

МИНОБРНАУКИ РОССИИ
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И АНАЛИЗА ДАННЫХ
КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СИСТЕМ

КУРСОВОЕ ПРОЕКТИРОВАНИЕ

Проектирование архитектуры системы поддержки
принятия решений для продвижения сайта

БИН-21-145358. 8847-с. 12. 000. КР

Студент
БИН-21-145358



И.И. Малиев

Руководитель,
Доктор. техн. наук,
Доцент



В.М. Гриняк

Нормоконтролер
Доктор. техн. наук,
Доцент



В.М. Гриняк

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВВГУ»)

Институт информационных технологий и анализа данных
Кафедра информационных технологий и систем

Индивидуальное задание
на производственную технологическую (проектно-технологическую) практику

Студенту гр. БИН-21-1 Малиеву Ивану Игоревичу

1 Составление нефункциональных требований (составление требований к системе и серверам; составление дизайн-требований на основе анализа целевой аудитории; составление требований к дашбордам на основе изучения потребностей в анализируемых данных для задач SEO)

2 Дизайн-проектирование системы (разработка макетов при помощи редактора Figma для экранов системы, соблюдая фирменные цвета компании, подходы к проектированию и руководствуясь логикой работы пользователя в системе)

3 Выбор СУБД и проектирование схемы базы данных для сервиса (схема базы данных должна отображать весь набор используемых данных и их декомпозиция в таблицы, на таблицах должна быть явно отображены зависимости и связи)

4 Проектирование схемы архитектуры сервиса (формирование схемы архитектуры с указанием движения потоков данных, взаимодействия микросервисов, баз данных и API)

5 Срок сдачи отчета на кафедру: 14.01.2025

Руководитель,
Доктор. техн. наук, Доцент

Задание получил:




Гриняк В.М.

Малиев И.И.

Содержание

Введение	5
1. Нефункциональные требования	6
1.1 Системные требования.....	6
1.2 Требования к дизайну	8
1.3 Требования к дашбордам	9
1.4 Роли пользователей в системе	11
2. Проектирование дизайна и прототипирование	13
2.1 Дизайн экрана “Личный кабинет”	13
2.1.1 Дизайн экрана “Мои семантические ядра”	13
2.1.2 Дизайн экрана “Добавить семантику”	14
2.1.3 Дизайн экрана “Мои дашборды”	15
2.1.4 Дизайн экрана “Мои настройки”	17
2.2 Дизайн страницы “Мониторинг”	20
2.2.1 Дизайн экрана с дашбордами	21
2.2.2 Дизайн экрана редактирования шаблонов	23
2.3 Дизайн страницы “Регистрация и авторизация”	25
2.3.1 Дизайн экрана Авторизации	25
2.3.2 Дизайн экрана Регистрация	25
3. Проектирование базы данных	27
3.1 Выбор инструментария	27
3.2 Схема базы данных.....	29
3.2.1 Схема потока данных	29
3.2.2 Схема БД для приложения.....	30
4. Проектирование схемы ПО	32
4.1 Разработка User Flow для модуля “Аналитика и мониторинг”.....	32
4.2 Разработка User Flow для модуля “Хранение семантических ядер”	33
4.3 Разработка схемы архитектуры сервиса.....	35
Заключение.....	37
Список использованных источников.....	38
Приложение А.....	40
Приложение Б	41
Приложение В.....	42
Приложение Г	43

Аннотация

В данной работе рассмотрен процесс проектирования системы поддержки принятия решений для продвижения сайтов. Определены нефункциональные требования к разрабатываемой системе, в том числе системные требования, UX/UI требования и требования к дашбордам. Также в работе приведены и описаны макеты модуля личного кабинета, мониторинга и авторизации. Для понимания структуры серверной части системы были разработаны схемы баз данных и проведен выбор используемой СУБД. В завершении, для формирования полной картины структуры сервиса, были проработаны пользовательские сценарии и сформирована схема архитектуры сервиса.

Введение

В современных условиях развития цифровой экономики и высокой конкуренции в онлайн-пространстве успех веб-ресурса напрямую зависит от его видимости в поисковых системах. SEO (Search Engine Optimization) – это комплекс мероприятий, направленных на повышение позиций сайта в результатах поиска, привлечение целевого трафика и улучшение взаимодействия с пользователями. Эффективная SEO-оптимизация требует глубокого анализа данных, работы с большими объемами информации и оперативного принятия решений на основе текущих показателей.

Целью данной работы является проектирование системы поддержки принятия решений для задач SEO-оптимизации сайта. Система должна объединять функциональные модули для сбора, анализа и визуализации данных, а также обеспечивать удобство работы специалистов и возможность адаптации под различные бизнес-цели.

В процессе работы над проектом были поставлены задачи: определить требования к системе, разработать ее архитектуру, спроектировать основные модули и интерфейсы, а также сформулировать рекомендации по ее дальнейшему развитию и внедрению.

Актуальность темы обусловлена потребностью бизнеса в качественных инструментах для повышения эффективности SEO-мероприятий [1]. Увеличивающийся объем данных, изменения в алгоритмах поисковых систем и растущая конкуренция требуют создания более интеллектуальных и адаптивных систем, которые помогут специалистам принимать обоснованные и оперативные решения.

Для достижения поставленной цели необходимо проанализировать и выполнить следующие задачи:

- определить перечень функциональных и нефункциональных требований к системе;
- разработать дизайн-макеты для всех страниц сервиса;
- разработать схему базы данных приложения и отдельных модулей;
- спроектировать схему логики работы приложения и его архитектуры.

Решение данных задачи позволит подготовиться к следующему этапу реализации проекта - разработке и интеграции сервиса во внутреннюю систему предприятия.

1. Нефункциональные требования

1.1 Системные требования

Прежде всего важно определить требования с точки зрения производительности системы. Среднее количество строк в семантическом ядре не более 1000, однако в объемах работы компании семантические ядра могут достигать сотни тысяч строк, из-за чего большинство существующих сервисов не могут обработать такие объемы данных в приемлемое время. Таким образом, важно определить следующие характеристики:

- скорость обработки данных должна быть не менее 100 запросов в секунду для обеспечения приемлемой скорости обработки дальше больших объемов данных;
- для операций с малыми объемами данных - до 10000 запросов - продолжительность выполнения задачи не должна превышать 10 минут;
- для базовых операций время отклика должно быть минимальным и стремиться к 1-2 секундам;
- на каждого пользователя необходимо выделить количество задач в размере количества доступных типов задач, чтобы каждый тип задач мог быть запущен каждым пользователем в текущий момент времени;
- одновременная работа пользователей должна быть более 500. Такое количество позволит обеспечить полное покрытие для всех сотрудников компании;
- задачи от разных пользователей должны обрабатываться в параллельных потоках;
- объемы хранилища для семантических ядер должны быть не менее 1 Гб данных на одного пользователя.

Следующий не менее важный аспект системных нефункциональных требований - возможность масштабирования сервиса. При этом важно отметить:

- горизонтальную масштабируемость, которая заключалась бы в возможности интеграции дополнительного функционала, API и интегрирование в общую внутреннюю систему предприятия [2];
- вертикальную масштабируемость, которая включала бы в себя увеличение объемов памяти, выделяемого на каждого конкретного пользователя, а также увеличение доступного количества одновременных сессий и параллельно выполняемых пользователями задач.

Далее рассмотрим такие критерии системы как надежность, доступность и безопасность. Внутри компании регламентированы строгие правила разграничения доступа и описаны методы доступа к корпоративной внутренней сети. Благодаря высокому уровню проработанности и защита внутренней сети самыми надежными методами, ключевым требованием к безопасности системы является соответствие стандартам информационной

безопасности внутри компании. Надежность системы в данном случае будет определена понятием доступности, то есть возможности получения доступа к сервису более чем 90% времени для любого сотрудника по стандартам SLA, а также уровень защиты от сбоев за счет создания резервных копий и логирования данных. Данный подход позволит обеспечить надежную, устойчивую и безопасную систему для внутренней работы на предприятии.

Далее рассмотрим требования к совместимости системы. В компании регламентированы правила комплектации рабочих мест сотрудников, а также существуют образы системы, которая устанавливается на рабочий персональный компьютер всех сотрудников. Тем не менее, в компании достаточно большое количество сотрудников, которые работают удаленно, что также необходимо учитывать при формулировании требований:

- сервис должен быть кроссплатформенным для всех наиболее популярных браузеров: Chrome, Firefox, Yandex, Edge, Safari - и одинаково корректно отображаться для каждого из них;

- возможности интеграции. Необходимо предусмотреть возможность интеграции системы разграничения доступа к ресурсу в соответствии с регламентом компании, а также учесть возможность интеграции посредством API с другими внутренними сервисами компании;

Последнее, что стоит рассмотреть в рамках нефункциональных требований к системе - это уровень технического обслуживания ресурса. Установим следующие требования:

- необходимо осуществлять круглосуточную техническую поддержку пользователей со временем реагирования до 1 часа для критических проблем и до 24 часов для стандартных запросов;

- необходимо регулярно обновлять и улучшать сервис, внедряя горизонтальные и вертикальные обновления;

- необходимо проработать и вести подробную систему логов работы системы и пользователей в системе для быстрого реагирования на внештатные ситуации в работе ресурса.

Соблюдения данных требований к системе обеспечат стабильную, удобную и безопасную работу в системе для всех сотрудников. Стоит отдельно отметить, что, поскольку сервис является интернет ресурсом, для взаимодействия с ним пользователю необходимо стабильное интернет соединение, а все остальные требования к рабочей станции уже соблюдены на рабочих местах сотрудников в соответствии с внутренними регламентирующими документами. Таким образом, предлагаемое решение обеспечит высокое качество предоставляемой услуги с минимальными требованиями к системе пользователя.

1.2 Требования к дизайну

Дизайн системы должен быть ориентирован на удобство пользователей, простоту восприятия и функциональность. Целевой аудиторией систему будут являться мужчины и женщины, сотрудники компании ООО “ФарПост”, в возрасте в среднем от 19 до 40 лет. Сотрудники, использующие сервис - это не только специалисты IT по направлениям SEO и аналитика, но и сотрудники отдела модерации контента, отдела маркетинга, управленческих и менеджерских должностей и даже сотрудники клиентских подразделений [3]. Таким образом, можно составить следующие требования к дизайну:

— интерфейс должен быть простым и минималистичным, избегая излишнего визуального шума. Важно, чтобы интерфейс был интуитивно понятным, с логичной структурой, позволяющей пользователям легко находить нужные функции и быстро осваивать систему [4]. Все элементы дизайна должны быть последовательными: одинаковый стиль кнопок, меню, шрифтов и цветов на всех страницах. При этом, не менее важно соответствовать корпоративному стилю компании: использовать соответствующие цвета, формы и пресеты форм и кнопок. Ключевой принцип дизайна в компании, в том числе и для внутренних сервисов - функциональность и минималистичность важнее всего;

— основная цветовая палитра включает нейтральные цвета для фона и акцентные цвета для интерактивных элементов, таких как кнопки и ссылки. Для выделения ошибок, предупреждений и успешных операций используются соответствующие цвета: красный, желтый и зеленый. Шрифты должны быть легкими для чтения, а текст – достаточно крупным, чтобы исключить напряжение для глаз, что особенно важно для офисных сотрудников;

— навигация по системе должна быть простой и удобной. Главное меню должно быть доступно на всех страницах, чтобы пользователи могли быстро переключаться между разделами. Интерфейс страниц для модуля “Хранение семантических ядер” должен включать элементы для поиска, сортировки и фильтрации;

— не менее важно отметить, что необходимо проработать адаптивность, однако нет необходимости прорабатывать систему для работы на мобильных устройствах. Таким образом, необходимо проработать адаптивность для минимального разрешения экранов ноутбуков, но не прорабатывать далее. Связано данное решение с тем, что система предназначена исключительно для работы на стационарных офисных, домашних компьютерах и корпоративных и личных ноутбуках сотрудников;

— особое внимание должно быть уделено визуализации данных [5]. Графики и таблицы должны быть интерактивными, позволяя пользователям увеличивать отдельные области, выбирать диапазоны дат и получать подсказки при наведении. Таблицы должны поддерживать сортировку, фильтрацию и возможность настройки отображаемых колонок.

Карты видимости должны использовать цветовую градацию для наглядного представления данных;

— для улучшения пользовательского опыта в системе должны быть реализованы всплывающие подсказки и уведомления. Подсказки помогут новым пользователям освоить интерфейс, а уведомления будут информировать о статусе задач или выполнении операции: ошибки или успешное завершение. Интерактивные элементы должны быть визуально привлекательными и реагировать на действия пользователя, в том числе изменяться по наведению и нажатию;

— важной частью интерфейса является функция экспорта данных. Процесс настройки отчетов должен быть простым, с возможностью выбора форматов: PDF или Excel.

Руководствуясь описанными принципами удастся создать универсальный, удобный и нативно понятный для пользователей дизайн. Важно при проектировании дизайна сделать акцент именно на простоту и качество UX составляющей, поскольку сервис является внутренним для компании и в первую очередь должен удовлетворять функциональным потребностям пользователя.

1.3 Требования к дашбордам

Описываем, какие нужно отображать метрики, какой должен быть флюу работы, какие метрики какими графиками должны отображаться, какие из графиков должны быть интерактивными.

Дашборды являются ключевым элементом системы для предоставления пользователям удобного доступа к аналитической информации и визуализации данных. Они должны быть интуитивно понятными, гибкими и наглядными, чтобы обеспечить эффективный мониторинг SEO-оптимизации и состояния семантических ядер [6]. Ниже описаны основные требования к дашбордам.

— информативность и структурированность метрик. Для обеспечения наиболее полной информативности графиков были выделены следующие ключевые метрики, которые необходимо визуализировать:

1. данные о сайте (органический трафик, позиции сайта, средние позиции сайта, CTR заголовков с соответствующими запросами);
2. графики тенденций (можно отнести как графики трафика, так и графики сезонности запросов и разделов сайта);
3. данные о семантическом ядре (изменение частотности, средние базовые, уточненные и точные частотности, позиции по семантике, популярные страницы по запросам из семантического ядра);

— качество визуализации данных. Наглядность графиков напрямую влияет на эффективность их интерпретации, в связи с чем крайне важно использовать различные варианты графического отображения (боксплоты, линейные графики, столбчатые диаграммы), а также отображать числовые параметры достаточно заметными и быстро считываемыми;

— интерактивность и персонализация дашбордов. Помимо этого, важным преимуществом будет настроить возможность настройки цветовой схемы, перестановки и масштабируемости виджетов, а также предусмотреть сохранение пользовательских локальных настроек отображения. Каждый визуальный элемент должен поддерживать интерактивность:

1. наведение курсора для получения детальной информации о точке данных;
2. возможность выбора временных диапазонов для анализа (например, сравнение недель, месяцев, лет).

— наличие фильтров и возможностей сегментации. Фильтры должны быть легко доступны и быстро применимы. Это важно, поскольку зачастую задачи требуют специфической выборки данных по конкретным полям. Пользователи должны иметь возможность фильтровать данные по различным параметрам:

1. по дате или диапазону дат (например, последние 7 дней, месяц, квартал);
2. по проектам, страницам сайта, кластерам или отдельным ключевым словам;
3. по группам ключевых слов (например, семантические ядра, категории);
4. по позициям (например, ТОП-3, ТОП-10, за пределами ТОП-50).

— настройка уведомлений и оповещений. Благодаря выбранным технологиям разработки, появляется возможность реализации оповещений для пользователей о резких изменениях каких-либо данных. Примеры метрик, которые могут быть подключены к экстренным оповещениям для своевременного реагирования специалистов:

1. снижение видимости сайта;
2. падение позиций ключевых слов;
3. аномальный рост частотности запросов в семантическом ядре;
4. устаревание семантического ядра (например, потеря актуальности запросов);

— возможности экспорта данных. Поскольку зачастую данные требуются для составления отчетов и презентаций, необходимо предусмотреть возможность быстрого экспорта графиков и данных во все популярные форматы: PDF, CSV, PNG. При этом при экспорте данные и графики должны сохранять пользовательские настройки;

— историчность данных. Поскольку компания существует много лет, для анализа данных важна их историчность и возможность сравнения периодов [7]. Помимо этого, для SEO применимость внесенных изменений можно оценить только спустя минимум 3 месяца, однако

зачастую полностью оценить эффективность можно только спустя год, в соответствующий сезон. Таким образом, важно предусмотреть возможность хранения данных за прошедшие периоды, а также акцент на временных и периодических изменениях в данных;

— возможности интеграции с другими сервисами компании для получения полных отчетов в одном месте. Интеграция с внутренними системами визуализации данных компании, базами данных соответствующих проектов, системами Google и Яндекс аналитики обеспечат более полную картину текущего состояния продукта: не только в отношении SEO оптимизации, но при этом и отследить изменения в соответствии с изменениями в SEO. Взаимодействие SEO и бизнес метрик позволят эффективно планировать ресурсы компании.

Дашборды в системе помогут более быстро и эффективно получать информацию о текущем состоянии SEO оптимизации сайта, а также сделать выводы о смежных проектах и направлениях развития компании. Это возможно, поскольку в электронной коммерции SEO изменения затрагивают продукт в целом, а не только отдельный его аспект. Помимо этого, многие работы по поддержке сайта могут стать значительно более эффективными за счет применения информации о текущем состоянии поисковой оптимизации сайта.

1.4 Роли пользователей в системе

Для эффективной работы и разделения рабочего пространства, в системе необходимо предусмотреть три типа ролей: “Редактор”, “Читатель” и “Администратор”. Данный вариант распределения ролей является оптимальным для данной системы, поскольку он позволит разграничить доступ среди различных пользователей, не переживая за повреждение или утерю данных. Рассмотрим подробнее каждую из ролей.

Роль “Редактор” является наиболее универсальной моделью и будет являться актуальной для всех сотрудников SEO отдела по умолчанию. Данная роль также может быть добавлена сотрудникам смежных отделов в том случае, если он обладает достаточными навыками для разработки и загрузки семантических ядер, а также создания и редактирования дашбордов.

Роль “Читатель” будет являться наиболее часто используемой в системе, поскольку будет присвоена всем сотрудникам смежных отделов по умолчанию. В данной роли сотрудник может просматривать данные и отправлять задачи, однако он не может добавлять свои семантические ядра, редактировать существующие, и также изменять и добавлять дашборды, за исключением локальных, собственных дашбордов и, соответственно, локальной библиотеки семантических ядер.

Роль “Администратор” наиболее объемная и нацелена в основном на добавление и редактирование прав других пользователей, разграничение видимости семантических ядер и

дашбордов. Таким образом, получится создать универсальный аккаунт, отвечающий за все остальные пользовательские кабинеты и их права доступа.

Данная ролевая модель уже используется в компании для текущих внутренних сервисов и является наиболее эффективной при совместной работе не только сотрудников одного отдела, но и сотрудников разных отделов.

Для каждой из ролей была составлена UML диаграмма, разделенная по модулям системы. UML (Unified Modeling Language) диаграмма — это графическое представление системы, которое используется для моделирования различных аспектов программного обеспечения. UML позволяет визуализировать структуру, поведение и взаимодействие компонентов системы. UML помогает разработчикам, аналитикам и другим специалистам четко представлять и документировать систему, упрощая процессы проектирования и разработки программного обеспечения. Так, диаграмма для модуля “Парсинг и семантический анализ” для пользователей с правами доступа “Читатель” приведена в приложении А, для пользователей с правами доступа “Редактор” и “Администратор” в приложении Б. Для модуля “Мониторинг и анализ данных” в приложении В, а для модуля “Хранение семантических ядер” в приложении Г.

2. Проектирование дизайна и прототипирование

2.1 Дизайн экрана “Личный кабинет”

Личный кабинет - это объединение личного пространства пользователя и настроек его профиля. Данный экран включает в себя следующие подразделы:

- личная библиотека семантических ядер пользователя. Уникальна вкладка тем, что на ней даже пользователь с ролью “Читатель” может добавлять семантические ядра, однако видны они будут только самому этому пользователю;

- индивидуальные дашборды. В данной вкладке отображаются дашборды по личным семантическим ядрам пользователя, то есть по тем семантическим ядрам, источником которых указан “Сам пользователь”;

- настройки. На данном экране пользователь может сменить пароль от своего личного кабинета. Это позволяет локально контролировать доступность кабинета каждому пользователю, без потребности обращения с заявкой в службу технической поддержки при каждой необходимости смены пароля;

- управление пользователями. Данный раздел доступен только пользователям с правами доступа “Администратор” и он позволяет удалять учетные записи пользователей из системы.

Рассмотрим макеты каждого из типа экранов в зависимости от роли пользователя.

2.1.1 Дизайн экрана “Мои семантические ядра”

На рисунке 1 представлен макет экрана “Мои семантические ядра” для пользователя с правами доступа “Читатель” и “Редактор”. Отметим, что экран дублирует экран “Библиотека семантических ядер”, однако по умолчанию в данном случае применен фильтр “Источник: Сам пользователь” без возможности отменить или снять данный фильтр для пользователя. Такой подход обеспечит разграничение личного и общего пользовательского пространства.

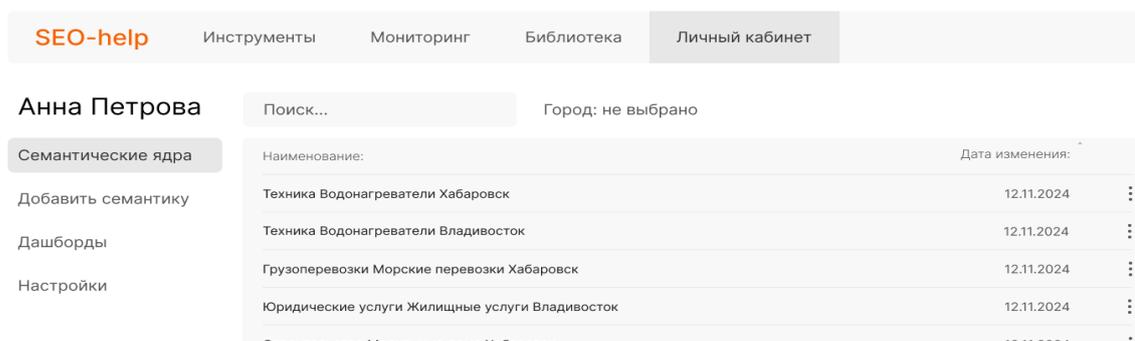


Рисунок 1 - Макет экрана “Мои семантические ядра” для пользователя с правами доступа “Читатель” и “Редактор” и “Администратор”

Каждый пользователь в таком случае также может осуществлять поиск, фильтрацию и сортировку списка, а также удалять семантические ядра вне зависимости от наличия роли “Администратор”.

2.1.2 Дизайн экрана “Добавить семантику”

Отличия для пользователей с правами доступа “Редактор” и “Читатель” заключаются в форме добавления семантических ядер. Форма для пользователей с правами доступа “Читатель” представлена на рисунке 2.

The screenshot shows a web interface for adding semantics. At the top, there is a navigation bar with the following items: "SEO-help", "Инструменты", "Мониторинг", "Библиотека", and "Личный кабинет". Below the navigation bar, the user's name "Анна Петрова" is displayed. The main form area contains several sections: "Семантические ядра" with a text input field containing "Водонагреватели Хабаровск"; a "Добавить семантику" button; "Дашборды" and "Настройки" sections; "Вариант загрузки фраз" with three radio button options: "Загрузить файл", "Ввести фразы", and "Прикрепить ссылку" (which is selected); "Ссылка на таблицу" with a text input field containing "Вставьте ссылку"; "Город" with a dropdown menu showing "Выберите город"; and a "Загрузить" button at the bottom.

Рисунок 2 - Макет страницы загрузки семантического ядра для пользователя с правами доступа “Читатель”

Ключевым отличием является вид формы загрузки семантического ядра, где помимо прочего пользователь может указать тип доступа к загружаемой семантике: общий или персональный доступ. В зависимости от выбранного варианта будет отличаться информация на странице “Мои дашборды”, а также загруженное ядро либо будет размещено только в личном кабинете пользователя, либо также и в общей библиотеке ресурсов. При этом, если ядро добавлено пользователем с правами доступа “Редактор”, оно может заранее быть задано с правами доступа “Общий доступ”, благодаря чему дашборды по нему будут отображаться не только в личном кабинете, но и для всех, кто может взаимодействовать с системой с любыми правами доступа. Данное разграничение необходимо для обеспечения качества библиотеки ресурсов и ограничения загрузки данных сотрудниками, обладающими достаточной квалификацией для гарантии корректности данных о семантических ядрах. На рисунке 3 представлен макет страницы “Добавить семантику” для пользователей с правами доступа “Редактора” и “Администратора”.

Анна Петрова

Семантические ядра

Добавить семантику

Дашборды

Настройки

Наименование

Водонагреватели Хабаровск

Вариант загрузки фраз

Загрузить файл Ввести фразы Прикрепить ссылку

Ссылка на таблицу

Вставьте ссылку

Город

Выберите город

Настройки доступа

Локальный доступ Общий доступ

Загрузить

Рисунок 3 - Макет экрана добавления семантического ядра для пользователей с правами доступа “Редактор” и “Администратор”

Ключевым отличием является наличие элементов типа радиобатон для выбора настроек доступа: локальный или общий доступ. В данном случае пользователь может выбрать только один из двух предложенных вариантов.

2.1.3 Дизайн экрана “Мои дашборды”

Страница экрана “Мои дашборды” для пользователя с правами доступа “Читатель” приведена на рисунке 3.

Анна Петрова

Семантические ядра

Добавить семантику

Дашборды

Настройки

Проект: Farpost Период: месяц Детализация: по дням

Наименование семантического ядра: Водонагреватели Владивосток Фарпост

Позиции

Частотности

Мах частотность

Ср. частотность

Мах позиция

Ср. позиция

Рисунок 3 - Макет страницы “Мои дашборды” для пользователя с правами доступа “Читатель”

Ключевым отличием данного экрана является отсутствие фильтра по типу доступа. Для пользователя отображаются дашборды только по тем семантическим ядрам, которые добавлены в его личную библиотеку. В то же время для пользователей с ролями доступа “Редактор” и “Администратор” доступен фильтр “Доступ”, который определяет вариант доступа семантического ядра: локальный или общедоступный. При выборе локального доступа отображаются только личные пользовательские загруженные семантические ядра, а при выборе общедоступного фильтра, будут отображены вообще все загруженные в систему семантические ядра, за исключением локальных семантических ядер других пользователей. Данный вариант разделения позволит сотрудникам создавать и локально проводить мониторинг информации по необходимым им ядрам. При этом для сотрудников, которые обладают достаточным уровнем квалификации, также будет доступна и возможность наполнения библиотеки ресурсов прямо из личного кабинета, но также и даст возможность размещать некоторые ресурсы только в своем локальном хранилище. На рисунке 4 изображен макет страницы “Мои дашборды” для пользователей с правами “Редактор” и “Администратор”.

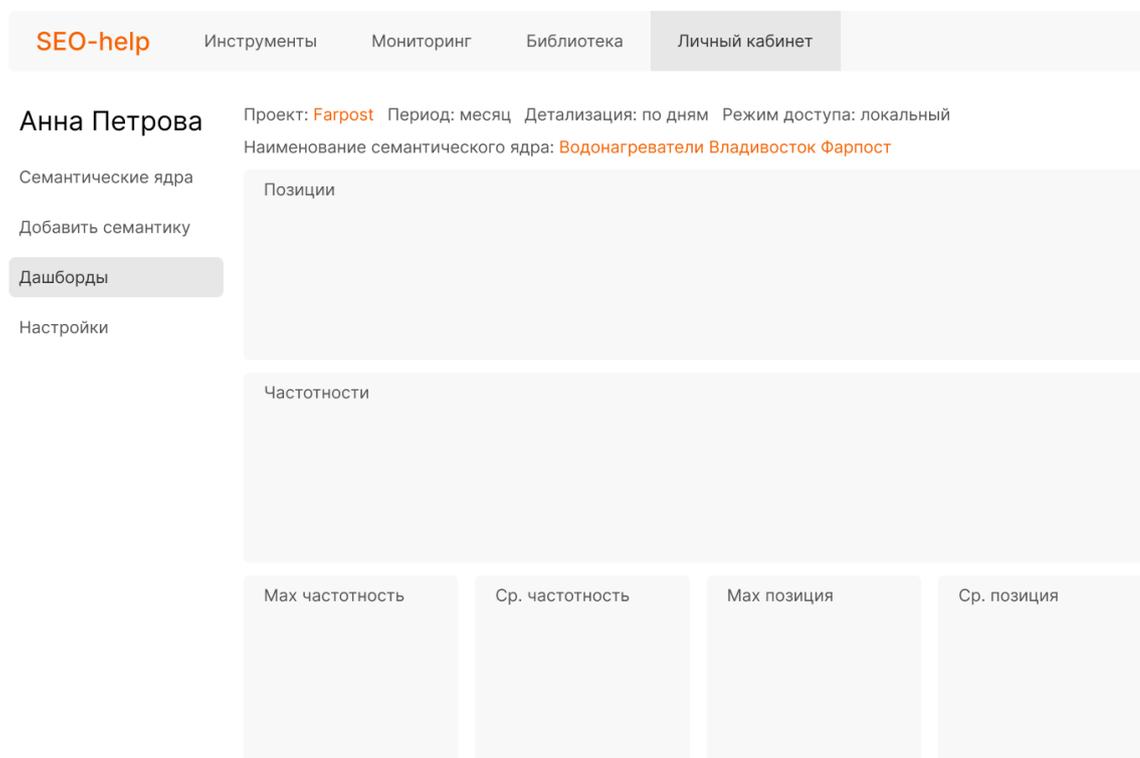


Рисунок 4 - Макет экрана “Мои дашборды” для пользователей с правами доступа “Редактор” и “Администратор”

Фильтр “Режим доступа” включает в себя два пункта: локальный режим доступа и общедоступный. В первом случае будут отображаться только дашборды по семантическим

ядрам с пометкой “Локальный”, а во втором случае - только дашборды для семантических ядер с пометкой “Общедоступный”.

2.1.4 Дизайн экрана “Мои настройки”

Следующий экран в личном кабинете - это экран “Настройки”. Данный экран выглядит одинаково для пользователей с правами доступа “Читатель” и “Редактор”. На данном экране есть поле “Имя”, в котором по умолчанию отображается имя пользователя в системе и “Почта”, которое содержит информацию о корпоративной почте пользователя. Также есть поле “Пароль”. Пользователь может сменить пароль прямо внутри системы. Для смены пароля пользователю потребуется ввести старый пароль, затем новый пароль с подтверждением. Поскольку доступ к системе будет обеспечен за счет внутренней системы VPN и выдаваемых каждому сотруднику сертификатов, что строго регламентируется внутренними требованиями безопасности компании, дополнительных методов подтверждения личности не требуется - таких как номер телефона или электронная почта. Визуализация макета страницы “Настройки” для пользователей с правами доступа “Читатель” и “Редактор” изображена на рисунке 5.

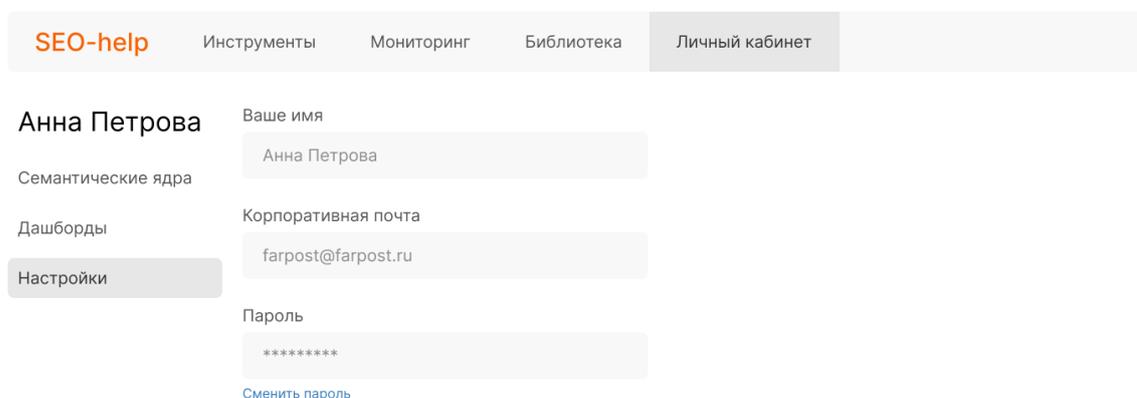


Рисунок 5 - Макет страницы “Настройки” для пользователей с правами доступа “Читатель” и “Редактор”

При смене пароля пользователь переходит на экран “Смена пароля”. Визуализация макета данного экрана представлена на рисунке 6.

The screenshot shows a user profile page for 'Анна Петрова'. The navigation menu includes 'SEO-help', 'Инструменты', 'Мониторинг', 'Библиотека', and 'Личный кабинет'. The left sidebar contains 'Семантические ядра', 'Дашборды', and 'Настройки'. The main content area is titled 'Смена пароля пользователя' and contains three password input fields: 'Старый пароль', 'Новый пароль', and 'Подтвердите новый пароль'. All fields contain masked characters (*****). A 'Сменить пароль' button is located at the bottom.

Рисунок 6 - Макет экрана “Смена пароля” для всех ролей пользователей

Для смены пароля необходимо предусмотреть ряд ситуаций, при которых пароль не будет успешно изменен. Такими ситуациями являются:

- неверный ввод текущего пароля;
- совпадение нового пароля и старого пароля;
- несовпадение нового пароля и повторения введенного пароля.

Визуализация макета с ошибками заполнения соответствующих полей для каждой из приведенных ситуаций представлена на рисунке 7.

This screenshot shows the same 'Change Password' form as Figure 6, but with error messages. The 'Старый пароль' field has a red asterisk error message. Below it, the text 'Неверный пароль' is displayed in red. The 'Новый пароль' field has a red asterisk error message, and below it, the text 'Старый и новый пароль не должны совпадать' is displayed in red. The 'Подтвердите новый пароль' field has a red asterisk error message, and below it, the text 'Пароли не совпадают' is displayed in red. The 'Сменить пароль' button remains at the bottom.

Рисунок 7 - Макет ошибок при смене пароля

Однако в случае успешной смены пароля пользователя возвращает на экран “Настройки” и отображается системное уведомление об успешной смене пароля. Визуализация данного уведомления представлена на рисунке 8.

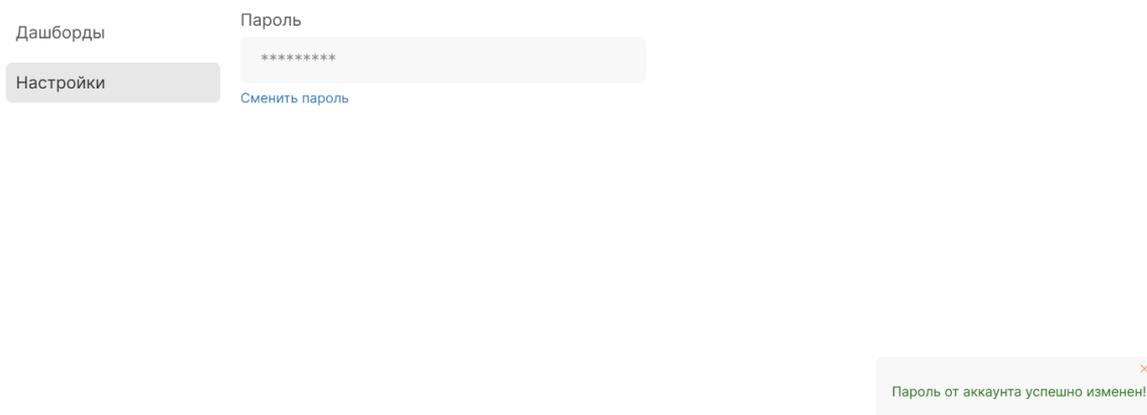


Рисунок 8 - Макет системного уведомления об успешной смене пароля

Основным отличием кабинета администратора является наличие возможности управления аккаунтами пользователей, а именно - удалять неактивные аккаунты или аккаунты сотрудников, которые больше не работают в компании. Необходимо предусмотреть возможность локального удаления аккаунта пользователей из системы в том случае, если сотрудник больше не может иметь доступа к системе. Помимо этого, на данной странице у администратора есть возможность регулировать уровень прав доступа для пользовательских аккаунтов. По умолчанию при регистрации любому пользователю выдается тип доступа “Читатель”. Далее изменить уровень прав доступа может только администратор по запросу сотрудника. Помимо этого, администратор также имеет возможность изменить пароль от своего собственного аккаунта. Визуализация макета страницы “Управление” для пользователей с правами доступа “Администратор” представлена на рисунке 9.

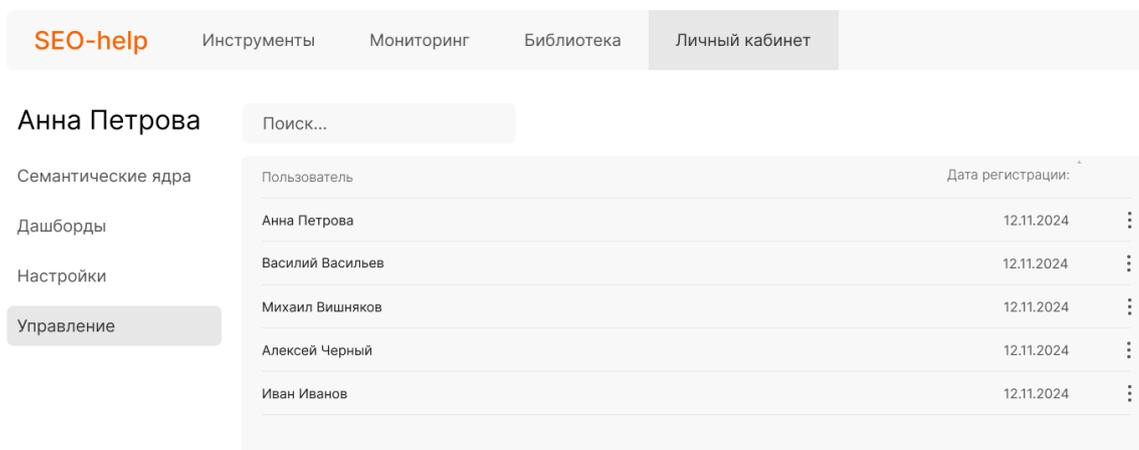


Рисунок 9 - Макет экрана “Управление” для пользователей с правами доступа “Администратор”

На данном макете администратор может искать конкретного пользователя по его никнейму, удалять пользователей и просматривать о них подробную информацию. Макеты

всплывающих окон для работы с конкретной записью о пользователе представлены на рисунке 10.

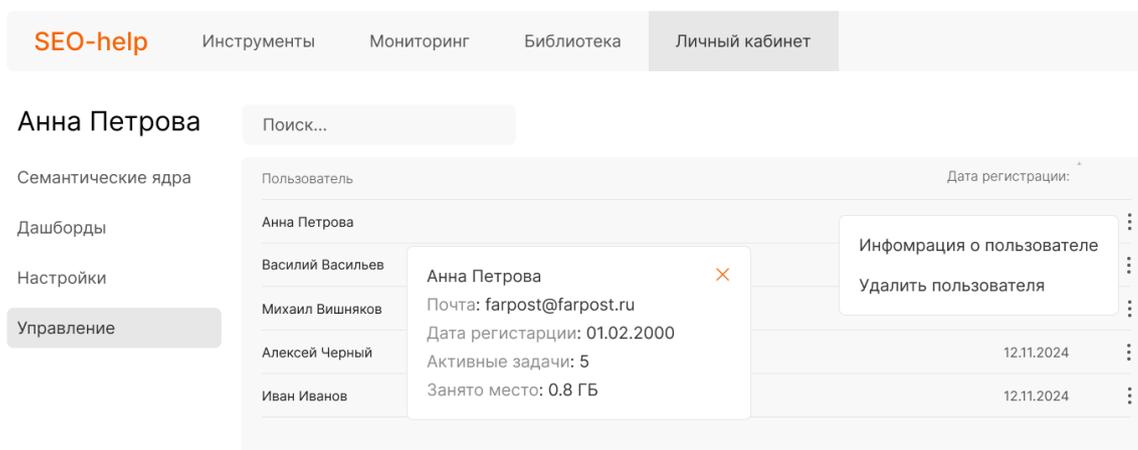


Рисунок 10 - Макет всплывающих окон “Действия” и “Подробная информация” на экране “Управление”

В всплывающем окне “Действия” при нажатии на “Информация о пользователе” можно просмотреть подробную информацию в всплывающем окне “Подробная информация”. На данном окне перечислена следующая информация о пользователе:

- имя пользователя в системе;
- корпоративная почта;
- дата регистрации пользователя;
- количество активный задач у пользователя;
- количество занятого пользователем места в личном хранилище.

Данный набор информации исчерпывающий для однозначного определения сотрудника, даже если два сотрудника имеют одинаковое имя и отчество, за счет уникального адреса почты, а также позволяет оценить затраченные пользователем ресурсы системы.

Таким образом, пользователем с правами доступа “Администратор” может достаточно гибко взаимодействовать с системой и управлять учетными записями пользователей, при этом не нарушая их конфиденциальности.

2.2 Дизайн страницы “Мониторинг”

Экран “Мониторинг” - это экран, на котором отображаются дашборды. По умолчанию отображается дашборд самой популярной семантики и все фильтры установлены в значение по умолчанию. Далее пользователь может взаимодействовать с данным экраном. Для пользователей с ролями “Читатель” и “Редактор” функционал практически не отличается, однако для пользователей с правами доступа “Администратор” появляется дополнительная

возможность редактирования шаблонов дашбордов и написания новых запросов. Рассмотрим данные два типа экранов в соответствии с ролями пользователей.

2.2.1 Дизайн экрана с дашбордами

На данном экране отображаются графики и фильтры. Фильтры включают в себя выбор проекта, выбор периода отображения, выбор параметров детализации графиков и непосредственно выбор анализируемого семантического ядра. Визуализация данного макета представлена на рисунке 11.

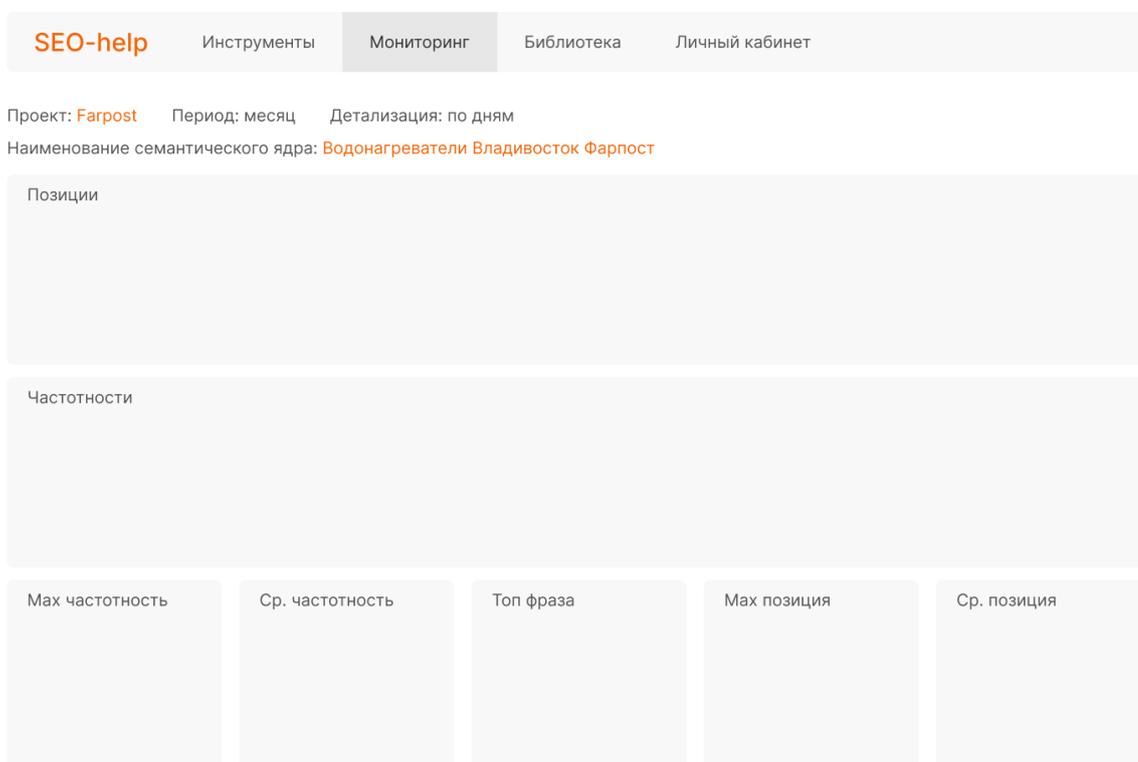


Рисунок 11 - Макет страницы “Дашборды” для пользователей с правами доступа “Читатель” и “Редактор”

Небольшие отличия для пользователей данных ролей можно наблюдать в модальном окне выбора семантического ядра. Для пользователей с правами доступа “Редактор” также доступна функция “Добавления семантики”, которая ведет на страницу “Добавить семантическое ядро”, в то время как у пользователей с правами доступа “Читатель” данная функция отсутствует. Данные отличия обусловлены уровнем прав доступа для соответствующих пользователей. Быстрая загрузка семантического ядра для редакторов позволит не прерывать пользовательский путь, а продолжать его с переходом на соответствующий экран. При таком подходе не возникнет необходимости формировать дополнительное модальное окно для загрузки, но при этом позволит повысить качество UX для данного модуля системы.

На рисунке 12 представлена визуализация макета для пользователей с правами доступа “Редактор”.

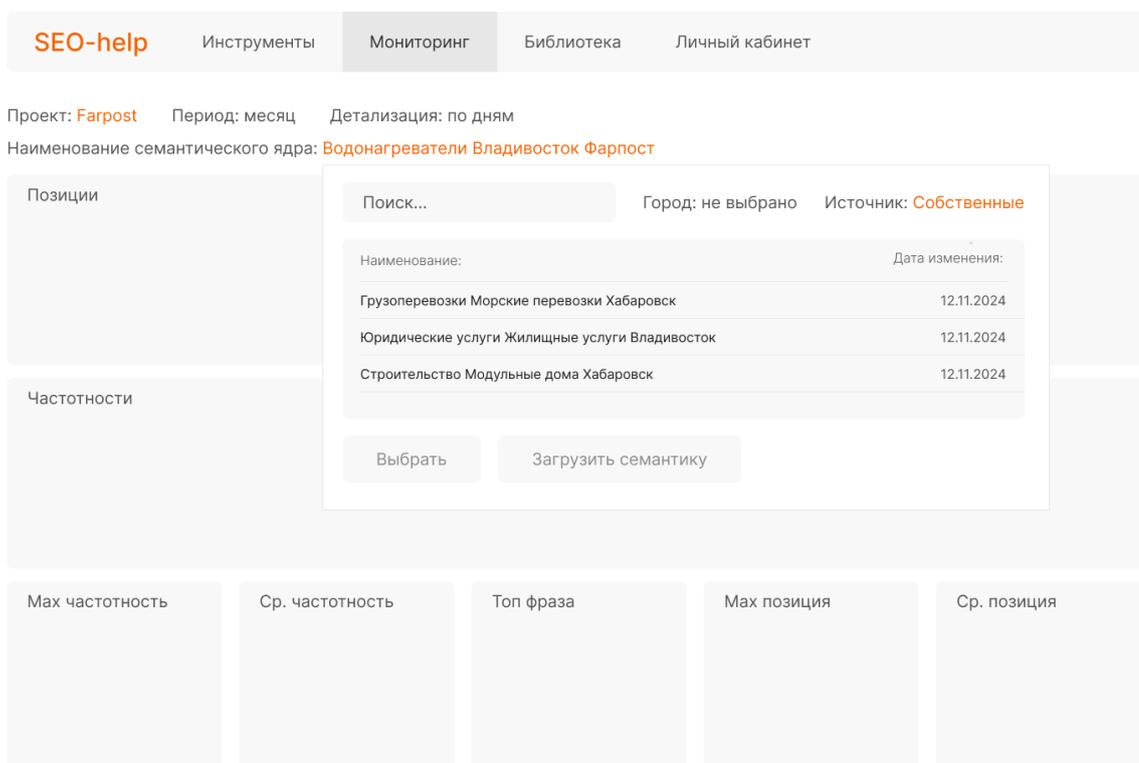


Рисунок 12 - Макет всплывающего окна выбора семантического ядра для пользователей с правами доступа “Редактор”

На рисунке 13 визуализирован макет того же всплывающего окна, но для пользователей с правами доступа “Читатель”. Поскольку они не могут загружать семантические ядра, на этом окне нет дальнейшего пути в загрузку семантики.

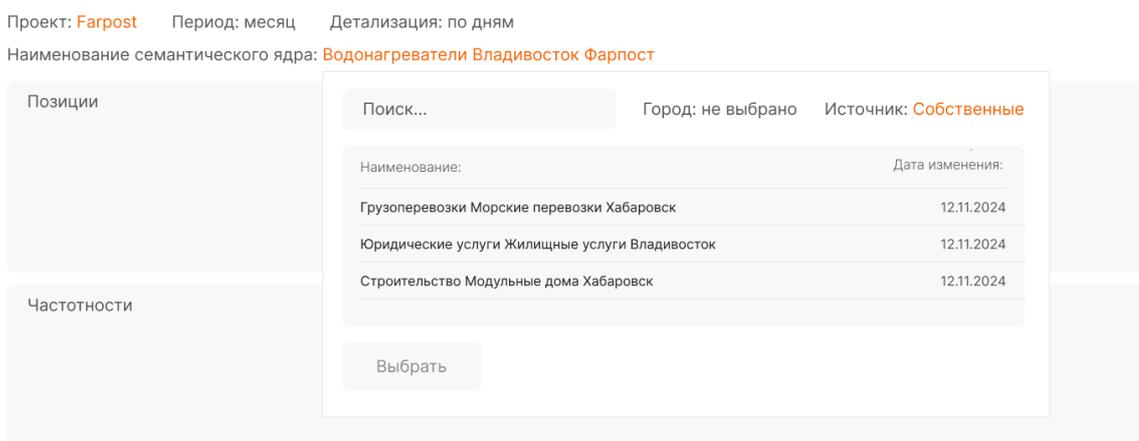


Рисунок 13 - Макет всплывающего окна выбора семантического ядра для пользователей с правами доступа “Читатель”

Ключевым отличием дизайна данного экрана для пользователей с правами доступа “Администратор” является наличие функционала редактирования шаблонов дашбордов. Это необходимо, поскольку информация о семантических ядрах динамическая и может дополняться новыми полями: например, может быть добавлена информация о позициях

сайтов-конкурентов, собрана дополнительная частотность или вовсе добавлены новые ключевых фразы. Таким образом, в различные моменты времени может потребоваться изменение запросов для построения дашбордов, удаление некоторых виджетов, их перемещение или изменение формата их входных данных. Для этого реализована кнопка перехода на дополнительный экран с редактированием, который будет подробнее описан далее. Для остальных пользователей данный функционал скрыт, поскольку при недостаточном уровне квалификации специалиста могут быть предоставлены в общий доступ некорректные данные, которые могут влиять на принятие различных бизнес-решений, стратегий продвижения и развития различных продуктов. Макет экрана “Мои дашборды” для пользователей с правами доступа “Администратор” представлено на рисунке 14.

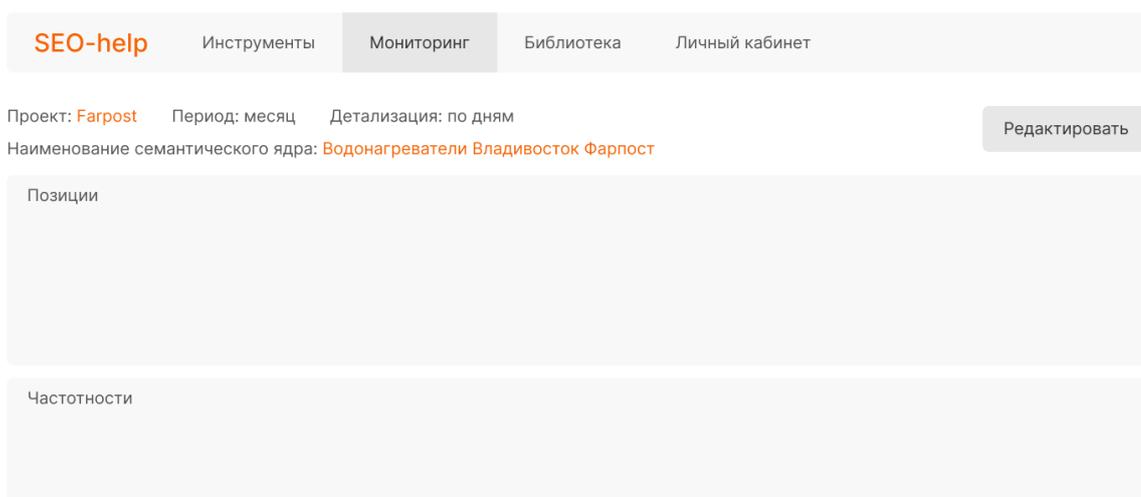


Рисунок 14 - Макет экрана “Мои дашборды” для пользователей с правами доступа “Администратор”

Таким образом, у администратора появляются дополнительные возможности для масштабирования возможностей дашбордов.

2.2.2 Дизайн экрана редактирования шаблонов

Данный экран доступен только пользователям с правами доступа “Администратор”. На нем доступно два действия: добавление новых виджетов и редактирование уже существующих. При добавлении нового виджета необходимо прописать запрос и после этого добавить виджет. он появится на странице “Мои дашборды”. Визуализация дизайна для данного экрана представлена на рисунке 15.

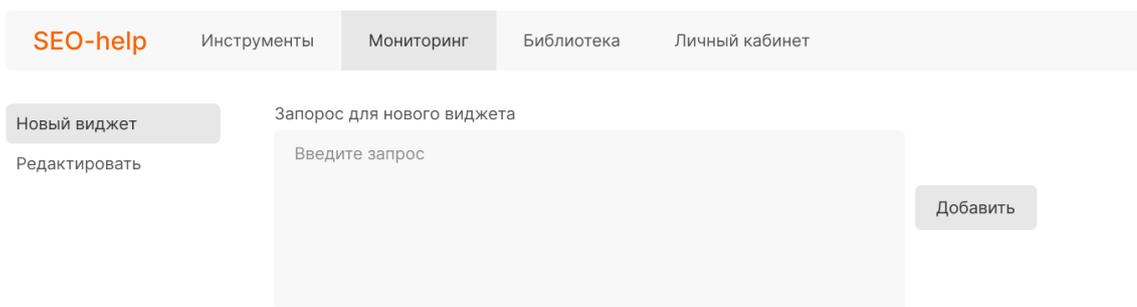


Рисунок 15 - Макет экрана “Добавить новый виджет” на странице шаблонов дашбордов

Вторая вкладка на данной странице — это редактирование текущих виджетов. Представляет собой набор полей, каждое из которых относится к конкретному виджету. Каждое поле можно сохранить после изменения или удалить, если данное поле более не актуально. Визуализация данного экрана представлена на рисунке 16.

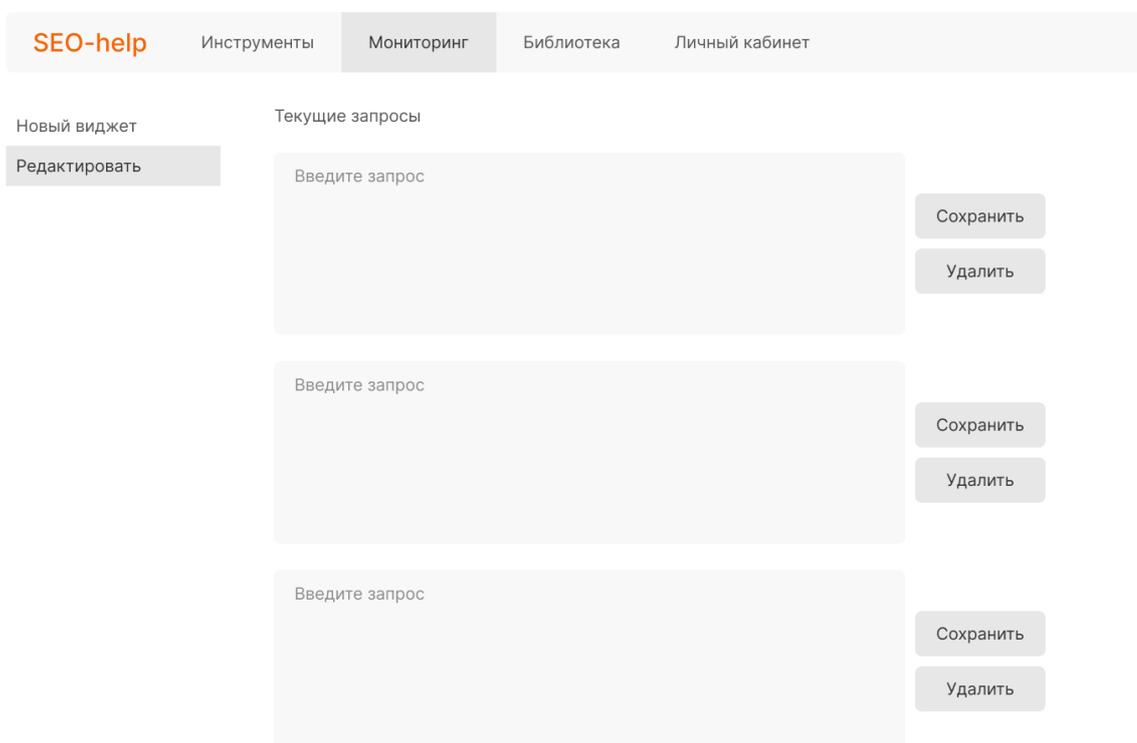


Рисунок 16 - Макет экрана “Редактирование виджетов” на странице редактирования шаблонов дашбордов

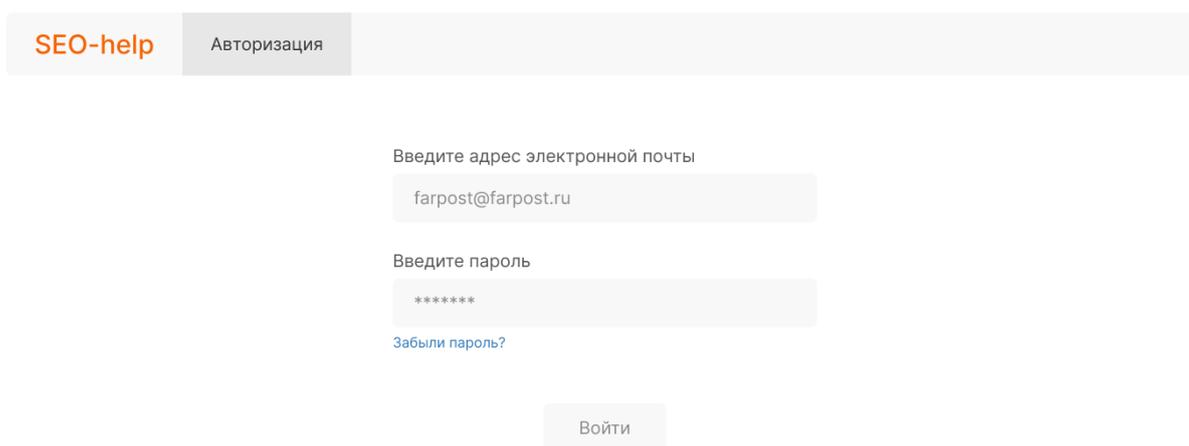
Таким образом, администратору предоставляется инструмент быстрого редактирования визуализации дашбордов без необходимости редактирования исходного кода программы.

2.3 Дизайн страницы “Регистрация и авторизация”

Экраны регистрации и авторизации универсальны для всех ролей пользователей и позволяют создавать и входить в учетные записи. Рассмотрим каждый из экранов подробнее.

2.3.1 Дизайн экрана Авторизации

Экран авторизации представляет собой способ входа в приложение в свою учетную запись с использованием корпоративного адреса электронной почты и пароля. Данный набор будет уникальным для каждого пользователя, поскольку корпоративная почта учитывает вероятность одинаковых имен. Помимо этого, корпоративная почта является гарантией санкционированности доступа сотрудников в систему. Визуализация данного экрана представлена на рисунке 16.



Макет страницы авторизации. Вверху слева находится ссылка "SEO-help" (оранжевый текст), а рядом с ней — заголовок "Авторизация" (серый фон). Основное поле ввода содержит текст "Введите адрес электронной почты" и значение "farpost@farpost.ru". Ниже — поле ввода пароля с текстом "Введите пароль" и маской "*****". Под полем пароля есть ссылка "Забыли пароль?". В центре находится кнопка "Войти".

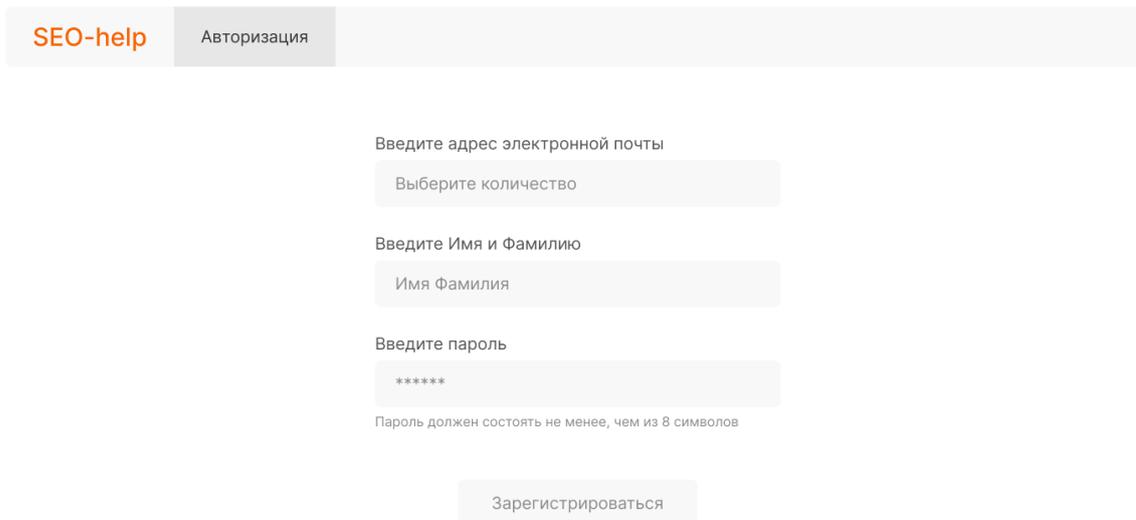
Рисунок 16 - Макет страницы авторизации

2.3.2 Дизайн экрана Регистрация

Экран регистрации позволяет создавать новые учетные записи для новых сотрудников. Регистрация в веб-сервисах необходима для того, чтобы пользователи могли получить доступ к персонализированным функциям и услугам. Во-первых, она позволяет создать уникальную учётную запись, которая хранит данные о пользователе, такие как имя, контактная информация, настройки и история активности. Во-вторых, регистрация даёт возможность обеспечить безопасность: пользователи могут устанавливать пароли для защиты своей учетной записи. Также регистрация помогает управлять доступом и предоставлять функции, требующие индивидуальных настроек или личных данных, например, сохранение покупок, отслеживание заказов или создание контента. Для регистрации пользователям потребуется заполнить три поля:

- никнейм пользователя, состоящий из его имени и фамилии;
- корпоративный адрес электронной почты для уникальной идентификации пользователей и дальнейшей авторизации при помощи неё в системе;
- пароль с ограничением на минимальную длину в 8 символов.

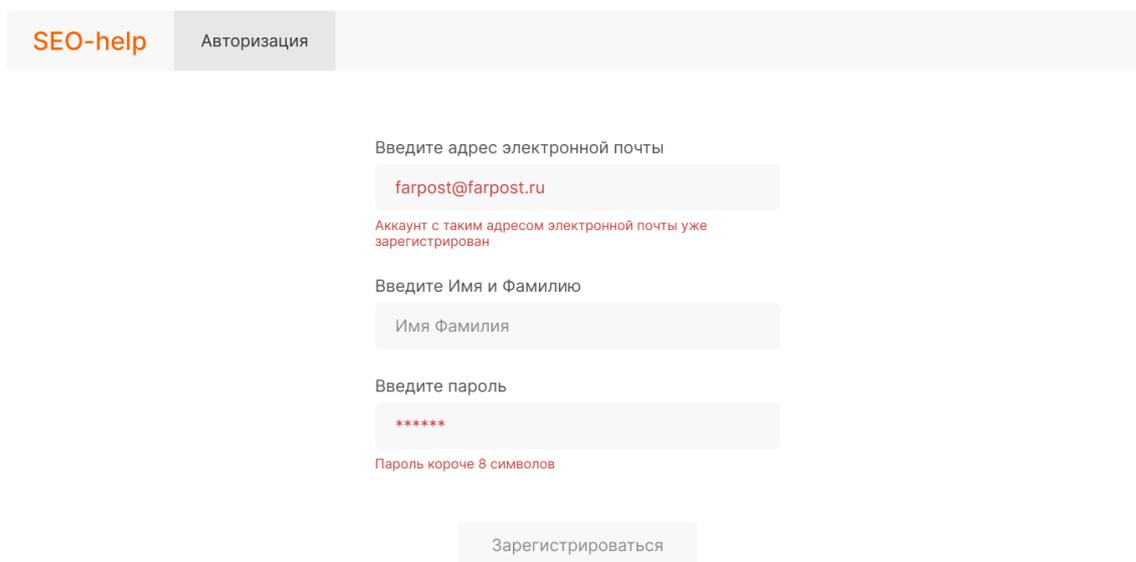
Такой набор данных позволит обеспечить безопасность и уникальность каждого пользователя с точки зрения модерации пользователей. Визуализация экрана регистрации представлена на рисунке 17.



The screenshot shows a registration form with a header containing 'SEO-help' and 'Авторизация'. The form includes three input fields: 'Введите адрес электронной почты' with a placeholder 'Выберите количество', 'Введите Имя и Фамилию' with a placeholder 'Имя Фамилия', and 'Введите пароль' with a placeholder '*****'. Below the password field is a note: 'Пароль должен состоять не менее, чем из 8 символов'. A 'Зарегистрироваться' button is at the bottom.

Рисунок 17 - Макет страницы регистрации

При регистрации пользователь может быть не зарегистрирован успешно в случае, если на указанную почту уже есть зарегистрированный аккаунт или если пароль не удовлетворяет требованию системы. Визуализация макета с обозначенными ошибками представлен на рисунке 18.



The screenshot shows the same registration form as in Figure 17, but with error messages. The email field contains 'farpost@farpost.ru' and has a red error message: 'Аккаунт с таким адресом электронной почты уже зарегистрирован'. The password field contains '*****' and has a red error message: 'Пароль короче 8 символов'. The 'Зарегистрироваться' button is still present.

Рисунок 18 - Макет экрана регистрации с ошибкой

3. Проектирование базы данных

3.1 Выбор инструментария

На текущий момент на рынке представлен достаточно широкий список СУБД, направленных на решение задач аналитики. Однако многие из них заточены под определенный спектр задач. Поэтому необходимо определить основные критерии, которые в большей степени повлияют на итоговый выбор. Для разрабатываемого приложения определим следующие критерии [8, 9]:

- аналитика в реальном времени. Проектируемое приложение должно быстро обрабатывать и агрегировать данные, чтобы соответствовать заявленным системным требованиям;

- работа с большим объемом данных. Размер семантических ядер может варьироваться от задач. Одно такое ядро может содержать как сотню записей, в случае маленьких одностраничных сайтов, так и сто миллионов, если ядро содержит информацию о постоянно меняющемся рынке (например, сайт с автомобилями). Предполагается хранение всех фраз в нем, а также история их изменений, для изучения влияний внесенных изменений;

- расширенный список аналитических функций. Помимо хранения большого объема данных, необходимо эти данные анализировать. Зачастую СУБД имеют скромный набор доступных функций, ограничиваясь средним и стандартным отклонением. Искомая база данных должна предоставлять возможность изучения данных с разных сторон, начиная с изучения медианы и ранжирования данных, заканчивая построением сложных воронок;

- база данных должна работать локально. Чтобы избежать проблем с доступностью облачных решений, база данных должна быть развернута локально для конкретного предприятия. Это позволит как ускорить скорость получения информации, так как не придется учитывать дополнительное время на передачу информации, так и усилить безопасность, за счет ограничения ее в рамках локальной сети.

Опираясь на требования, предъявляемые к базе данных, из списка рассматриваемых баз будут исключены облачные решения, такие как Amazon Redshift и Google BigQuery [10]. Последняя хоть и работает в связке с Google Analytics [11], для предоставления данных об использовании сайта и, в частности, SEO-данных, например, данные о трафике, о пользователях, о пользовательских сценариях, о паттернах взаимодействия пользователей с сайтом и различных других метриках, является отдельным инструментарием, в связи с чем имеет право быть оценено в рамках данного конкурентного анализа. Рассмотрим следующие СУБД, представленные в таблице 1. В данной таблице каждая СУБД оценена от 1 до 5, где 1 -

это полное несоответствие критерию в рамках разработки, а 5 - это полное соответствие критерию в рамках работы.

Таблица 1 - Конкурентный анализ СУБД

Параметр сравнения	MySQL	PostgreSQL	Greenplum	ClickHouse
Аналитика в реальном времени	1	2	3	5
Работа с большими объемами данных	2	3	5	5
Аналитические функции	1	3	5	5
Нетребовательность к ресурсам	5	5	3	4

Изучив представленные СУБД, можно сделать следующие выводы:

— система управления PostgreSQL хоть и имеет возможность работать как аналитическая СУБД, делается это через отдельные расширения, что усложняет обслуживание, а также ограничивает функционал из-за отсутствия специализации на аналитике [12];

— система управления Greenplum - хорошая СУБД для анализа данных, но избыточна для данного проекта [13]. Она заточена на работу с еще большими объемами данных, которые распределены по различным кластерам. Из-за чего возникают задержки в предоставлении информации, что критично для аналитики в реальном времени. Также данная СУБД достаточно требовательна к предоставляемым ресурсам, что может стать проблемой для некоторых предприятий;

— система управления ClickHouse демонстрирует хорошие результаты в рассматриваемых категориях. Хоть он и не лишен недостатков, таких как сложности в выполнении команд Update/Delete, на решения задач поставленных в этом проекте это никак не отразится [14]. Из дополнительных преимуществ, ClickHouse является свободно распространяемым программным обеспечением, а также разрабатывается отечественной компанией Яндекс, что минимизирует риски в прекращении его поддержки;

— система управления MySQL продемонстрировал достаточно плохие результаты. Однако это лишь демонстрирует, выбор базы данных зависит от поставленных задач: MySQL

плохо справляется с аналитическими задачами, однако в задачах, связанных с обработкой сложных транзакций, данная СУБД продемонстрировала бы достойные результаты.

Основываясь на приведенном сравнении, для работы над проектом планируется использовать СУБД ClickHouse, поскольку она имеет явные преимущества в сравнении с конкурентными аналогами, а также внутренние системы компании используют и могут быть легко интегрированы именно с этой СУБД.

3.2 Схема базы данных

3.2.1 Схема потока данных

Для проведения дальнейших работ по проектированию приложения, необходимо понять, с какими данными планируется вестись работа. Для этого в целом рассмотрим, как потоки данных есть в приложении. Для этого рассмотрим два основных сценария: сбор данных (рисунок 19) и анализ данных (рисунок 20).



Рисунок 19 - Передача данных при сборе семантического ядра

Изучив поток данных при сборе семантического ядра, можно выделить следующие объекты:

- фраза. Является частью семантического ядра. Каждая фраза, после обработки поисковым движком, получает свои характеристики (позиция, частота);
- семантическое ядро. Совокупность фраз, объединенных по смысловому содержанию страницы. Формируется на основе данных, полученных из поискового движка [15];
- задача. Необходимо также хранить историю задач, чтобы была возможность повторного выполнения определенных действий, для получения исторических данных, которые позволят отслеживать эффективность внедренных в проект изменений.

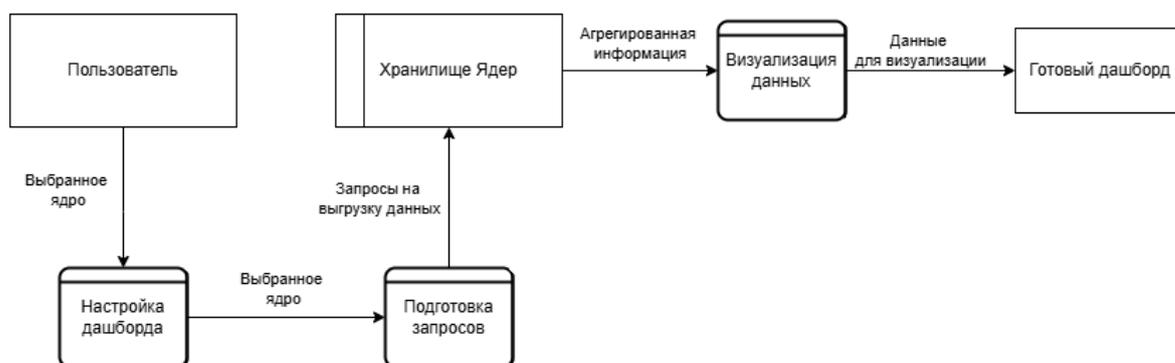


Рисунок 20 - Передача данных при построении дашборда

В свою очередь работа с дашбордами не добавляет новую информацию в базу данных, а работает уже с существующей, которая была добавлена во время сбора семантического ядра. Пользователь выбирает, по какому ядру ему необходимо подготовить информацию, после чего формируется список запросов на агрегацию, отправляемых в аналитическую базу данных. Далее на основе агрегированных данных производится визуализация, которая и демонстрируется пользователю.

3.2.2 Схема БД для приложения

Предварительно изучив, какими данными будет оперировать разрабатываемое приложение, можно приступить к формированию таблиц баз данных и созданию связей между данными [16]. Данная структура позволит хранить информацию задачах, ядрах и фразах, а также их взаимосвязь [17], что в дальнейшем позволит организовать систему поиска и фильтраций, необходимую для работы дашбордов [18].

В текущей схеме можно выделить три основные таблицы:

- `phrase`. Данная таблица хранит только информацию о конкретной фразе. Однако её расширяют две другие таблицы - `frequency` и `position`, которые хранят информацию о частоте поиска фразы и позиции в поиске у конкретного сайта. Данная структура позволит быстро получить доступ к конкретным фразам, а также получить историческую информацию;

- `semantic_core`. Данная таблица нужна для выделения конкретных ядер и определения принадлежности фразы к конкретному ядру, посредством связи многие ко многим с таблицей `phrase`;

- `task`. Фактически является служебной таблицей, для определения задачи и хранения истории обновлений. Для работы необходимо реализовать ряд словарей, которые определяют будущий функционал;

- `task_type` - определяет тип выполняемой задачи (сбор ядра, сбор позиций, сбор частотностей);

— `search_engine` - указывает, какой поисковый движок будет задействован при выполнении задачи;

— `region` - выделяет область, для которой будет собираться информация (Владивосток, Забайкальский край, Ташкент).

Описанная схема позволяет наиболее эффективно хранить и агрегировать данные. Схема базы данных — это структура, которая описывает организацию и отношения между данными в базе данных. Она включает в себя таблицы, поля, типы данных, а также связи между таблицами, такие как первичные и внешние ключи. Схема определяет, как данные будут храниться, структурироваться и обрабатываться, обеспечивая целостность и эффективный доступ к информации. Важно, что схема не содержит самих данных, а лишь их описание и правила взаимодействия между различными элементами базы. Преимущества схемы базы данных включают в себя несколько ключевых аспектов. Визуализация данной схемы представлена на рисунке 21.

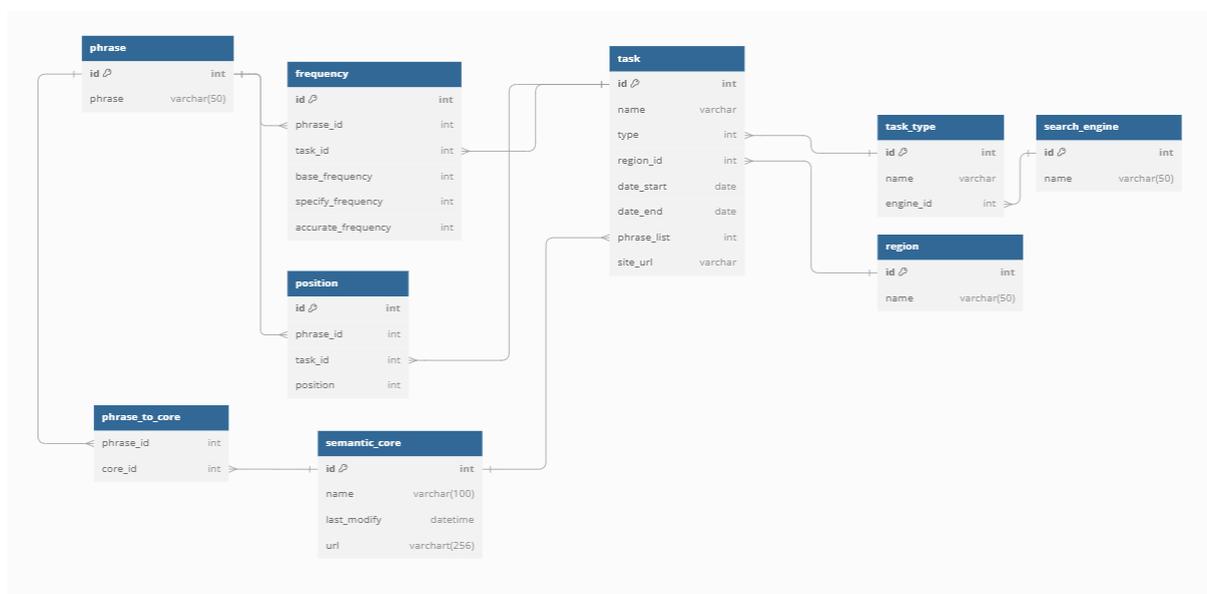


Рисунок 21 - Схема баз данных приложения

Данная структура данных позволит обращаться как к совокупности данных (распределение фраз и ядер), так и к конкретным записям таблицы, что позволит хранить исторические данные, не перезаписывая имеющиеся.

4. Проектирование схемы ПО

4.1 Разработка User Flow для модуля “Аналитика и мониторинг”

Для формирования полного представления о логике работы системы, были составлены схемы пользовательских путей для каждого модуля. Схемы реализованы в виде диаграмм User Flow - блок-схема пользовательских маршрутов, который зачастую охватывает либо продукт в целом, либо отдельную его функцию. Несмотря на то, что данный подход используется скорее в UX-проектировании, он позволяет понять логику работы приложения и наглядно определить весь необходимый для реализации функционал. Рассмотрим подробнее схему работы модуля “Аналитика и мониторинг”. Визуализация данной диаграммы представлена на рисунке 22.

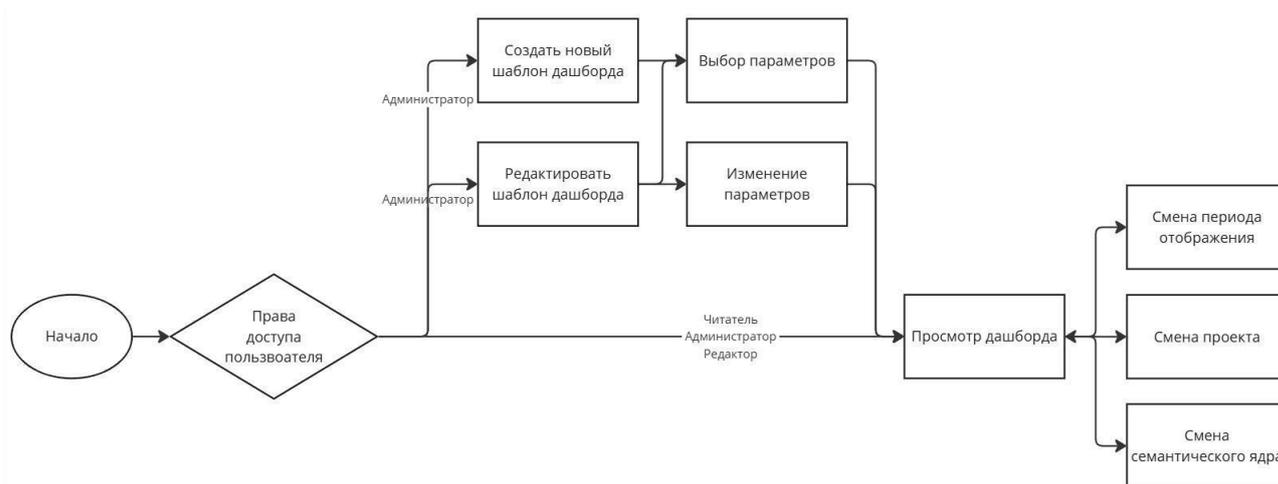


Рисунок 22 - User Flow для модуля мониторинга и аналитики

Рассмотрим схему подробнее. У пользователя есть три ключевых действия: создание нового шаблона дашборда, изменение текущих шаблонов дашбордов или их просмотр. При этом, правами на взаимодействие с шаблонами обладает только администратор, все остальные пользователи могут только просматривать дашборды. Интерактивные элементы взаимодействия для дашборда включают в себя:

- выбор периода, за который отображаются графики;
- выбор детализации графика (дни, недели, месяцы);
- выбор проекта (сайт, к которому относятся семантические ядра);
- выбор семантического ядра в рамках выбранного проекта.

Логика работы в таком случае является следующей: по умолчанию отображается наиболее популярный дашборд, далее любой пользователь может выбрать требуемый проект и семантическое ядро, привязанное к данному проекту, при этом если пользователь имеет статус “Администратор”, он может редактировать шаблоны дашбордов, выбирая или

прописывая новые запросы. Данный подход позволит сделать дашборды универсальными и доступными для всех, при этом гарантирует корректность отображаемых данных и графиков.

Таким образом, данный модуль обеспечит быструю и легкую работу с данными, связанными с SEO-оптимизацией сайта или конкретного раздела сайта [19].

4.2 Разработка User Flow для модуля “Хранение семантических ядер”

Следующим логическим модулем с точки зрения пользовательского пути является блок хранения семантических ядер. Данный модуль предназначен для агрегации и структуризации семантических ядер в компании, поскольку они являются важным ресурсом, который на данный момент часто может быть утерян или не может использоваться повторно из-за ненадежности локальных хранилищ у сотрудников.

Структура семантического ядра включает множество элементов, которые обеспечивают его функциональность и полезность для SEO-оптимизации. Семантическое ядро представляет собой упорядоченный список ключевых слов и фраз, отражающих интересы целевой аудитории и соответствующих целям конкретного веб-ресурса. Оно строится на основе анализа пользовательских запросов и состоит из нескольких ключевых компонентов.

Первым элементом являются ключевые слова, которые составляют основу семантического ядра. Они могут быть высокочастотными, среднечастотными и низкочастотными, в зависимости от частоты их использования пользователями в поисковых системах. Высокочастотные запросы привлекают широкий трафик, но имеют высокую конкуренцию, тогда как низкочастотные запросы более специфичны и часто обеспечивают более целевой трафик. Ключевые фразы важны для SEO, потому что они помогают поисковым системам понять, о чём конкретно ваш сайт или страница, а также какие запросы пользователей могут привести их к вашему контенту. Использование правильных ключевых фраз повышает вероятность того, что ваш сайт будет отображаться в поисковых результатах по релевантным запросам.

Следующим важным элементом является группировка ключевых слов по смыслу, целям или тематике. Это позволяет структурировать семантическое ядро и упростить дальнейшую работу с ним. Группы могут быть построены на основе категорий товаров и услуг, разделов сайта или этапов воронки продаж. Каждая группа объединяет ключевые слова, связанные с одной темой или задачей.

Для понимания логики работы данного модуля была составлена схема в виде User Flow данного модуля. Её визуализация представлена на рисунке 23.

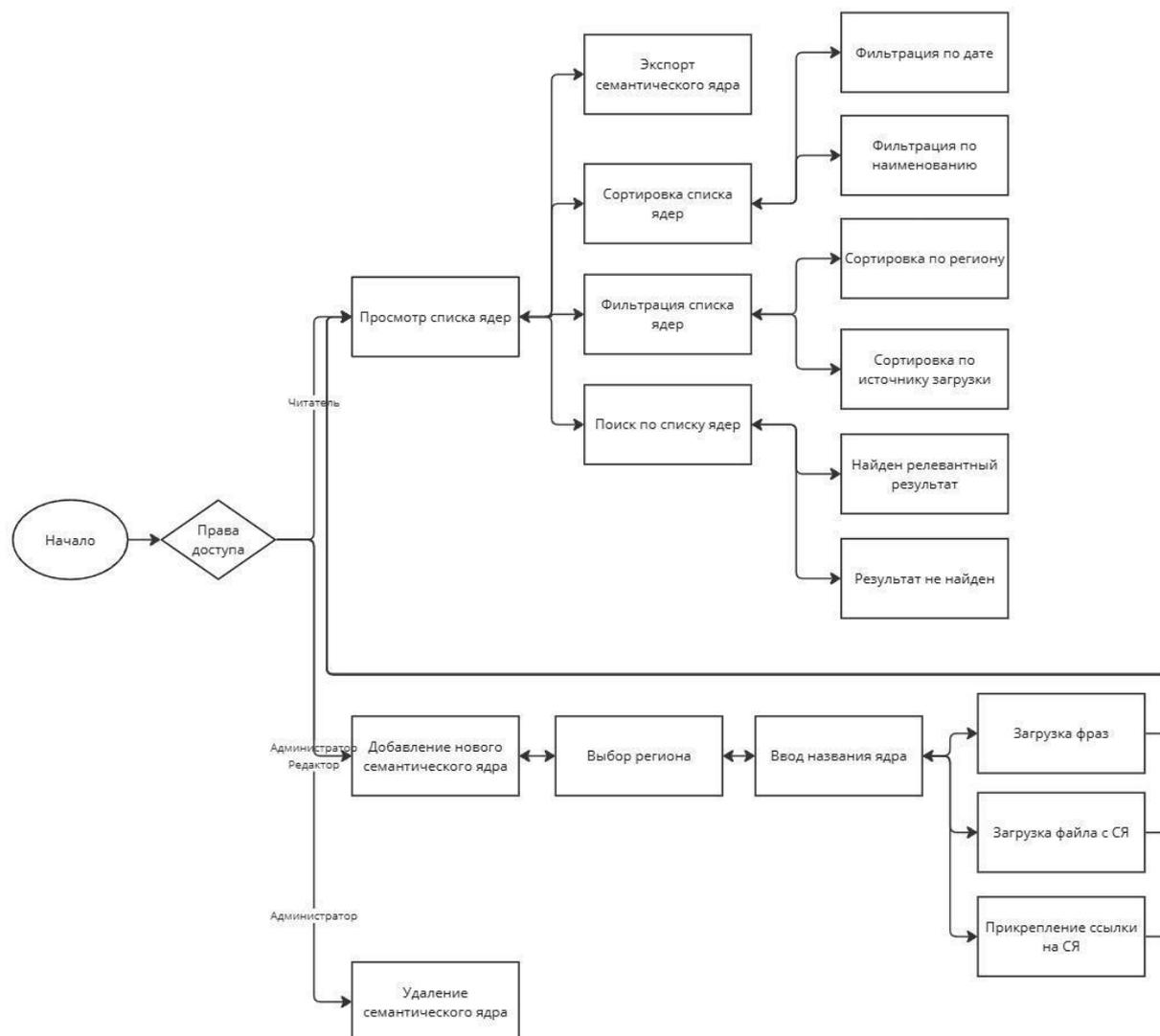


Рисунок 23 - User Flow для модуля хранения семантических ядер

Рассмотрим данную схему подробнее. В данном случае есть два ключевых действия: просмотр списка семантических ядер и добавление нового семантического ядра. Обратим внимание, что добавление доступно только пользователям с ролями “Администратор” и “Редактор”, а для роли “Администратор” также есть возможность дополнительного действия - удаления семантических ядер из библиотеки. Рассмотрим каждое действие подробнее:

- логика работы добавления нового семантического ядра заключается в последовательном вводе информации о семантическом ядре с его последующей загрузкой в базу данных. Для этого в приложении реализована отдельная кнопка и модальное окно. Важно отметить, что есть три варианта добавления фраз в семантическое ядро: ручной ввод списка фраз, загрузка файла с фразами и прикрепление ссылки на таблицу;

- логика работы просмотра семантических ядер включает в себя ряд возможностей для взаимодействия со списком: фильтрация, сортировка и поиск по семантическому ядру. Также отметим, что в качестве результатов поиска или выбора некоторых сочетаний фильтров

может приводить к получению результата “Семантическое ядро не найдено”, что подразумевает отсутствие ядра, которое бы подходило под введенные параметры;

— удаление семантического ядра приведет к полному удалению данных о нем: в том числе и из списка доступных дашбордов с данным семантическим ядром.

4.3 Разработка схемы архитектуры сервиса

Для завершения проектирования и составления полного представления о работе будущей системы, была также составлена схема архитектуры серверной части программы. Схема архитектуры сервиса при разработке необходима для того, чтобы наглядно отобразить структуру и взаимосвязи между компонентами системы, а также определить, как они будут взаимодействовать друг с другом. Она помогает разработчикам, архитекторам и другим участникам проекта понять, как будет строиться приложение, какие технологии и подходы будут использованы, и как компоненты будут распределены по инфраструктуре. Визуализация данной схемы представлена на рисунке 24.

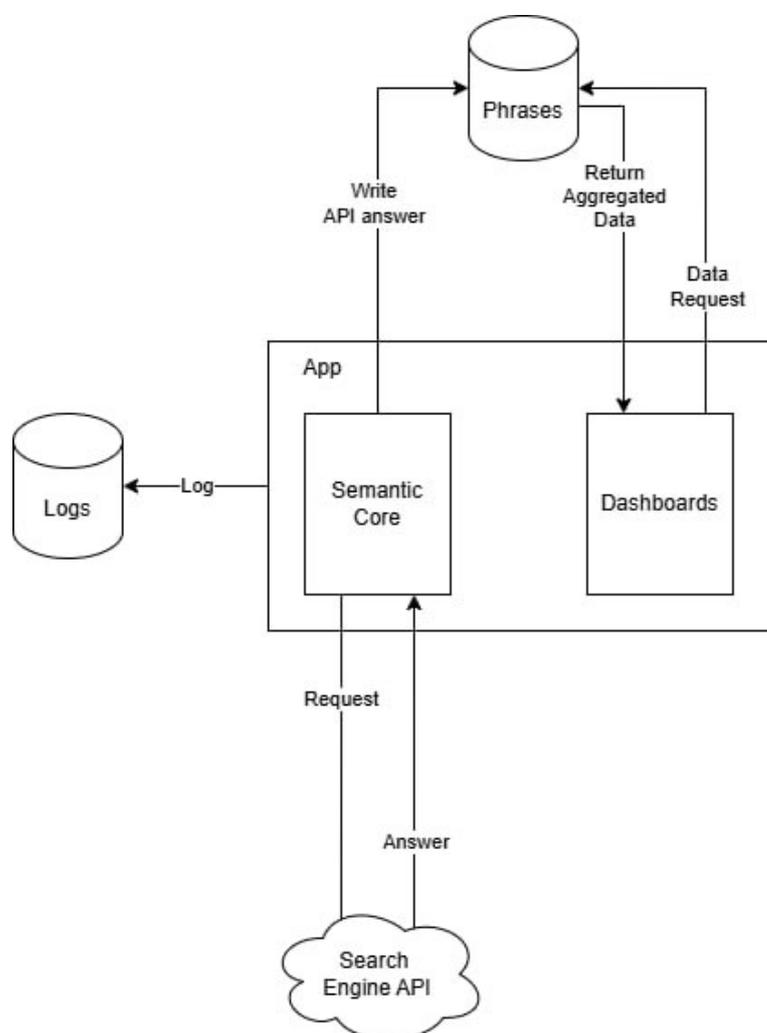


Рисунок 24 - Схема архитектуры сервиса

Рассмотрим данную схему подробнее. Общая система содержит в себе два алгоритмических модуля: модуль для хранения и работы с семантическими ядрами и аналитический модуль для работы с дашбордами и аналитикой [20]. Поскольку уже неоднократно была отмечена важность логирования данных в качестве требований к системе, весь сервис взаимодействует с отдельным хранилищем логов работы сервиса. Далее, модуль семантического ядра имеет два логических пути:

— взаимодействие с внешним интерфейсом API для парсинга, сбора и обработки данных, касающейся семантических ядер. API определяет набор правил и протоколов, с помощью которых одна программа может запрашивать данные или выполнять действия в другой программе или сервисе. Это позволяет интегрировать внешние сервисы, обмениваться данными и расширять функциональность без необходимости переписывать код с нуля. API может быть использован для работы с веб-сервисами, базами данных, операционными системами и другими программными компонентами, делая взаимодействие между ними более гибким и стандартизированным. Модуль отправляет запросы к API и получает ответ в виде обработанных или расширенных данных;

— взаимодействие с основной базой данных, в которой хранится вся информация о семантических ядрах.

Описанная база данных является связующим звеном и используется как в качестве хранилища данных для модуля семантических ядер, а также и источником данных для аналитического модуля.

Аналитический модуль при этом может обращаться в ключевую базу данных для получения необходимых данных и их обработки, но также может и записывать или перезаписывать данных в базе по результатам обработки или анализа фраз.

Данный подход к формированию архитектуры сервиса позволит обеспечить надежность, целостность и полноту данных в любой момент, а также обеспечит разделение функционала по различным модулям. Использование подходов параллельного выполнения задач, такое решение предоставит возможности для быстрого отклика и грамотного распределения ресурса в условиях многопоточковой работы большого количества пользователей в системе одновременно.

Заключение

В результате проделанной работы была спроектирована система поддержки принятия решений для задач SEO. SEO-оптимизация представляет собой сложный процесс, требующий анализа большого объема данных, мониторинга изменений и выработки стратегий на основе постоянно изменяющихся условий. Спроектированная система направлена на автоматизацию ключевых этапов этого процесса, что позволяет существенно повысить эффективность работы SEO-специалистов.

Прежде всего были четко сформулированы требования к обозначенной системе, в зависимости от которого и строилось дальнейшее проектирование. Затем были разработаны дизайн-макеты экранов, в соответствии с которыми будет вестись разработка клиентской части системы. Данные макеты прорабатывают большинство возможных ситуаций, однако в процессе разработки могут понадобиться дополнительные изменения и разработки всплывающих и модальных окон. Особое значение было уделено визуализации данных, которая позволяет не только анализировать текущие метрики, но и оперативно выявлять проблемы и тренды. Дашборды, разработанные в рамках проектирования системы, обеспечивают удобный доступ к ключевым показателям, что значительно ускоряет процесс принятия решений.

Также были проведены мероприятия по подготовке к разработке серверной части системы: разработаны схемы баз данных для хранения как семантических ядер, так и аналитической информации, диаграммы пользовательских путей для определения логики работы сервиса и схема архитектуры сервиса в целом для понимания структуры предстоящих работ.

Таким образом, разработанная система поддержки принятия решений способствует повышению конкурентоспособности веб-ресурсов за счет более точного и оперативного подхода к управлению их видимостью в поисковых системах.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

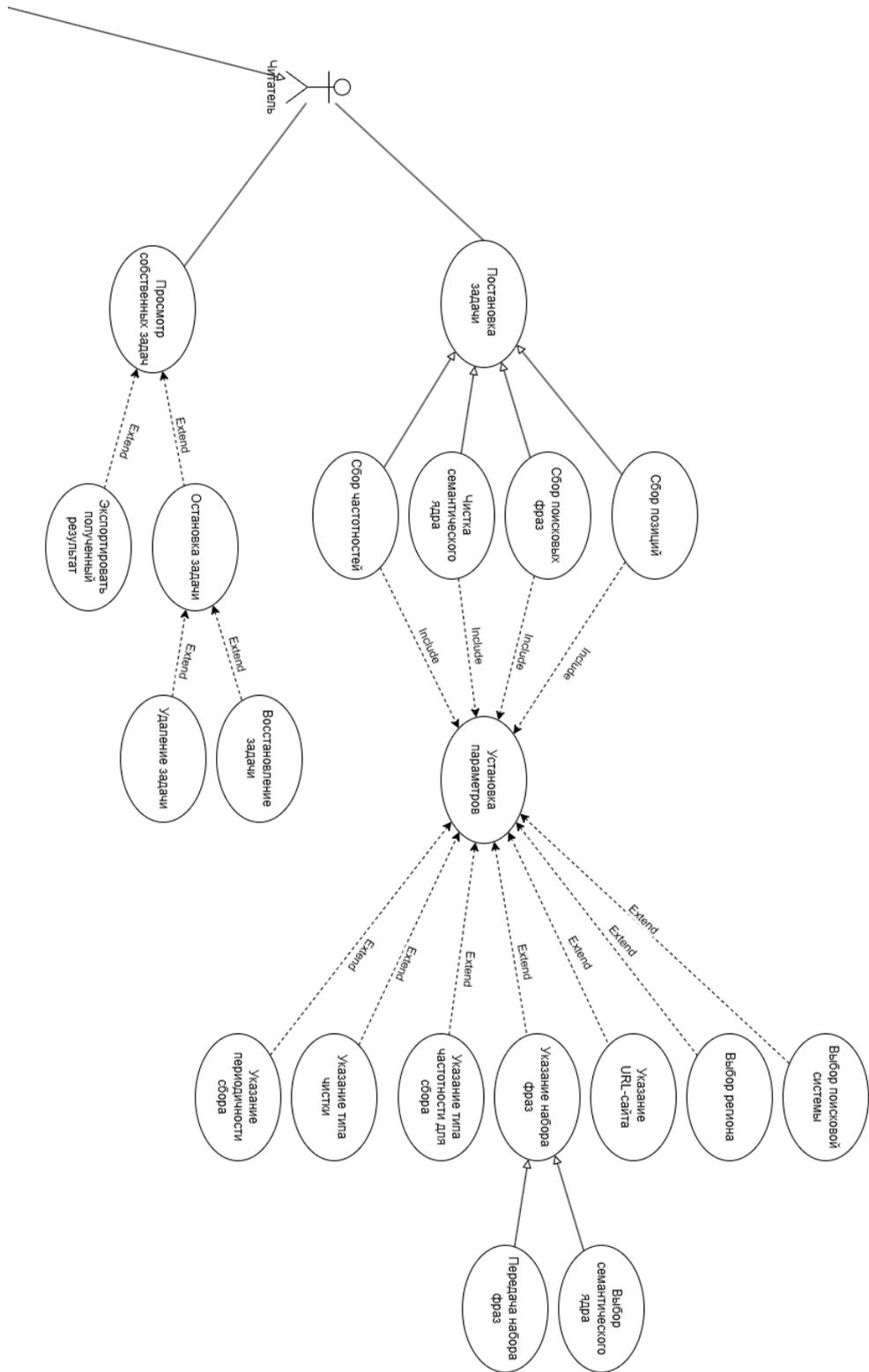
1. Системы поддержки принятия решений и бизнес-аналитика / Турбан Э., Шарда Р., Деллен Г. – Москва: Вильямс, 2021. – 504 с.
2. Данные и решения: Как построить умную компанию в цифровую эпоху / Хаммер М. – Москва: Альпина Паблишер, 2021. – 320 с.
3. SEO 2023: Комплексный подход к продвижению сайтов / Бондаренко А. – Москва: Манн, Иванов и Фербер, 2023. – 280 с.
4. Дизайн привычных вещей / Норман Д. – Москва: Манн, Иванов и Фербер, 2021. – 400 с.
5. UX-дизайн: руководство для аналитиков и разработчиков / Крюгер Дж. – Москва: Альпина Паблишер, 2022. – 360 с.
6. The Art of SEO [Электронный ресурс] / Enge E., Spencer S. // O'Reilly Media – Режим доступа: <https://www.oreilly.com/library/view/the-art-of/> (дата обращения: 29.12.2024).
7. Анализ данных в больших системах / Матвейчев А. – Москва: БХВ-Петербург, 2022. – 456 с.
8. SQL for Data Scientists / Petkovic D. – Hoboken: Wiley, 2022. – 320 с.
9. Практическое руководство по базам данных: от проектирования до оптимизации / Розенберг Д. – Москва: Альпина Диджитал, 2023. – 360 с.
10. Google Cloud BigQuery Documentation [Электронный ресурс] // Google – Режим доступа: <https://cloud.google.com/bigquery/docs> (дата обращения: 06.01.2025).
11. Google Analytics API Documentation [Электронный ресурс] // Google – Режим доступа: <https://developers.google.com/analytics> (дата обращения: 02.01.2025).
12. PostgreSQL Documentation [Электронный ресурс] // PostgreSQL – Режим доступа: <https://www.postgresql.org/docs/> (дата обращения: 29.12.2024).
13. Greenplum Documentation [Электронный ресурс] // Greenplum – Режим доступа: <https://greenplum.org/documentation/> (дата обращения: 06.01.2025).
14. ClickHouse Documentation [Электронный ресурс] // ClickHouse – Режим доступа: <https://clickhouse.com/docs/> (дата обращения: 30.12.2024).
15. Яндекс Метрика API Документация [Электронный ресурс] // Яндекс – Режим доступа: <https://yandex.ru/dev/metrika/> (дата обращения: 07.01.2025).
16. Cloud Architecture Patterns [Электронный ресурс] / Beal V. // O'Reilly Media – Режим доступа: <https://www.oreilly.com/library/view/cloud-architecture/> (дата обращения: 05.01.2025).
17. Designing Data-Intensive Applications / Kleppmann M. – Sebastopol: O'Reilly Media, 2022. – 540 с.
18. Архитектура баз данных для аналитических задач / Дейта К., Чайка С. – Москва:

Питер, 2022. – 348 с.

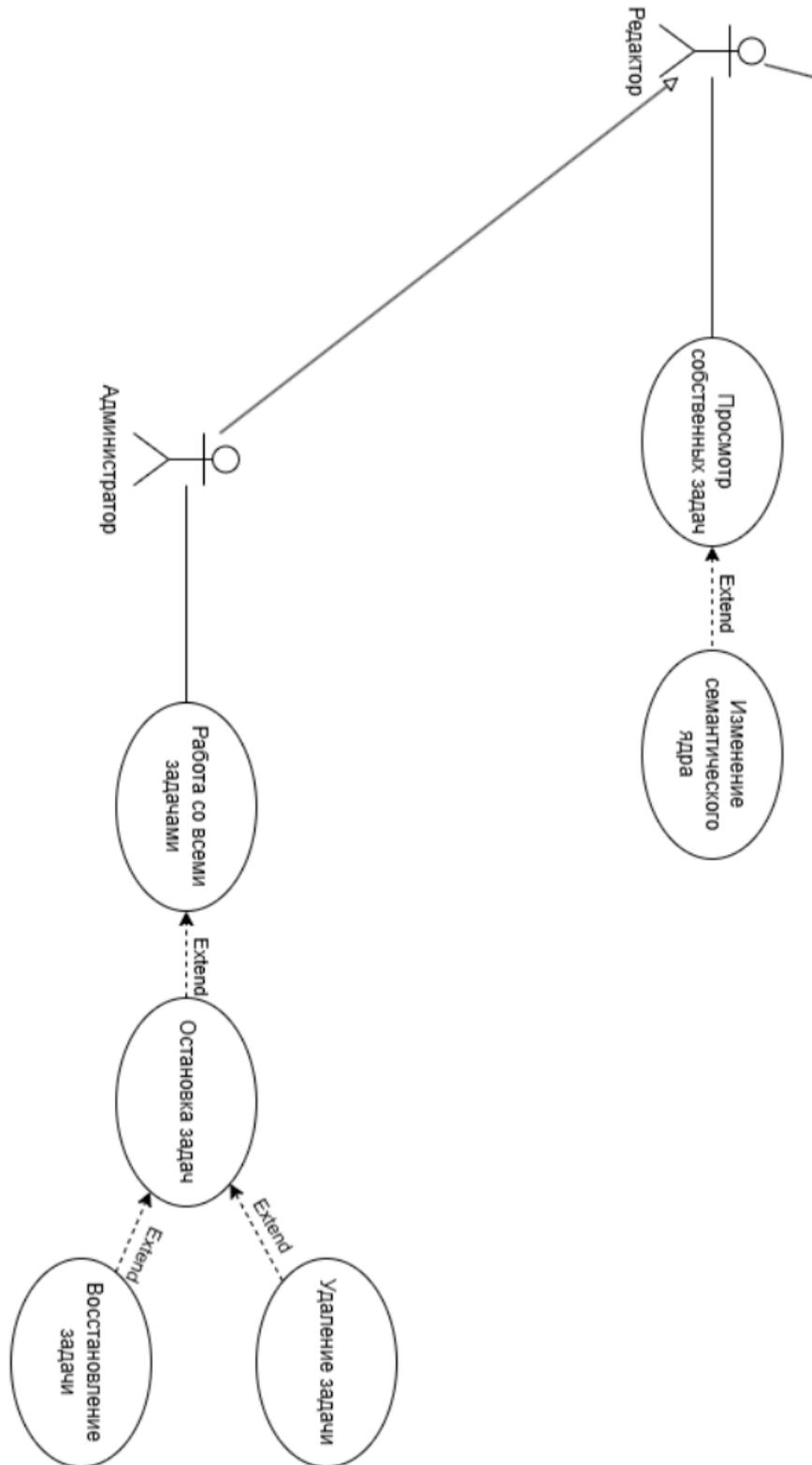
19. Программирование для анализа данных: от сбора до визуализации / О'Нил К., Шах Д. – Москва: ДМК Пресс, 2022. – 320 с.

20. Data Science for Business [Электронный ресурс] / Mongeon P. // O'Reilly Media – Режим доступа: <https://www.oreilly.com/library/view/data-science/> (дата обращения: 03.01.2024).

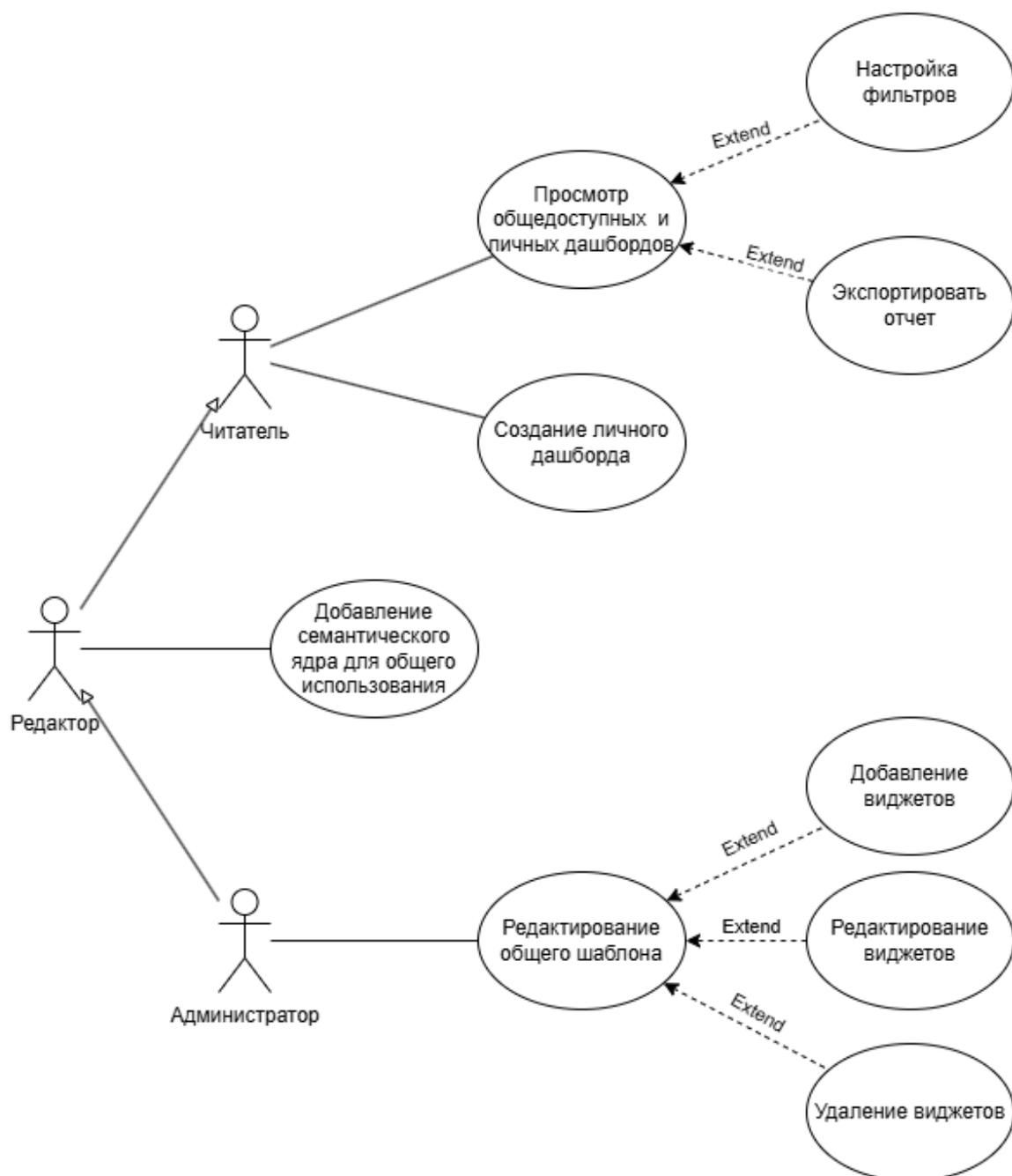
Приложение А



Приложение Б



Приложение В



Приложение Г

