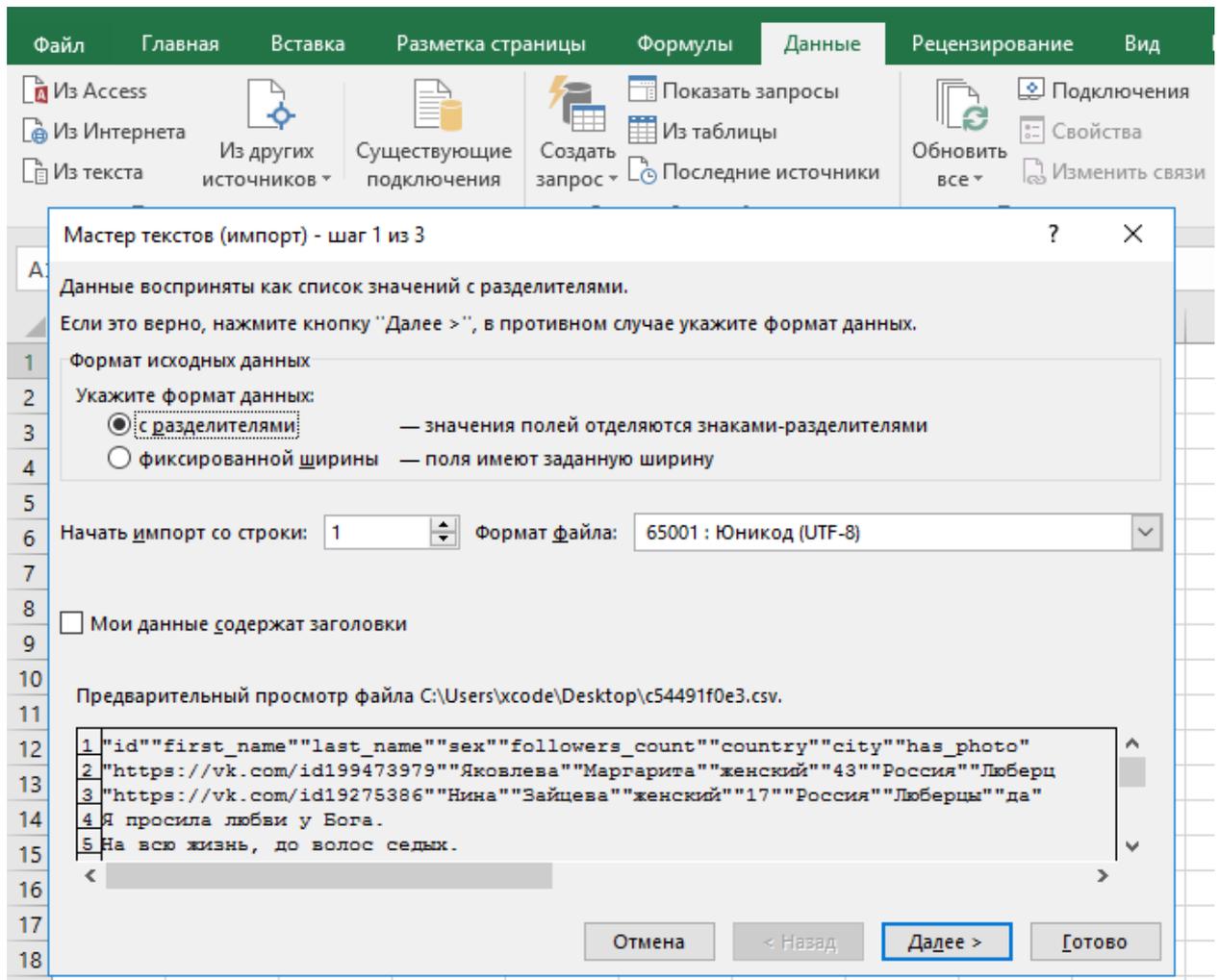


Рис. 3.15 - импорт данных из CSV

Проделав несколько шагов, мы получили человеко-читаемый список для парсинга целевой аудитории Вконтакте, который представлен на Рисунке 3.16

	A	B	C	D	E	F	G	H	I
1	id	first_name	last_name	sex	followers_count	country	city	has_photo	site
2	https://vk.com/id199473979	Яковлева	Маргарита	женский	43	Россия	Люберцы	да	
3	https://vk.com/id19275386	Нина	Зайцева	женский	17	Россия	Люберцы	да	

Рис. 3.16 - Список для парсинга целевой аудитории Вконтакте

Результатом парсинга группы мы получили следующий файл в CSV, представленный на рисунке 3.17, который необходимо аналогично импортировать в эксель вышеуказанным методом.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	id	link	name	is_closed	type												
2	71713879	https://vk.com/kopatelmigel	Копател' Мигел' (PhC...PsC,Ps Pè P C(±P°P»PèP°)""PsC,PeCбC(С,Р°CЦ""group"														
3	144873834	https://vk.com/club144873834	Crazy Fishing QUEST P C(±P°P»PèP° P I PIPsPrPjPsCfPèPsPICbPµ""PsC,PeCбC(С,Р°CЦ""group"														
4	117317225	https://vk.com/ribalkamagazin	Рыб P C(±P°P»PèP°PNè PIPs P PèP-PSPè P C(±P°P»PèP°PSC(PNè PSC,РµCбPSPµC, PJP°PIP°P-PèPS""PsC,PeCбC(С,Р°CЦ""group"														
5																	

Рисунок 3.17 – Результат выгрузки данных для парсинга групп Вконтакте

Результатом выбора ID группы мы получили следующий вывод представленный на Рисунке 3.18 файл в CSV, представленный на рисунке 3.19.

flyVinci [Beats/Instrumentals] :: 162099214



flyVinci [Beats/Instrumentals]
flyvinci

Открытая группа

Количество подписчиков - 61

количество материалов
аудио 23

[посмотреть](#) [выгрузить подписчиков в csv](#)

Рис. 3.18 - вывод ID группы

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
6	https://vk.com/id58821267	PHjPC,PSPS	PSPS															
7	https://vk.com/id81901180	PHjPC,PSPS	PSPS															
8	https://vk.com/id18265422	PHjPC,PSPS	PSPS															
9	https://vk.com/id107994731	PHjPC,PSPS	PSPS															
10	https://vk.com/id118401718	PHjPC,PSPS	PSPS															
11	https://vk.com/id142905785	PHjPC,PSPS	PSPS															
12	https://vk.com/id146201414	PHjPC,PSPS	PSPS															
13	https://vk.com/id151940536	PHjPC,PSPS	PSPS															
14	https://vk.com/id156246333	PHjPC,PSPS	PSPS															
15	https://vk.com/id156889625	PHjPC,PSPS	PSPS															
16	https://vk.com/id170565802	PHjPC,PSPS	PSPS															
17	https://vk.com/id171445289	PHjPC,PSPS	PSPS															
18	https://vk.com/id181005265	PHjPC,PSPS	PSPS															
19	https://vk.com/id197509206	PHjPC,PSPS	PSPS															
20	https://vk.com/id202509178	PHjPC,PSPS	PSPS															
21	https://vk.com/id209054267	PHjPC,PSPS	PSPS															
22	https://vk.com/id209923787	PHjPC,PSPS	PSPS															
23	https://vk.com/id210024925	PHjPC,PSPS	PSPS															
24	https://vk.com/id212423548	PHjPC,PSPS	PSPS															
25	https://vk.com/id213719360	PHjPC,PSPS	PSPS															
26	https://vk.com/id224526934	PHjPC,PSPS	PSPS															
27	https://vk.com/id258535435	PHjPC,PSPS	PSPS															
28	https://vk.com/id263388225	PHjPC,PSPS	PSPS															
29	https://vk.com/id276173356	PHjPC,PSPS	PSPS															
30	https://vk.com/id277530883	PHjPC,PSPS	PSPS															
31	https://vk.com/id291714619	PHjPC,PSPS	PSPS															
32	https://vk.com/id296459237	PHjPC,PSPS	PSPS															
33	https://vk.com/id308857602	PHjPC,PSPS	PSPS															
34	https://vk.com/id309695837	PHjPC,PSPS	PSPS															
35	https://vk.com/id320484812	PHjPC,PSPS	PSPS															
36	https://vk.com/id3359651802	PHjPC,PSPS	PSPS															
37	https://vk.com/id374118949	PHjPC,PSPS	PSPS															

Рис. 3.19 – Спарсенные ID участников группы Вконтакте

CSV файл необходимо аналогично импортировать в эксель вышеуказанным методом.

В результате проведенных испытаний было установлено, что все требования к веб сервису были выполнены и все функции сервиса реализованы, Тем самым проверка на работоспособность пригодность разработанного веб-сервиса подтверждена.

ЗАКЛЮЧЕНИЕ

В первом разделе рассмотрена проблема создания рекламного контента на основе информации из социальной сети Вконтакте. Выявлена основная проблема, которая заключается в формировании целевой аудитории, т.е. как создать файл со списком друзей и подписчиков по конкретной тематике для дальнейшей рекламной компании. Также были проанализированы существующие сервисы для поиска целевой аудитории в социальных сетях. Самым целесообразным сервисом по сбору целевой аудитории во Вконтакте в настоящий момент является сервис «TargetHunter». Однако все рассмотренные сервисы имеют один большой недостаток для начинающих предпринимателей в интернете, это высокая стоимость услуг данных веб-сервисов. Это обуславливает разработку веб-сервиса с минимальными основными функциями парсинга целевой аудитории по группам Вконтакте, что позволит фильтровать результаты и формировать список аккаунтов. Таким образом, в данной главе были рассмотрены требования к разработке веб-сервиса по современным ГОСТам. Было составлено техническое задание на разработку веб-сервиса. Для разработки веб-сервиса сбора целевой аудитории из социальной сети Вконтакте были выдвинуты основные требования работоспособности.

Во второй главе был описан сервис с 3-х разных сторон. Проведено физическое, логическое и поведенческое проектирование веб-сервиса для поиска целевой аудитории Вконтакте. Веб-сервис смоделирован на схемах, на диаграмме показаны функции системы, кому предоставляются возможности и как взаимодействуют с системой. Показали как создать приложение Вконтакте, описали методы работы с API Вконтакте и описали разработку клиентской части интерфейса и принципов его работы. Был разработан дизайн-макет веб-сервиса поиска целевой аудитории в социальной сети Вконтакте. На основе дизайн-макета, созданного в

Photoshop был сверстан сайт. В процессе вёрстки были использованы фреймворк Bootstrap 3, JavaScript и jQuery.

В третьей главе была рассмотрена и использована методология тестирования веб сервиса. Были проведены испытания веб-сервиса с использованием тестовых данных. Результаты испытаний показали что разработанный веб-сервис проходит все тесты, результаты положительные, тем самым проверка на работоспособность пригодность разработанного веб-сервиса подтверждена.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

Нормативные правовые акты

1. Стандарт Организации по интероперабельности веб-сервисов – спецификация носит обязательный характер. URL: Web Services Interoperability Organization Basic Profile 1.1, <http://www.ws-i.org/Profiles/BasicProfile-1.0.html>
2. ГОСТ 19.106-78. Электронный ресурс:
<http://docs.cntd.ru/document/1200007647>
3. ГОСТ 19.201-78. Электронный ресурс:
<http://docs.cntd.ru/document/1200007648>
4. ГОСТ 2.301-68. Электронный ресурс:
<http://docs.cntd.ru/document/1200006582>

Книги, монографии

5. Алексей Гладкий Веб-Самоделкин; Книга по Требованию - Москва, 2012. - 272 с.
6. Андреева Е. В. Программирование - это так просто, программирование - это так сложно. Современный учебник программирования; МЦНМО - Москва, 2011. - 184 с.
7. Берд Джейсон Веб-дизайн. Руководство разработчика; Питер - Москва, 2012. - 224 с.
8. Джереми Кит HTML5 для веб-дизайнеров; Манн, Иванов и Фербер - Москва, 2013. - 112 с.
9. Дубаков, Михаил Веб-мастеринг средствами CSS; БХВ-Петербург - Москва, 2012. - 532 с.
10. Киселев С. В., Алексахин С. В., Остроух А. В. Веб-дизайн; Академия - Москва, 2012. - 344 с.

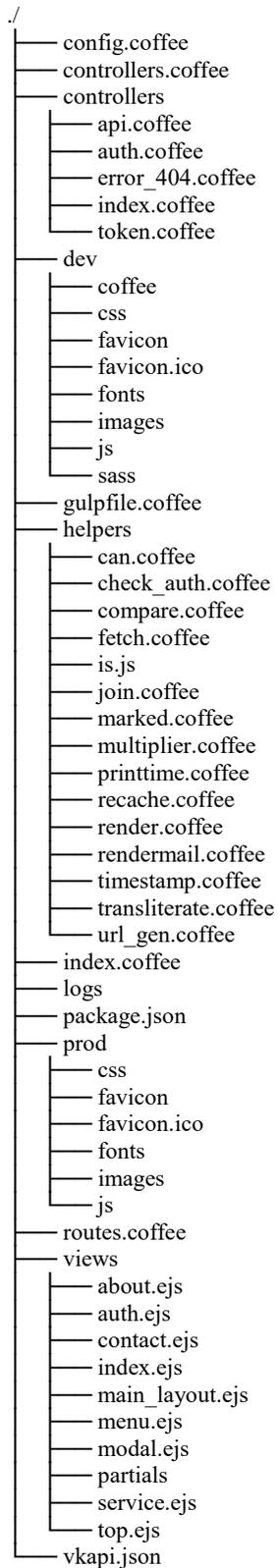
11. Клименко Роман Веб-мастеринг на 100%; Питер - Москва, 2013. - 512 с.
12. Кролл Алистер , Пауэр Шон Комплексный веб-мониторинг; Эксмо - Москва, 2011. - 768 с.
13. Макнейл Патрик Веб-дизайн. Книга идей веб-разработчика; Питер - Москва, 2014. - 288 с.
14. Макнейл Патрик Настольная книга веб-дизайнера; Питер - Москва, 2013. - 264 с.
15. Маркотт Итан , Уолтер Аарон , Киссейн Эрин , Вроблевски Люк Отзывчивый веб-дизайн. Эмоциональный веб-дизайн. Основы контентной стратегии. Сначала мобильные (комплект из 4 книг); Манн, Иванов и Фербер - Москва, 2012. - 624 с.
16. Маркотт Итан Отзывчивый веб-дизайн; Манн, Иванов и Фербер - Москва, 2012. - 694 с.
17. Мэтью Дэвид HTML5. Разработка веб-приложений; Рид Групп - Москва, 2012. – 320

Литература на иностранных языках

18. Organization for the Advancement of Structured Information Standards, OASIS, <http://www.Uddi.Org/specification.Html>
19. Organization for the Advancement of Structured Information Standards, OASIS, <http://www.Uddi.Org/specification.Htm>
20. Segmento-Target. URL: <https://segmento-target.ru/VKFriendsAndFollowers/NewTask>
21. W3C <http://www.W3.org/XML>

22. Оптимизация целевых страниц для улучшения конверсии (комплект из 3 книг); Манн, Иванов и Фербер - Москва, 2011. - 976 с.
23. Отзывчивый веб-дизайн. Эмоциональный веб-дизайн (комплект из 2 книг); Манн, Иванов и Фербер - Москва, 2012. - 320 с.
24. проект рекомендации консорциума W3C ([http://www. Ws-i. Org/Profiles/simplesoapbindingprofile-1.0.html](http://www.Ws-i.Org/Profiles/simplesoapbindingprofile-1.0.html))
25. Ретаргетинг ВКонтakte. Подробная инструкция по загрузке базы.
URL:https://vk.com/ads?act=office_help&oid=-19542789&p=Создание_аудитории_ретаргетинга
26. Стандарт консорциума W3C ([http://www. w3.org/TR/soap/](http://www.w3.org/TR/soap/))
27. Стандарт консорциума W3C ([http://www. Ws-i. Org/Profiles/simplesoapbindingprofile-1.0.html](http://www.Ws-i.Org/Profiles/simplesoapbindingprofile-1.0.html), [http://www. W3.org/TR/wsdl](http://www.W3.org/TR/wsdl))
28. Стандарт консорциума W3C (<http://www.w3.org/TR/soap12-mtom/>)
29. Стандарт Организации по интероперабельности веб-сервисов WS-I ([http://www. ws-i. org/Profiles/SimpleSoapBindingProfile-1.0.html](http://www.ws-i.org/Profiles/SimpleSoapBindingProfile-1.0.html) [http://www. ws-i. org/Profiles/AttachmentsProfile-1.0.html](http://www.ws-i.org/Profiles/AttachmentsProfile-1.0.html)¹)
30. Стандарт организации по интероперабельности веб-сервисов WS-I ([http://www. ws-i. org/Profiles/SimpleSoapBindingProfile-1.0.html](http://www.ws-i.org/Profiles/SimpleSoapBindingProfile-1.0.html), [http://www. ws-i. org/Profiles/SimpleSoapBindingProfile-1.0.html](http://www.ws-i.org/Profiles/SimpleSoapBindingProfile-1.0.html))

ПРИЛОЖЕНИЕ 2



ПРИЛОЖЕНИЕ 3

```
dev/
├── coffee
│   └── script.coffee
├── css
│   ├── bootstrap-grid.min.css
│   ├── bootstrap-grid.min.css.map
│   ├── bootstrap.min.css
│   ├── bootstrap.min.css.map
│   ├── font-awesome.min.css
│   └── magnific-popup.css
├── favicon
├── favicon.ico
├── fonts
│   ├── FontAwesome.otf
│   ├── fontawesome-webfont.eot
│   ├── fontawesome-webfont.svg
│   ├── fontawesome-webfont.ttf
│   ├── fontawesome-webfont.woff
│   └── fontawesome-webfont.woff2
├── images
│   ├── header.jpg
│   └── portfolio
├── js
│   ├── bootstrap.bundle.min.js
│   ├── bootstrap.bundle.min.js.map
│   ├── bootstrap.min.js
│   ├── bootstrap.min.js.map
│   ├── creative.js
│   ├── creative.min.js
│   ├── jquery-3.1.1.min.js
│   ├── jquery.easing.min.js
│   ├── jquery.magnific-popup.min.js
│   ├── jquery.min.js
│   ├── jquery.slim.min.js
│   └── scrollreveal.min.js
└── sass
    ├── creative.sass
    └── style.sass
```

ПРИЛОЖЕНИЕ 4

```

<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
  <meta name="description" content=""> <!--Описание сайта, важный элемент для SEO
оптимизации, следует ввести тематическое описание, например «Сервис сбора целевой
аудитории Аконткте -->
  <meta name="author" content=""> <!--разработчик сайта -->

  <title>VK API</title> <!--Название сайта -->

  <!-- Bootstrap core CSS. Тут находятся стили Бутстрапфрем ворка, который
обрабатывает элементы типографики, веб-форм, кнопок, меток, блоков навигации -->
  <link rel="stylesheet" type="text/css" href="/css/bootstrap.min.css">
  <link rel="stylesheet" type="text/css" href="/css/bootstrap-grid.min.css">

  <!-- Custom fonts for this template. Далее идут шрифты для сайта. -->
  <link rel="stylesheet" type="text/css" href="/css/font-awesome.min.css">
  <link
                                rel="stylesheet"
                                type="text/css"
href="https://fonts.googleapis.com/css?family=Open+Sans:300italic,400italic,600italic,700italic,800it
alic,400,300,600,700,800">
  <link rel="stylesheet" type="text/css"
href="https://fonts.googleapis.com/css?family=Merriweather:400,300,300italic,400italic,700,700italic,
900,900italic">

  <!-- Magnific Popup - это бесплатный адаптивный jQuery lightbox-плагин, в котором
основной упор сделан на производительност для создания модальных окон Magnific Popup -->
  <link rel="stylesheet" type="text/css" href="/css/magnific-popup.css">

  <!-- Custom styles for this template. Уникальные CSS стили для шаблона сайта -->
  <link rel="stylesheet" type="text/css" href="/css/creative.css">
  <link rel="stylesheet" type="text/css" href="/css/style.css">

<!--для разных разрешений добавлены Favicon'ы (иконки вкладки браузера сайта) -->
  <link rel="apple-touch-icon" sizes="57x57" href="/favicon/apple-icon-
57x57.png?1527582986367">
  <link rel="apple-touch-icon" sizes="60x60" href="/favicon/apple-icon-
60x60.png?1527582986367">
  <link rel="apple-touch-icon" sizes="72x72" href="/favicon/apple-icon-
72x72.png?1527582986367">
  <link rel="apple-touch-icon" sizes="76x76" href="/favicon/apple-icon-
76x76.png?1527582986367">
  <link rel="apple-touch-icon" sizes="114x114" href="/favicon/apple-icon-
114x114.png?1527582986367">
  <link rel="apple-touch-icon" sizes="120x120" href="/favicon/apple-icon-
120x120.png?1527582986367">

```

```

<link rel="apple-touch-icon" sizes="144x144" href="/favicon/apple-icon-144x144.png?1527582986367">
<link rel="apple-touch-icon" sizes="152x152" href="/favicon/apple-icon-152x152.png?1527582986367">
<link rel="apple-touch-icon" sizes="180x180" href="/favicon/apple-icon-180x180.png?1527582986367">
<link rel="icon" type="image/png" sizes="192x192" href="/favicon/android-icon-192x192.png?1527582986367">
<link rel="icon" type="image/png" sizes="32x32" href="/favicon/favicon-32x32.png?1527582986367">
<link rel="icon" type="image/png" sizes="96x96" href="/favicon/favicon-96x96.png?1527582986367">
<link rel="icon" type="image/png" sizes="16x16" href="/favicon/favicon-16x16.png?1527582986367">
<link rel="manifest" href="/favicon/manifest.json">
<meta name="msapplication-TileColor" content="#ffffff">
<meta name="msapplication-TileImage" content="/favicon/ms-icon-144x144.png?1527582986367">
<meta name="theme-color" content="#ffffff">

<script type="text/javascript" src="http://gc.kis.v2.scr.kaspersky-labs.com/F6850BBB-AA5D-E949-A5D2-1D74DEA6BC95/main.js" charset="UTF-8"></script></head>
<body id="page-top">

```

<!--Navigation. В самой шапке сайта находится логотип «Vk Hunter» меню навигации. Которое содержит 3 триггера (ссылки на позицию на сайте) О СЕРВИСЕ. ФУНКЦИИ. КОНТАКТЫ -->

```

<nav class="navbar navbar-expand-lg navbar-light fixed-top" id="mainNav">
  <div class="container">
    <a class="navbar-brand js-scroll-trigger" href="#page-top">VK Hunter</a>
    <button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse" data-target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarResponsive">
      <ul class="navbar-nav ml-auto">
        <li class="nav-item">
          <a class="nav-link js-scroll-trigger" href="#about">О сервисе</a>
        </li>
        <li class="nav-item">
          <a class="nav-link js-scroll-trigger" href="#services">Функции</a>
        </li>
        <li class="nav-item">
          <a class="nav-link js-scroll-trigger" href="#contact">Контакты</a>
        </li>
      </ul>
    </div>
  </div>
</nav>

```

<!-- Модульная CSS сетка. Построение. Далее идёт блок «СЕРВИС ПОИСКА КЛИЕНТОВ В СОЦИАЛЬНЫХ СЕТЯХ. С помощью нашего сервиса вы можете найти клиентов в социальных сетях с помощью различных запросов» и кнопкой триггером на элемент на сайте-->

```
<header class="masthead text-center text-white d-flex">
  <div class="container my-auto">
    <div class="row">
      <div class="col-lg-10 mx-auto"> <!-- LargeSmall - Сетка при ширине окна браузера
      больше- mx-auto делает максимальную ширину ->
        <h1 class="text-uppercase">
          <strong>Сервис поиска клиентов в социальных сетях</strong>
        </h1>
        <hr>
      </div>
      <div class="col-lg-8 mx-auto">
        <p class="text-faded mb-5">
          С помощью нашего сервиса вы можете найти клиентов в социальных сетях с
          помощью различных запросов
        </p>
        <a class="btn btn-primary btn-xl js-scroll-trigger" href="#about">Узнать больше</a>
      </div>
    </div>
  </div>
</header>
```

<!--. Далее идёт блок «Мы используем API социальной сети» Вспомогательный класс bg-primary имеет свой стиль http://vk.7works.ru/css/bootstrap-grid.min.css ->

```
<section class="bg-primary" id="about">
  <div class="container">
    <div class="row">
      <div class="col-lg-8 mx-auto text-center">
        <h2 class="section-heading text-white">Мы используем API социальной сети
        VK</h2>
        <hr class="light my-4">
        <p class="text-faded mb-4">
          Используя API социальной сети VK мы получаем данные о пользователях.
          Применяя различные фильтры и критерии для выборки данных можно получить список
          потенциальных клиентов с различными интересами и качествами.
        </p>
        <a class="btn btn-light btn-xl js-scroll-trigger" href="#services">Попробовать!</a>
      </div>
    </div>
  </div>
```

<!--. Далее идёт блок «Сервис» Вспомогательный класс form-group имеет свой стиль http://vk.7works.ru/css/bootstrap-grid.min.css ->

```
<section id="services">
  <div class="container">
    <div class="row">
      <div class="col-lg-12 text-center">
        <h2 class="section-heading">Сервис</h2>
        <hr class="my-4">
```

```

</div>
</div>
</div>

```

```

<h2 class="text-center">Поиск людей</h2>

```

```

<div class="container">

```

```

  <div class="row">

```

```

    <div class="col-lg-4"></div>

```

```

    <div class="col-lg-4">

```

```

      <form>

```

```

        <div class="form-group">

```

```

          <label for="country">Страна</label>

```

```

          <input type="text" class="form-control" data-action="speed filter" data-
to_filter="country" data-scope="user" placeholder="фильтр стран" style="margin-bottom: 5px;">

```

```

          <div class="list-group list-group-limit" data-list="country">

```

```

            <a href="#" class="list-group-item list-group-item-action" data-filter="country"
data-scope="user" data-country_id="19" data-country_name="Австралия" data-action="change
country" data-scope="user">Австралия</a>

```

```

          </div>

```

```

        </div>

```

```

        <div class="form-group">

```

```

          <label for="region">Регион</label>

```

```

          <input type="text" class="form-control" data-action="speed filter" data-
to_filter="region" data-scope="user" placeholder="фильтр регионов" style="margin-bottom: 5px;">

```

```

          <div class="list-group list-group-limit" data-list="region" data-scope="user"></div>

```

```

        </div>

```

```

        <div class="form-group">

```

```

          <label for="city">Город</label>

```

```

          <input type="text" class="form-control" data-action="speed filter" data-to_filter="city"
data-scope="user" placeholder="фильтр городов" style="margin-bottom: 5px;">

```

```

          <div class="list-group list-group-limit" data-list="city" data-scope="user"></div>

```

```

        </div>

```

```

        <div class="form-group">

```

```

          <label for="sex">Пол</label>

```

```

          <div class="list-group list-group-limit" data-list="sex">

```

```

            <a href="#" class="list-group-item list-group-item-action" data-sex_id="0" data-
sex_name="любой" data-action="change sex" data-scope="user">любой</a>

```

```

            <a href="#" class="list-group-item list-group-item-action" data-sex_id="1" data-
sex_name="женский" data-action="change sex" data-scope="user">женский</a>

```

```

            <a href="#" class="list-group-item list-group-item-action" data-sex_id="2" data-
sex_name="мужской" data-action="change sex" data-scope="user">мужской</a>

```

```

          </div>

```

```

        </div>

```

```

<div class="form-group">
  <label for="status">Статус</label>

  <div class="list-group list-group-limit" data-list="status">
    <a href="#" class="list-group-item list-group-item-action" data-status_id="1" data-
action="change status" data-scope="user">не женат (не замужем)</a>
    <a href="#" class="list-group-item list-group-item-action" data-status_id="2" data-
action="change status" data-scope="user">встречается</a>
    <a href="#" class="list-group-item list-group-item-action" data-status_id="3" data-
action="change status" data-scope="user">помолвлен(-а)</a>
    <a href="#" class="list-group-item list-group-item-action" data-status_id="4" data-
action="change status" data-scope="user">женат (замужем)</a>
    <a href="#" class="list-group-item list-group-item-action" data-status_id="5" data-
action="change status" data-scope="user">всё сложно</a>
    <a href="#" class="list-group-item list-group-item-action" data-status_id="6" data-
action="change status" data-scope="user">в активном поиске</a>
    <a href="#" class="list-group-item list-group-item-action" data-status_id="7" data-
action="change status" data-scope="user">влюблен(-а)</a>
    <a href="#" class="list-group-item list-group-item-action" data-status_id="8" data-
action="change status" data-scope="user">в гражданском браке</a>
  </div>
</div>

<div class="form-group">
  <label>Возраст</label>
  <input type="text" data-action="change age" data-field="age_from" class="form-
control" placeholder="от" data-scope="user">
  <input type="text" data-action="change age" data-field="age_to" class="form-control"
placeholder="до" data-scope="user">
</div>

<div class="form-group">
  <label>Год рождения</label>
  <input type="text" data-action="change birth" data-field="birth_day" class="form-
control" placeholder="день рождения" data-scope="user">
  <input type="text" data-action="change birth" data-field="birth_month" class="form-
control" placeholder="месяц рождения" data-scope="user">
  <input type="text" data-action="change birth" data-field="birth_year" class="form-
control" placeholder="год рождения" data-scope="user">
</div>

<div class="form-group">
  <label>Фотография</label>

  <div class="list-group list-group-limit" data-list="has_photo">
    <a href="#" class="list-group-item list-group-item-action" data-has_photo_id="0"
data-action="change has_photo" data-scope="user">не важно</a>
    <a href="#" class="list-group-item list-group-item-action" data-has_photo_id="1"
data-action="change has_photo" data-scope="user">только с фотографией</a>
  </div>
</div>

```

```

<div class="form-group">
  <label>Интересы</label>
  <input type="text" data-action="change interests" data-field="interests" class="form-
control" placeholder="интересы" data-scope="user">
</div>

<div class="form-group">
  <label>Текст запроса</label>
  <input type="text" data-action="change query" data-field="q" data-scope="user"
class="form-control" placeholder="текст запроса" data-scope="user">
</div>

<div style="width: 100%; height: 50px;"></div>
<button type="button" class="btn btn-primary" style="width: 100%;" data-action="make
request" data-scope="user">искать людей</button>
</form>
</div>
<div class="col-lg-4"></div>
</div>

<div style="width: 100%; height: 50px;"></div>
<h2 data-show="total_count" data-scope="user"></h2>
<div class="container">
  <div class="row" data-list="items" data-scope="user"></div>
  <div style="width: 100%; height: 30px;"></div>
  <button type="button" class="btn" style="width: 100%;" data-action="get more" data-
scope="user" data-show="current_count" data-scope="user"></button>
</div>
</div>
<!-- Футер. В низу находится контактная информация ->
<section class="bg-dark text-white" id="contact">
  <div class="container">
    <div class="row">
      <div class="col-lg-8 mx-auto text-center">
        <h2 class="section-heading"></h2>
        <hr class="my-4">
        <p class="mb-5">
          </p>
        </div>
      </div>
    </div>
    <div class="row">
      <div class="col-lg-4 ml-auto text-center">
        <i class="fa fa-phone fa-3x mb-3 sr-contact"></i>
        <p>8-800-000-0000</p>
      </div>
      <div class="col-lg-4 mr-auto text-center">
        <i class="fa fa-envelope-o fa-3x mb-3 sr-contact"></i>
        <p>
          <a href="/cdn-cgi/l/email-protection#bfdad2ded6d3ffdbd0d2ded6d191dcd0d2"><span
class="__cf_email__" data-
cfemail="a2c7cfc3bcce2c6cdcfc3bcc8cc1cdf">[email&#160;protected]</span></a>

```

```
</p>
</div>
</div>
</div>
</section>
<!--.. В самом низу находятся подгружаемые библиотеки bootstrap, jquery, JavaScript ->
  <script data-cfasync="false" src="/cdn-cgi/scripts/f2bf09f8/cloudflare-static/email-
decode.min.js"></script><script type="text/javascript" src="/js/jquery-3.1.1.min.js"></script>
  <script type="text/javascript" src="/js/bootstrap.bundle.min.js"></script>
  <!-- Plugin JavaScript -->
  <script type="text/javascript" src="/js/jquery.easing.min.js"></script>
  <script type="text/javascript" src="/js/scrollreveal.min.js"></script>
  <script type="text/javascript" src="/js/jquery.magnific-popup.min.js"></script>
  <!-- Custom scripts for this template -->
  <script type="text/javascript" src="/js/creative.js"></script>
  <script type="text/javascript" src="/js/script.js"></script>
```

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)

(Ф.И.О.)