



**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ «ПЛАТФОРМА ВИЗУАЛИЗАЦИИ
МЕДИЦИНСКИХ ДАННЫХ»**

Содержание

[ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ ХАРАКТЕРИСТИК ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
«ПЛАТФОРМА ВИЗУАЛИЗАЦИИ МЕДИЦИНСКИХ ДАННЫХ»](#)

[ОПИСАНИЕ ПРОЦЕССОВ, ОБЕСПЕЧИВАЮЩИХ ПОДДЕРЖАНИЕ ЖИЗНЕННОГО ЦИКЛА
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ «ПЛАТФОРМА ВИЗУАЛИЗАЦИИ МЕДИЦИНСКИХ
ДАННЫХ», В ТОМ ЧИСЛЕ УСТРАНЕНИЕ НЕИСПРАВНОСТЕЙ И СОВЕРШЕНСТВОВАНИЕ, А
ТАКЖЕ ИНФОРМАЦИЮ О ПЕРСОНАЛЕ, НЕОБХОДИМОМ ДЛЯ ОБЕСПЕЧЕНИЯ ТАК](#)

[ОПИСАНИЕ ТЕХНИЧЕСКОЙ АРХИТЕКТУРЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
«ПЛАТФОРМА ВИЗУАЛИЗАЦИИ МЕДИЦИНСКИХ ДАННЫХ»](#)

[ИНСТРУКЦИЯ ПО РАЗВЕРТЫВАНИЮ ЭКЗЕМПЛЯРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
«ПЛАТФОРМА ВИЗУАЛИЗАЦИИ МЕДИЦИНСКИХ ДАННЫХ»](#)

[ИНСТРУКЦИЯ ПО ИСПОЛЬЗОВАНИЮ СИСТЕМЫ](#)

[ЦЕНОВАЯ ПОЛИТИКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ «ПЛАТФОРМА ВИЗУАЛИЗАЦИИ
МЕДИЦИНСКИХ ДАННЫХ»](#)

**ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ ХАРАКТЕРИСТИК
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ «ПЛАТФОРМА ВИЗУАЛИЗАЦИИ
МЕДИЦИНСКИХ ДАННЫХ»**

Перечень сокращений, терминов и определений

Сокращение	Определение
МИС	Медицинские информационные системы
ПО	Программное обеспечение
СУБД	Система управления базами данных

Программное обеспечение «Платформа визуализации медицинских данных» (далее – ПО «Платформа визуализации медицинских данных») – программное обеспечение, предназначенное для организации централизованного сбора и предоставления аналитической информации по различным направлениям деятельности организации.

ПО «Платформа визуализации медицинских данных» функционирует как отдельная система, так и в составе программных комплексов.

Основными процессами, для автоматизации которых предназначено ПО «Платформа визуализации медицинских данных», является:

- процесс сбора, консолидации и обработки оперативных и исторических данных из различных источников;
- процесс централизованного хранения информации и предоставления доступа к консолидированным данным;
- процесс подготовки аналитической и регламентированной отчетности на основе консолидированных данных.

В рамках процесса сбора, консолидации и обработки информации ПО «Платформа визуализации медицинских данных» выполняет следующие функции:

- преобразование данных в единый формат, оптимальный с точки зрения анализа;
- обработка и загрузка данных в аналитическое хранилище.

В рамках процесса централизованного хранения информации и предоставления доступа к ней ПО «Платформа визуализации медицинских данных» выполняет следующие функции:

- хранение данных;
- обновление данных по регламенту;
- обеспечение целостности структуры данных;
- оперативное предоставление информации;
- разграничение прав доступа;

В рамках процесса подготовки отчетности на основе консолидированных данных ПО «Платформа визуализации медицинских данных» выполняет следующие функции:

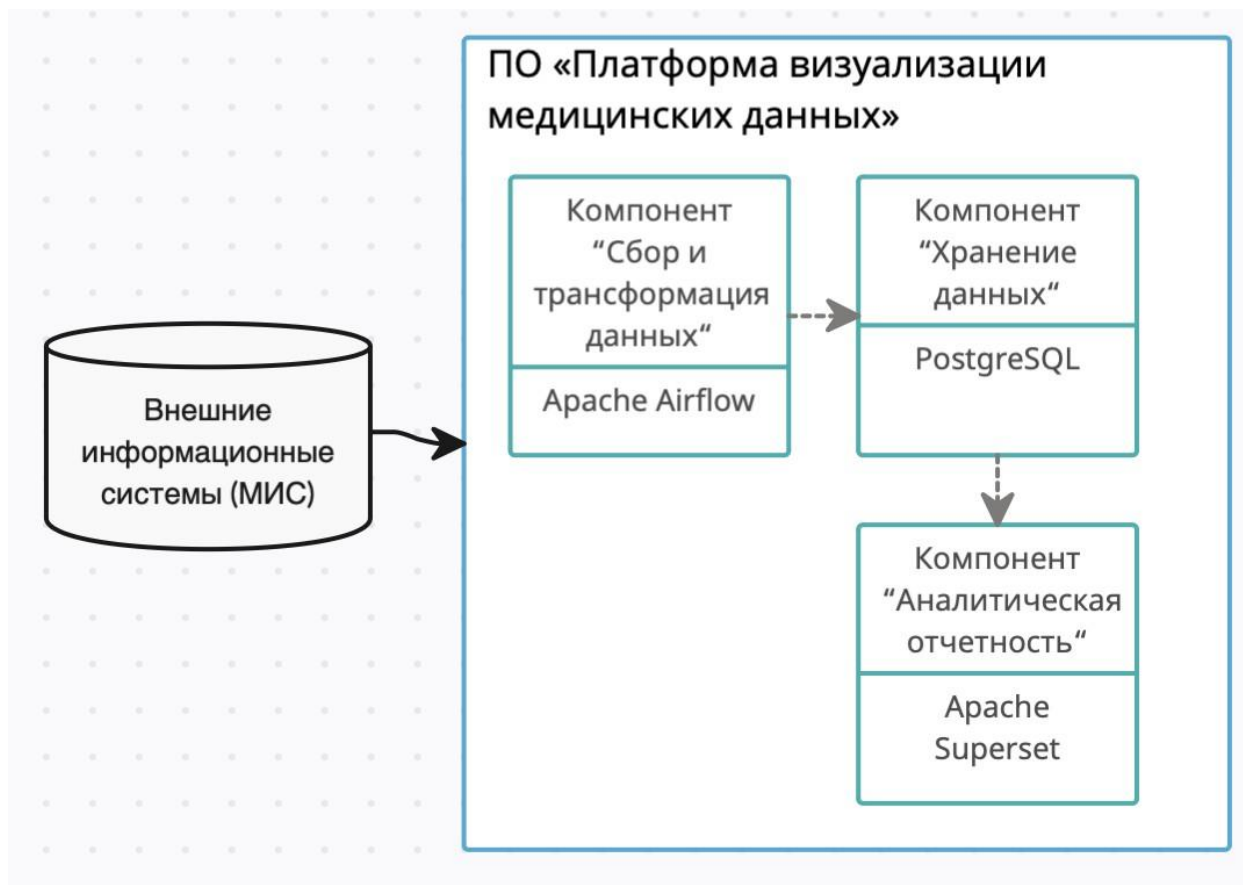
- предоставление данных для многомерного анализа;
- формирование аналитических отчетов;
- выгрузка сформированных отчетов в определенных требованиями форматах.

1.1. Архитектура ПО «Платформа визуализации медицинских данных»

В состав ПО «Платформа визуализации медицинских данных» входят следующие компоненты:

- компонент «Сбор и трансформация данных»;
- компонент «Хранение данных»;
- компонент «Аналитическая отчетность».

Концептуальная схема построения ПО «Платформа визуализации медицинских данных» представлена на Рисунке 1.



Н

иже описаны функциональные характеристики компонентов.

2. Компонент «Сбор и трансформация данных»

Компонент «Сбор и трансформация данных» предназначен для выполнения процессов сбора, проверки, очистки, трансформации и доставки данных от систем-источников в область постоянного хранения данных. Загрузка данных осуществляется в реляционную СУБД PostgreSQL, где хранится копия исходных данных на постоянной основе.

Доступ к исходным базам данных осуществляется с использованием Apache Airflow.

Загрузка данных из систем-источников происходит в следующих режимах:

- начальная загрузка данных - первичное наполнение хранилища данными;

- регламентная загрузка данных - производится ежедневно в ночное время;
- загрузка данных по требованию – принудительная загрузка данных за определенные периоды по требованию пользователей.

Информация о статусе проведения и параметрах загрузки в разрезе исходных баз данных (исходных таблиц баз данных) представлена в интерфейсе Apache Airflow. Производится запись следующей информации:

- время начала загрузки;
- количество загруженных записей;
- время окончания загрузки.

Загрузка может осуществляться отдельно по каждому источнику данных с возможностью выбора определённой таблицы.

В компоненте проводятся следующие шаги преобразования данных:

- преобразование исходной структуры данных. Производится первичная очистка данных (удаление пробелов, лишних символов);
- преобразование структуры данных. Производится дальнейшая очистка данных (удаляются дубликаты, преобразование данных и т. д.);

Компонент «Сбор и трансформация данных» предоставляет следующие функциональные возможности:

- хранение полной копии данных из систем источников;
- обеспечение протоколирования событий, происходящих в процессе сбора и передачи данных;
- обеспечение возможности получения данных от дополнительных систем-источников;
- обеспечение обработки нештатных ситуаций.
- контроль целостности данных;
- унификация данных;
- обогащение данных;
- поддержка историчности справочников;
- приведение данных к структурам, оптимальным для выполнения аналитических запросов;
- определение и фиксация ошибочных данных;
- ведение журналов обработки и загрузки данных;
- извещение пользователей о нештатных ситуациях в процессе работы компонента;
- архивация и очистка журналов.

3. Компонент «Хранение данных»

Компонент «Хранение данных» предназначен для хранения витрин данных и предоставления информации для построения аналитических отчетов системой визуализации.

Компонент «Хранение данных» предоставляет следующие функциональные возможности:

- хранение агрегированных данных;
- хранение значений измерений и аналитических показателей;
- обновление данных в компоненте хранения данных по требованию и по графику;
- хранение значений справочников, словарей и метаданных, необходимых для функционирования ПО «Платформа визуализации медицинских данных»;
- обеспечение доступа к хранимым данным;
- контроль качества данных в компоненте «Хранение данных»;
- протоколирование действий по загрузке, хранению и обработке данных в компоненте «Хранение данных»;
- обеспечение сохранности данных в случае нештатной ситуации и их восстановление в случае необходимости.

4. Компонент «Аналитическая отчетность»

Компонент «Аналитическая отчетность» предназначен для анализа данных и формирования отчетов.

Взаимодействие пользователя с компонентами аналитической отчетности реализовано с помощью функциональности Apache Superset:

- взаимодействие происходит в web-среде;
- отчеты формируются пользователем в конструкторе отчетов;
- информация визуализируется в виде Отчетов и Информационных панелей.

Компонент предоставляет следующие функциональные возможности:

- обеспечение возможности получения данных из хранилища данных и формирование на их базе необходимых пользователям аналитических отчетов;
- автоматизированное создание произвольных форм отчетности средствами встроенного конструктора формирования отчетов (конструктора) самостоятельно пользователями без необходимости использования специализированных языков и алгоритмов;
- автоматизированное создание следующих видов представлений детализированных данных: график, круговая диаграмма, столбчатая диаграмма, таблица среза, линейная таблица;
- добавление новых показателей в конструктор аналитических отчетов из хранилища

данных;

- построение интерактивных отчетов;

- формирование параметризованных отчетов на основе значений параметров, переданных пользователем;

- детализация данных в отчетах, переход от агрегированных данных к детализированным показателям;

- возможность параметризованного перехода с одного отчета на другой или с отчета на информационную панель;

- возможность сохранения на одной информационной панели различных видов представлений;

- настройка панелей в зависимости от ролей пользователя в ПО «Платформа визуализации медицинских данных»;

- сохранение отчетов;

- экспорт отчетов;

- настройка пользователей, их ролей и прав доступа;

- логирование изменений витрин данных.

**ОПИСАНИЕ ПРОЦЕССОВ, ОБЕСПЕЧИВАЮЩИХ ПОДДЕРЖАНИЕ ЖИЗНЕННОГО
ЦИКЛА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ «ПЛАТФОРМА ВИЗУАЛИЗАЦИИ
МЕДИЦИНСКИХ ДАННЫХ», В ТОМ ЧИСЛЕ УСТРАНЕНИЕ НЕИСПРАВНОСТЕЙ И
СОВЕРШЕНСТВОВАНИЕ, А ТАКЖЕ ИНФОРМАЦИЮ О ПЕРСОНАЛЕ, НЕОБХОДИМОМ
ДЛЯ ОБЕСПЕЧЕНИЯ ТАКОЙ ПОДДЕРЖКИ**

Содержание

Перечень сокращений, терминов и определений	11
1. Жизненный цикл программного обеспечения «Платформа визуализации медицинских данных»	12
1.1. Участники жизненного цикла.....	12
1.2. Описание жизненного цикла.....	13
2. Организация производственного процесса	13
2.1. Планирование версии и управление изменениями	14
2.2. Реализация.....	15
2.3. Тестирование.....	15
2.4. Выпуск (релиз) версии Системы.....	16
2.4.1. Подготовка Системы (версии) к промышленной эксплуатации	16
2.4.2. Запуск Системы в промышленную эксплуатацию	16
2.5. Сопровождение Системы.....	16
2.5.1. Порядок работы при возникновении инцидентов.....	17
2.5.1.1. Участники и ответственность.....	17
2.5.1.2. План реагирования.....	17
2.5.1.3. Численность, функции и квалификация персонала.....	18
2.5.1.3.1. Требования и численность персонала (пользователей).....	18
2.5.1.3.2. Требования к квалификации и показу взаимодействия персонала	19
2.5.1.3.3. Требуемый режим работы персонала.....	19
3. Фактический адрес размещения инфраструктуры разработки Системы.....	20
4. Фактический адрес размещения разработчиков Систем.....	20
5. Фактический адрес размещения службы поддержки Системы.....	20

Перечень сокращений, терминов и определений

Сокращение	Определение
ЖЦ	Жизненный цикла
Ответственный сотрудник	Сотрудник, ответственный за реализацию и сопровождение ПО «Платформа визуализации медицинских данных»
ПДн	Персональные данные
ПО	Программное обеспечение
СТП	Служба технической поддержки
ТЗ	Техническое задание

2. Жизненный цикл программного обеспечения «Платформа визуализации медицинских данных»

2.1. Участники жизненного цикла

Участниками жизненного цикла ПО «Платформа визуализации медицинских данных» являются:

– Сотрудник, ответственный за реализацию и сопровождение ПО «Платформа визуализации медицинских данных»:

- управляет изменениями, формирует содержание ПО «Платформа визуализации медицинских данных»/версии ПО «Платформа визуализации медицинских данных»;

- планирует бюджет проекта, оценивает риски и разрабатывает план их предотвращения;

- участвует и определяет техническое решение по реализации ПО «Платформа визуализации медицинских данных»/версии ПО «Платформа визуализации медицинских данных»;

- описывает функциональные требования и участвует в разработке технического задания;

- взаимодействует с партнёрами и заказчиками;

- взаимодействует со службой технической поддержки и службой контроля качества, а также с отделом администрирования в рамках реализации и сопровождения ПО «Платформа визуализации медицинских данных»;

– Служба технической поддержки:

- обеспечивает контроль и сопровождение работоспособности ПО «Платформа визуализации медицинских данных» в процессе опытной и промышленной эксплуатации;

- осуществляет сопровождение и консультирование пользователей ПО «Платформа визуализации медицинских данных»;

– Служба контроля качества:

- проверяет качество реализации и соответствие ПО «Платформа визуализации медицинских данных» заявленным требованиям;

- осуществляет тестирование ПО «Платформа визуализации медицинских данных»/версии ПО «Платформа визуализации медицинских данных»;

- определяет готовность версии ПО «Платформа визуализации медицинских данных» к промышленной эксплуатации;

– Отдел разработки:

- осуществляет техническую реализацию функциональных требований ПО «Платформа визуализации медицинских данных»;
- Отдел администрирования:
 - обеспечивает обновление ПО «Платформа визуализации медицинских данных» и программного обеспечения;
 - осуществляет настройку и актуализацию серверов и ПО «Платформа визуализации медицинских данных».
 - производит настройку мониторинга и реагирования на инциденты.

2.2. Описание жизненного цикла

Развитие ПО «Платформа визуализации медицинских данных» происходит в рамках проектной деятельности и выполняется с использованием гибкой методологии управления проектами на базе методологии **Scrum**. ЖЦ ПО «Платформа визуализации медицинских данных» состоит из следующих этапов:

1. Планирование – формирование содержания следующей версии.
2. Реализация – техническая реализация функциональных возможностей ПО «Платформа визуализации медицинских данных», согласно требованиям и ТЗ.
3. Тестирование – проверка технической реализации на соответствие требованиям ТЗ, проверка качества реализации.
4. Выпуск (релиз) ПО «Платформа визуализации медицинских данных» – подготовка пакета обновления (установки), выполнение обновления ПО «Платформа визуализации медицинских данных» в промышленной среде.
5. Сопровождение ПО «Платформа визуализации медицинских данных» – процесс поддержания функционирования ПО «Платформа визуализации медицинских данных» в промышленной среде, помощь пользователям ПО «Платформа визуализации медицинских данных», мониторинг работоспособности и управление инцидентами.

2. Организация производственного процесса

Разработка, тестирование и промышленная эксплуатация ПО «Платформа визуализации медицинских данных» должны происходить в средах, отделенных друг от друга:

- **Среда разработки**– стенд, включающий компоненты, на которых происходит разработка и отладка ПО «Платформа визуализации медицинских данных». К данной среде имеют доступ все Программисты.

–Тестовая среда – стенд, включающий компоненты, на которых происходит тестирования ПО «Платформа визуализации медицинских данных» перед передачей ее в промышленную эксплуатацию. К данной среде имеют доступ специалисты СКК.

–Промышленная среда – включает компоненты, на которых происходит промышленная эксплуатация ПО «Платформа визуализации медицинских данных». К данной среде имеют доступа Администраторы ПО «Платформа визуализации медицинских данных» и специалисты СТП.

Обязанности и ответственность по разработке, тестированию и промышленной эксплуатации разделены и возложены на разных сотрудников компании:

– за разработку ПО «Платформа визуализации медицинских данных» отвечает Отдел разработки;

– за тестирование – СКК;

– за промышленную эксплуатацию – СТ.

2.1. Планирование версии и управление изменениями

Планированием версии ПО «Платформа визуализации медицинских данных» и управлением изменениями занимается сотрудник, ответственный за реализацию и сопровождение ПО «Платформа визуализации медицинских данных».

Ответственный сотрудник осуществляет планирование версии на основе следующих входных данных:

– формирование новых требований к функциональности ПО «Платформа визуализации медицинских данных», регламентируемых потребностями пользователей, а также изменениями в законодательстве;

– неисправности или функциональные несоответствия, выявленные при эксплуатации ПО «Платформа визуализации медицинских данных».

Ответственный сотрудник формирует план развития ПО «Платформа визуализации медицинских данных», включающий следующие разделы:

– перечень задач, их приоритет;

– сроки реализации;

– плановые сроки выпуска ПО «Платформа визуализации медицинских данных» (релизы).

Ответственный сотрудник следит за исполнением плана развития ПО «Платформа визуализации медицинских данных», корректирует его в случае появления срочных задач, а также управляет и контролирует выпуск новых версий ПО «Платформа визуализации медицинских данных».

Все выпуски версий ПО «Платформа визуализации медицинских данных» делятся на следующие виды:

- Плановые выпуски (релизы);
- Оперативное исправление;
- Срочное исправление.

Плановые выпуски (релизы) выпускаются согласно общему плану развития ПО «Платформа визуализации медицинских данных», обязательно включают в себя изменения функциональности, могут также включать в себя исправления ошибок.

Оперативное исправление – это набор исправлений, включающий только исправления ошибок, обнаруженных в ходе тестирования планового выпуска ПО «Платформа визуализации медицинских данных». В оперативное исправление включаются ошибки, имеющие блокирующий и критичный приоритет, т. е. ошибки, непосредственно влияющие на основной функционал ПО «Платформа визуализации медицинских данных».

Срочное исправление – это набор исправлений, включающий только исправление ошибок, обнаруженных при промышленной эксплуатации ПО «Платформа визуализации медицинских данных» и требующих безотлагательного исправления.

2.2. Реализация

На основании планов и ТЗ, разработанных Ответственным сотрудником, происходит разделение задач, определяется трудоемкость и исполнители, формируются краткосрочные планы.

Ответственный сотрудник определяет содержание краткосрочных планов, сроки реализации, выпуск версии в промышленную эксплуатацию.

Ответственный сотрудник контролирует выполнение плана реализации, актуализирует техническую документацию и инструкции СКК и СТП в ходе реализации.

По окончании реализации версии Ответственный сотрудник контролирует актуальность материалов по ПО «Платформа визуализации медицинских данных»: планов, технической документации, описания ПО «Платформа визуализации медицинских данных», пользовательских инструкций; - публикует актуальную информацию, делает ее доступной для пользователей, СКК и СТП.

2.3. Тестирование

Тестирование осуществляется СКК.

На основании ТЗ и задач на реализацию:

- формируется тест-план;
- распределяются задачи по тестированию между сотрудниками СКК;
- подготавливаются/изменяются тестовые сценарии.

При тестировании строго запрещено использовать производственные данные.

К таким данным относятся: пароли, ключи шифрования, персональные данные (далее – ПДн) и прочая конфиденциальная информация.

Передача планового релиза в тестирование производится согласно графику выхода версий.

По окончании тестирования формируется файл отчетности о проведенных работах.

2.4. Выпуск (релиз) версии ПО «Платформа визуализации медицинских данных»

2.4.1. Подготовка ПО «Платформа визуализации медицинских данных» (версии) к промышленной эксплуатации

Ответственный сотрудник на основании отчета по результатам тестирования принимает решение о готовности версии ПО «Платформа визуализации медицинских данных» к промышленному запуску. Ответственный сотрудник ставит задачу отделу администрирования по обновлению версии ПО «Платформа визуализации медицинских данных», с указанием данных о пакете обновления, инструкцией установки обновления и процедуры отмены обновления.

При формировании пакета обновления ПО «Платформа визуализации медицинских данных» в промышленной среде строго запрещено включение в пакет разработческих/тестовых данных (ключей шифрования, учетных записей и паролей, ПДн и т.д.). Данные должны отличаться от тестовых/разработческих и настраиваться/определяться непосредственно в промышленной среде.

2.4.2. Запуск ПО «Платформа визуализации медицинских данных» в промышленную эксплуатацию

После согласования запуска ПО «Платформа визуализации медицинских данных» в промышленную эксплуатацию и подтверждения готовности к работе с проектом/релизом ПО «Платформа визуализации медицинских данных» в промышленном режиме СКК и СТК, ответственный сотрудник ставит задачу отделу администрирования на обновление ПО «Платформа визуализации медицинских данных» с точными сроками запуска в промышленную эксплуатацию (или предоставление заказчику пакета обновления в промышленной среде).

2.5. Сопровождение ПО «Платформа визуализации медицинских данных»

Регламент сопровождения ПО «Платформа визуализации медицинских данных» определяется и согласовывается в рамках контракта с конкретным заказчиком.

Общий порядок сопровождения ПО «Платформа визуализации медицинских данных» осуществляется СТП, которая выполняет следующие задачи:

- Мониторинг работоспособности ПО «Платформа визуализации медицинских данных» – просмотр отчетов и оповещений от системы мониторинга и логирования;
- Реагирование на инциденты – обработка инцидентов;
- Обработка обращений пользователей ПО «Платформа визуализации медицинских данных»;
- Регистрация ошибок, выявленных в процессе эксплуатации ПО «Платформа визуализации медицинских данных», и передача ошибок в отдел разработки.

2.5.1. Порядок работы при возникновении инцидентов

2.5.1.1. Участники и ответственность

Отдел администрирования несет ответственность за настройку мониторинга и отслеживание состояния эксплуатационных площадок.

Отдел технической поддержки несет ответственность за своевременное уведомление пользователей ПО «Платформа визуализации медицинских данных» о сбоях в работе ПО «Платформа визуализации медицинских данных».

Отдел разработки несет ответственность за исправление критических ситуаций совместно с Отделом администрирования.

Ответственный сотрудник несет ответственность за постановку ПО «Платформа визуализации медицинских данных» на мониторинг и своевременное уведомление заказчика о сбое в работе ПО «Платформа визуализации медицинских данных».

2.5.1.2. План реагирования

При выявлении инцидента администратор должен выполнить следующий порядок действий:

1. Уведомить ответственных о сбое:
 - (по телефону) ответственного от службы технической поддержки;
 - (по телефону) ответственного от отдела разработки;
 - (по телефону) ответственного сотрудника.

Должно быть направлено письмо с сообщением об инциденте.

Уведомление должно включать следующую информацию:

- проблема (если идентифицируется сразу), и время ее начала;
- последствия: что не работает;
- прогнозируемое время восстановления работоспособности.

2. Приступить к исправлению инцидента.

В случае, если решение проблемы начинает превышать первичную оценку, вернуться к п. 1 плана реагирования и уведомить ответственные лица о предпринимаемых действиях и новом прогнозе на исправление инцидента.

3. При исправлении инцидента сообщить участникам, приведенным в п. 1 плана реагирования о решении проблемы с описанием предпринятых действий.

2.5.1.3. Численность, функции и квалификация персонала

2.5.1.3.1. Требования и численность персонала (пользователей)

Штатный состав персонала, эксплуатирующего ПО «Платформа визуализации медицинских данных», должен формироваться на основании нормативных документов Российской Федерации и Трудового кодекса. ПО «Платформа визуализации медицинских данных» используется на персональных компьютерах, поэтому требования к организации труда и режима отдыха при работе с ней должны устанавливаться, исходя из требований к организации труда и режима отдыха при работе с этим типом средств вычислительной техники.

Деятельность персонала по эксплуатации ПО «Платформа визуализации медицинских данных» должна регулироваться должностными инструкциями.

Численность и квалификация персонала должна определяться с учетом следующих требований:

- количество сотрудников, эксплуатирующих ПО «Платформа визуализации медицинских данных», должно обеспечивать корректную работу ПО «Платформа визуализации медицинских данных»: своевременное наполнение информацией, обработку информации, выгрузку данных и т. д.;

- структура ПО «Платформа визуализации медицинских данных» предоставляет возможность управления всем доступным функционалом как одному администратору, так и предоставлять возможность разделения ответственности по администрированию между несколькими администраторами;

- для администрирования ПО «Платформа визуализации медицинских данных» к администратору не предъявляются требования по знанию всех особенностей функционирования элементов, входящих в состав администрируемых компонентов ПО «Платформа визуализации медицинских данных»;

- аппаратно-программный комплекс ПО «Платформа визуализации медицинских данных» не требует круглосуточного обслуживания и присутствия администраторов у консоли управления.

Группы пользователей ПО «Платформа визуализации медицинских данных» и их рекомендуемая численность:

- технический администратор – 2 штатные единицы;
- авторизованный пользователь – по числу администраторов и пользователей административных частей подсистем;

- неавторизованный пользователь – не являются штатной единицей, количество не ограничено.

Техническими администраторами являются представители организации, осуществляющие обслуживание и техническую поддержку ПО «Платформа визуализации медицинских данных». Имеют право осуществлять управление и настройку групп пользователей.

Авторизованными пользователями ПО «Платформа визуализации медицинских данных» должны являться должностные лица организаций, участвующих в процессах управления организацией\группой организаций.

2.5.1.3.2. Требования к квалификации и показу взаимодействия персонала

Авторизованные и неавторизованные пользователи ПО «Платформа визуализации медицинских данных» должны иметь опыт работы с персональным компьютером на уровне квалифицированного пользователя и свободно осуществлять базовые операции.

Технический администратор должен иметь знание администрирования промышленной СУБД, в частности PostgreSQL, опыт работы с персональным компьютером на базе операционных систем Linux Ubuntu на уровне квалифицированного пользователя и свободно осуществлять базовые операции.

К техническому администратору должны предъявляться следующие требования:

- стаж работы с аналогичными системами не менее 2 лет;
- знание английского языка (технический перевод).
- знание и опыт администрирования Linux;
- знание и опыт конфигурирования серверов Nginx;
- опыт работы с Python-проектами, настройки фреймворка Django, настройки серверов TomCat 8, Net.Framework 4.4.5.
- знание SQL;
- опыт администрирования систем управления базами данных PostgreSQL, в частности, знание механизмов работы СУБД, владение навыками оптимальной настройки БД под существующие нагрузки и аппаратные условия, знание механизмов репликации, резервного копирования и восстановления БД.

2.5.1.3.3. Требуемый режим работы персонала

ПО «Платформа визуализации медицинских данных» используется на персональных компьютерах, поэтому требования к организации труда и режима отдыха при работе с ней должны устанавливаться, исходя из требований к организации труда и режима отдыха при работе с этим типом средств вычислительной техники. Режим работы персонала соответствует требованиям СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к



персональным электронно-вычислительным машинам и организации работы», ТОО Р-45- 084-01 «Типовая инструкция по охране труда при работе на персональном компьютере».

Для технического администратора установлен следующий режим работы: по рабочим дням с 09:30 до 18:00.

Для авторизованных пользователей требования к режиму работы должны быть установлены правилами внутреннего трудового распорядка в соответствии с трудовым законодательством и иными нормативными правовыми актами, содержащими нормы трудового права, коллективным договором, соглашениями, а для работников, режим рабочего времени которых отличается от общих правил, установленных у данного работодателя, - трудовым договором.

3. Фактический адрес размещения инфраструктуры разработки ПО «Платформа визуализации медицинских данных»

191186, Санкт-Петербург, улица Итальянская, дом 17, литер А, офис 258. Тел.: 8 800 302 5510

4. Фактический адрес размещения разработчиков ПО «Платформа визуализации медицинских данных»

191186, Санкт-Петербург, улица Итальянская, дом 17, литер А, офис 258. Тел.: 8 800 302 5510

5. Фактический адрес размещения службы поддержки ПО «Платформа визуализации медицинских данных»

191186, Санкт-Петербург, улица Итальянская, дом 17, литер А, офис 258. Тел.: 8 800 302 5510

**ОПИСАНИЕ ТЕХНИЧЕСКОЙ АРХИТЕКТУРЫ ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ «ПЛАТФОРМА ВИЗУАЛИЗАЦИИ МЕДИЦИНСКИХ
ДАНЫХ»**

Содержание

Перечень сокращений, терминов и определений.....	24
1. Компонентная и функциональная структура программного обеспечения «Платформа визуализации медицинских данных».....	25
2. Программный интерфейс (API).....	26
2.1. Импорт/экспорт.....	26
2.2. Графики.....	26
2.3. Dashboard.....	27
2.4. Базы данных.....	28
2.5. Наборы данных.....	29
3. Структура базы данных.....	30
3.1. Структура массива «ab_role».....	30
3.2. Структура массива «ab_user».....	30
3.3. Структура массива «dashboards».....	31
3.4. Структура массива «datasources».....	32
3.5. Структура массива «dbs».....	33
4. Применяемые технологии разработки ПО «Платформа визуализации медицинских данных».....	35

Перечень сокращений, терминов и определений

Сокращение	Определение
БД	База данных
ПО	Программное обеспечение
СУБД	Система управления базами данных
JSON	Текстовый формат обмена данными
SQL	Язык структурированных запросов
URL	Унифицированный указатель ресурса

4. Компонентная и функциональная структура программного обеспечения Платформа визуализации медицинских данных»

Программное обеспечение «Платформа визуализации медицинских данных» (далее – ПО «Платформа визуализации медицинских данных») состоит из следующих компонентов:

- Веб-сервер – компонент, в котором реализована основная бизнес-логика и пользовательский интерфейс программного обеспечения. Выполняет следующие функции:
 - Управление данными, хранящимися в компоненте БД Метаданных, частности, основными сущностями системы:
 - график (chart) – отвечает за визуализацию информации, полученной из набора данных, например в виде таблицы, графика, круговой диаграммы;
 - набор данных (dataset) – набор агрегированных данных, на которые опираются графики для построения визуализации;
 - база данных (database) – сконфигурированное подключение к SQL базе данных, используемое наборами данных;
 - дашборд (dashboard) – смысловое объединение нескольких графиков, объединенных общей темой;
 - Управление учетными записями пользователей;
 - Предоставление доступа к пользовательскому интерфейсу системы;
 - Предоставления доступа к программному интерфейсу системы;
 - БД Метаданных – компонент, обеспечивающий хранение и предоставление доступа к основным сущностям системы. Реализуется поверх поддерживаемой реляционной СУБД (рекомендуется использовать PostgreSQL или аналоги).
 - Кеш данных – компонент, обеспечивающий промежуточное хранение данных, необходимых для построения графиков, имеет своей целью снизить нагрузку на серверную часть веб-сервера и внешние хранилища;
 - Очередь асинхронных запросов – компонент, обеспечивающий выполнение запросов, формирующих данные, хранящиеся в компоненте «Кеш данных» в отрыве от основных пользовательских сценариев, имеет своей целью своевременное обновление данных в компоненте «Кеш данных» для отрисовки в графиках;
 - Внешние хранилища – не являются встроенным компонентом системы, но используются в ходе работы для формирования наборов данных с последующей визуализацией в графиках. В качестве внешнего хранилища может использоваться любая совместимая реляционная база данных.

На рисунке 1 представлена структура ПО «Платформа визуализации медицинских данных»

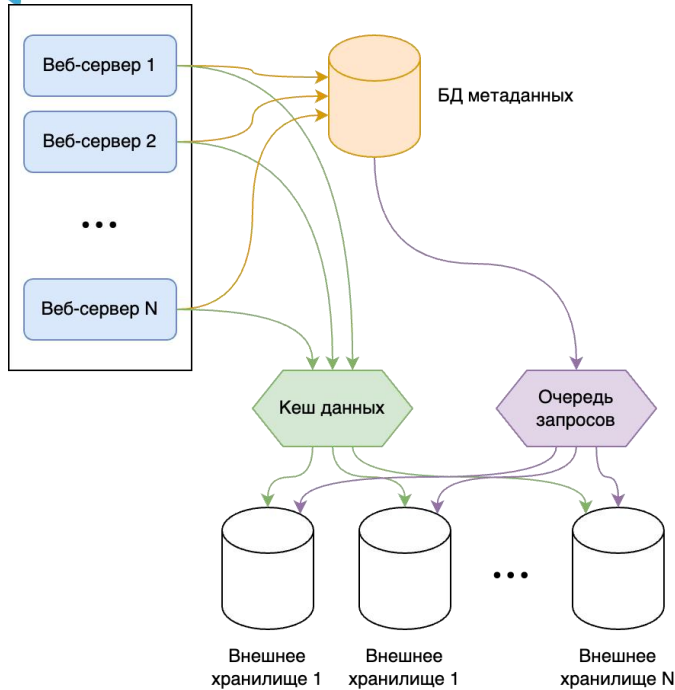


Рисунок 1 – Структура ПО «Платформа визуализации медицинских данных»

6. Программный интерфейс (API)

Приводим описание основных функций, доступных в программном интерфейсе.

6.1. Импорт/экспорт

GET /api/v1/assets/export – выгрузка полного набора конфигурации и управляемых сущностей в формате YAML.

POST /api/v1/assets/import – загрузка конфигурации и управляемых сущностей в формате YAML.

Import/export

GET	/api/v1/assets/export/	^
POST	/api/v1/assets/import/	^

6.2. Графики

DELETE /api/v1/chart – удаление графика.

GET /api/v1/chart – получение конфигурации графика.

POST /api/v1/chart – создание нового графика.

GET /api/v1/chart/_info – получение описательной информации по графику.

GET /api/v1/chart/data – получение текущего слепка данных графика.

GET /api/v1/chart/data/{cache_key} – получение закешированного слепка данных графика по идентификатору.

GET /api/v1/chart/export – выгрузка конфигурации графика в формате YAML.

GET `/api/v1/chart/favourite_status` – получение данных статуса нахождения графика в предпочтительных списках.

POST `/api/v1/chart/import` – загрузка конфигурации графика в формате YAML.

Charts ^

DELETE	<code>/api/v1/chart/</code>	▼	🔒
GET	<code>/api/v1/chart/</code>	▼	🔒
POST	<code>/api/v1/chart/</code>	▼	🔒
GET	<code>/api/v1/chart/_info</code>	▼	🔒
POST	<code>/api/v1/chart/data</code>	▼	🔒
GET	<code>/api/v1/chart/data/{cache_key}</code>	▼	🔒
GET	<code>/api/v1/chart/export/</code>	▼	🔒
GET	<code>/api/v1/chart/favorite_status/</code>	▼	🔒
POST	<code>/api/v1/chart/import/</code>	▼	🔒

6.1. Dashboard

DELETE `/api/v1/dashboard` – удаление дашборда.

GET `/api/v1/dashboard` – получение конфигурации дашборда.

POST `/api/v1/dashboard` – создание нового дашборда.

GET `/api/v1/dashboard/_info` – получение описательной информации по дашборду.

GET `/api/v1/dashboard/data` – получение текущего снимка данных дашборда.

GET `/api/v1/dashboard/export` – выгрузка конфигурации дашборда в формате YAML.

GET `/api/v1/dashboard/favourite_status` – получение данных статуса нахождения дашборда в предпочтительных списках.

POST `/api/v1/dashboard/import` – загрузка конфигурации дашборда в формате YAML.

GET `/api/v1/dashboard/related/{column_name}` – получение данных расчетного столбца связанного с дашбордом набора данных.

GET `/api/v1/dashboard/{id_or_slug}` – получение конфигурации дашборда по его идентификатору или коду из URL.

GET `/api/v1/dashboard/{id_or_slug}/charts` – получение списка связанных с дашбордом графиков по его идентификатору или коду из URL.



GET /api/v1/dashboard/{id_or_slug}/datasets – получение списка связанных с дашбордом наборов данных по его идентификатору или коду из URL.

Dashboards ^

DELETE	/api/v1/dashboard/	▼	🔒
GET	/api/v1/dashboard/	▼	🔒
POST	/api/v1/dashboard/	▼	🔒
GET	/api/v1/dashboard/_info	▼	🔒
GET	/api/v1/dashboard/export/	▼	🔒
GET	/api/v1/dashboard/favorite_status/	▼	🔒
POST	/api/v1/dashboard/import/	▼	🔒
GET	/api/v1/dashboard/related/{column_name}	▼	🔒
GET	/api/v1/dashboard/{id_or_slug}	▼	🔒
GET	/api/v1/dashboard/{id_or_slug}/charts	▼	🔒
GET	/api/v1/dashboard/{id_or_slug}/datasets	▼	🔒

6.2. Базы данных

GET `/api/v1/database` – получение конфигурации базы данных.

POST `/api/v1/database` – создание новой базы данных.

GET `/api/v1/database/_info` – получение описательной информации по базе данных.

GET `/api/v1/database/available` – проверка доступности базы данных.

GET `/api/v1/database/export` – выгрузка конфигурации базы данных в формате YAML.

POST `/api/v1/database/import` – загрузка конфигурации базы данных в формате YAML.

POST `/api/v1/database/test_connection` – проверка соединения с базой данных.

POST `/api/v1/database/validate_parameters` – валидация параметров соединения с базой данных.

Database		^
GET	<code>/api/v1/database/</code>	∨ 🔒
POST	<code>/api/v1/database/</code>	∨ 🔒
GET	<code>/api/v1/database/_info</code>	∨ 🔒
GET	<code>/api/v1/database/available/</code>	∨ 🔒
GET	<code>/api/v1/database/export/</code>	∨ 🔒
POST	<code>/api/v1/database/import/</code>	∨ 🔒
POST	<code>/api/v1/database/test_connection/</code>	∨ 🔒
POST	<code>/api/v1/database/validate_parameters/</code>	∨ 🔒

6.3. Наборы данных

DELETE `/api/v1/dataset` – удаление набора данных.

GET `/api/v1/dataset` – получение конфигурации набора данных.

POST `/api/v1/dataset` – создание нового набора данных.

GET `/api/v1/dataset/_info` – получение описательной информации по набору данных.

GET `/api/v1/dataset/distinct/{column_name}` – получение уникальных данных расчетного столбца связанного с набором данных.

POST /api/v1/dataset/duplicate – копирование набора данных.

GET /api/v1/dataset/export – выгрузка конфигурации набора данных в формате YAML.

POST /api/v1/dataset/import – загрузка конфигурации набора данных в формате YAML.

Datasets		^
DELETE	/api/v1/dataset/	⌵ 🔒
GET	/api/v1/dataset/	⌵ 🔒
POST	/api/v1/dataset/	⌵ 🔒
GET	/api/v1/dataset/_info	⌵ 🔒
GET	/api/v1/dataset/distinct/{column_name}	⌵ 🔒
POST	/api/v1/dataset/duplicate	⌵ 🔒
GET	/api/v1/dataset/export/	⌵ 🔒
POST	/api/v1/dataset/import/	⌵ 🔒

7. Структура базы данных

Компонент БД Метаданные работает под управлением любой совместимой реляционной СУБД (рекомендуется PostgreSQL или аналоги). Приведено детальное описание ключевых таблиц БД:

7.1. Структура массива «ab_role»

В массиве содержится информация о ролях пользователей.

Состав и формат данных массива «ab_role» приведен в таблице 1.

Таблица 1. Состав и формат данных массива «ab_role»

№ п.п.	Наименование показателя	Тип показателя	Использование показателя
1.	id	integer	Идентификатор роли пользователя
2.	name	character varying(64)	Наименование роли пользователя

7.2. Структура массива «ab_user»

В массиве содержится информация об учетных записях пользователей.

Состав и формат данных массива «ab_user» приведен в таблице 2.

Таблица 2. Состав и формат данных массива «ab_user»

№ п.п.	Наименование показателя	Тип показателя	Использование показателя
1.	id	integer	Идентификатор учетной записи пользователя
2.	first_name	character varying(64)	Имя пользователя
3.	last_name	character varying(64)	Фамилия пользователя
4.	username	character varying(64)	Имя учетной записи пользователя
5.	password	character varying(256)	Пароль пользователя
6.	active	boolean	Признак активности учетной записи
7.	email	character varying(64)	Адрес электронной почты пользователя
8.	last_login	timestamp	Дата-время последнего входа в си
9.	login_count	integer	Общее число входов в систему
10.	fail_login_count	integer	Количество неудачных попыток входа в систему
11.	created_on	timestamp	Дата-время создания учетной записи
12.	changed_on	timestamp	Дата-время обновления учетной записи
13.	created_by_fk	integer	Идентификатор учетной записи, создавшей текущую
14.	changed_by_fk	integer	Идентификатор учетной записи, изменившей текущую

7.3. Структура массива «dashboards»

В массиве содержится информация о дашбордах.

Состав и формат данных массива «dashboards» приведен в таблице 3.

Таблица 3. Состав и формат данных массива «dashboards»

№ п.п.	Наименование показателя	Тип показателя	Использование показателя
1.	created_on	timestamp	Дата-время создания дашборда
2.	changed_on	timestamp	Дата-время изменения дашборда
3.	id	integer	Идентификатор дашборда
4.	dashboard_title	character varying(500)	Наименование дашборда
5.	position_json	text	Данные (в формате JSON) о расположении графиков на дашборде
6.	created_by_fk	integer	Идентификатор учетной записи, создавшей дашборд
7.	changed_by_fk	integer	Идентификатор учетной записи, изменившей дашборд
8.	css	text	Исходный код (CSS) таблицы стилей дашборда
9.	description	text	Описание дашборда

№ п.п.	Наименование показателя	Тип показателя	Использование показателя
10.	slug	character varying(255)	Уникальный код дашборда для использования в URL
11.	json_metadata	text	Дополнительная описательная информация (в формате JSON)
12.	published	boolean	Признак доступности дашборда (опубликован / черновик)
13.	uuid	uuid	Идентификатор дашборда (в формате UUID), используется для экспорта-импорта данных
14.	certified_by	text	Указание на физическое лицо / организацию заверившую правильность дашборда
15.	certification_details	text	Детали заверения
16.	is_managed_externally	boolean	Признак внешнего управления дашбордом
17.	external_url	text	Внешний URL дашборда

7.4. Структура массива «datasources»

В массиве содержится информация о наборе данных.

Состав и формат данных массива «datasources» приведен в таблице 4.

Таблица 4. Состав и формат данных массива «datasources»

№ п.п.	Наименование показателя	Тип показателя	Использование показателя
1.	created_on	timestamp	Дата-время создания набора данных
2.	changed_on	timestamp	Дата-время изменения набора данных
3.	id	integer	Идентификатор набора данных
4.	datasource_name	character varying(255)	Наименование набора данных
5.	is_featured	boolean	Признак скрытости набора данных
6.	is_hidden	boolean	Признак скрытости набора данных
7.	description	text	Описание набора данных
8.	default_endpoint	text	Дополнительное описание набора данных
9.	created_by_fk	integer	Идентификатор учетной записи, создавшей набор данных
10.	changed_by_fk	integer	Идентификатор учетной записи, изменившей набор данных
11.	offset	integer	Индекс набора данных (используется для сортировки)
12.	cache_timeout	integer	Время жизни кеша набора данных
13.	filter_select_enabled	boolean	Индикатор включенности дополнительной фильтрации

№ п.п.	Наименование показателя	Тип показателя	Использование показателя
			данных, получаемых из хранилища
14.	params	character varying(1000)	Параметры запроса к хранилищу
15.	fetch_values_from	character varying(100)	Указатель на используемое хранилище
16.	schema_perm	character varying(1000)	Указатель на схему в хранилище
17.	cluster_id	timestamp	Идентификатор кластера
18.	uuid	uuid	Идентификатор набора данных (в формате UUID), используется для экспорта-импорта данных
19.	is_managed_externally	boolean	Признак внешнего управления набором данных
20.	external_url	text	Внешний URL набора данных

7.5. Структура массива «dbs»

В массиве содержится информация о наборе данных.

Состав и формат данных массива «dbs» приведен в таблице 5.

Таблица 5. Состав и формат данных массива «dbs»

№ п.п.	Наименование показателя	Тип показателя	Использование показателя
1.	created_on	timestamp	Дата-время создания базы данных
2.	changed_on	timestamp	Дата-время изменения базы данных
3.	id	integer	Идентификатор базы данных
4.	sqlalchemy_uri	character varying(1024)	URI подключения к базе данных (в формате библиотеки SQL Alchemy)
5.	created_by_fk	integer	Идентификатор учетной записи, создавшей базу данных
6.	changed_by_fk	integer	Идентификатор учетной записи, изменившей базу данных
7.	password	bytea	Пароль для подключения к базе данных
8.	cache_timeout	integer	Время жизни кеша соединения с базой данных

№ п.п.	Наименование показателя	Тип показателя	Использование показателя
9.	extra	text	Дополнительные описательные параметры
10.	select_as_create_table_as	boolean	Признак необходимости использовать конструкцию SELECT AS CREATE TABLE в запросах
11.	allow_ctas	boolean	Признак возможности использовать конструкцию CREATE TABLE AS SELECT в запросах
12.	expose_in_sqllab	boolean	Признак отображения бахи данных в редакторе SQL
13.	force_ctas_schema	character varying(250)	Дополнительный параметр конструкции CREATE TABLE AS SELECT
14.	allow_run_async	boolean	Признак возможности использования асинхронных запросов
15.	allow_dml	boolean	Признак возможности использования языка манипуляции данными
16.	verbose_name	character varying(250)	Подробное наименование базы данных
17.	impersonate_user	boolean	Признак необходимости подмены пользователя
18.	allow_multi_schema_metadata_fetch	boolean	Признак возможности получения метаданных сразу из нескольких схем базы данных
19.	allow_file_upload	boolean	Признак возможности

№ п.п.	Наименование показателя	Тип показателя	Использование показателя
			загрузки данных из файла
20.	encrypted_extra	bytea	Дополнительные описательные параметры (зашифрованные)
21.	server_cert	bytea	Данные сертификата сервера баз данных, который должен проверяться при подключении
22.	allow_cvas	boolean	Признак возможности использовать конструкцию CREATE VIEW AS SELECT в запросах
23.	uuid	uuid	Идентификатор базы данных (в формате UUID), используется для экспорта-импорта данных
24.	configuration_method	character varying(255)	Метод конфигурации подключения к базе данных (в терминах библиотеки SQL Alchemy)
25.	is_managed_externally	boolean	Признак внешнего управления базой данных
26.	external_url	text	Внешний URL базы данных

8. Применяемые технологии разработки ПО «Платформа визуализации медицинских данных»

Компонент веб-сервер состоит из двух относительно изолированных частей:

1. Серверная часть – реализована на языке программирования Python 3, использует фреймворк Flask (фреймворк с открытым исходным кодом под лицензией BSD для создания веб-приложений). В качестве библиотеки реализации непосредственно компонента взаимодействия по протоколу HTTP (и реализации программного интерфейса) используется gunicorn (библиотека с открытым исходным кодом под лицензией MIT). Для взаимодействия с компонентом БД Метаданных и Внешними хранилищами используется библиотека SQLAlchemy (библиотека с открытым исходным кодом под лицензией MIT);

2. Клиентская часть, в которой реализуется непосредственно пользовательский интерфейс, реализована на языке программирования JavaScript с использованием фреймворка React (фреймворк с открытым исходным кодом под лицензией MIT).



Прочие компоненты реализуются на базе готового ПО с открытым исходным кодом, которое при необходимости может быть заменено на аналоги:

- PostgreSQL (лицензия PostgreSQL License, совместимая с BSD и/или MIT лицензией);
- Redis (лицензия BSD).

Внешние хранилища не являются составной частью ПО «Платформа визуализации медицинских данных», могут быть подключены любые СУБД при условии их поддержки на уровне библиотеки SQLAlchemy



**ИНСТРУКЦИЯ ПО РАЗВЕРТЫВАНИЮ ЭКЗЕМПЛЯРА
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ «ПЛАТФОРМА ВИЗУАЛИЗАЦИИ
МЕДИЦИНСКИХ ДАННЫХ»**

Содержание

1. Перечень сокращений, терминов и определений.....	40
2. Программно-аппаратные требования	40
3. Инструкция по установке ПО «Платформа визуализации медицинских данных».....	40
4. Обеспечение бесперебойной и надежной эксплуатации ПО «Платформа визуализации медицинских данных».....	40
5. Контактные лица для консультации по развертыванию ПО «Платформа визуализации медицинских данных».....	41

5. Перечень сокращений, терминов и определений

Сокращение	Определение
БД	База данных
ОЗУ	Оперативное запоминающее устройство
ПО	Программное обеспечение
СУБД	Система управления базами данных
ЦПУ	Центральное процессорное устройство

6. Программно-аппаратные требования

Для установки программного обеспечения «Платформа визуализации медицинских данных» необходимы программно-аппаратные ресурсы, соответствующие рекомендованным требованиям:

Аппаратные требования к демонстрационному стенду:

ЦПУ: 8 vCPU

ОЗУ: 16 GB vRAM

Дисковое пространство: рассчитывается по формуле: общий объем исходных данных для анализа * 3

Программные требования к демонстрационному стенду:

Операционная система: Ubuntu Linux Server 18.04 LTS

7. Инструкция по установке ПО «Платформа визуализации медицинских данных»

Для установки ПО «Платформа визуализации медицинских данных» необходимо осуществить следующие действия:

1. установить последнюю версию docker;
2. развернуть архив в папку на сервере;
3. при необходимости скорректировать параметры соединения с СУБД в файле `install.sh`;
4. выполнить скрипт `install.sh` в папке развернутого архива.

8. Обеспечение бесперебойной и надежной эксплуатации ПО «Платформа визуализации медицинских данных»

Для организации бесперебойной и надёжной эксплуатации ПО «Платформа визуализации медицинских данных» должно быть обеспечено:

– ежедневное копирование всей размещенной в ПО «Платформа визуализации медицинских данных» информации и логов операций на резервный носитель, обеспечивающий возможность их восстановления;

- хранение резервной копии установочного пакета ПО «Платформа визуализации медицинских данных», а также актуальной версии конфигурационных

файлов, обеспечивающих работоспособность ПО «Платформа визуализации медицинских данных».

Используемые аппаратно-программные платформы должны обеспечивать сохранность и целостность информации при полном или частичном отключении электропитания, аварии сетей телекоммуникации, полном или частичном отказе технических средств, на которых эксплуатируется ПО «Платформа визуализации медицинских данных».

Сохранность информации должна быть обеспечена в случае:

- отключения электропитания;
- отказа одного или нескольких серверов БД;
- временного отказа линий связи. С целью обеспечения сохранности информации должно быть предусмотрено:

- еженедельное полное резервное копирование баз данных ПО «Платформа визуализации медицинских данных» на промышленных серверах БД (хранение копии – не менее 30 дней);

- ежедневное резервное копирование схем баз данных ПО «Платформа визуализации медицинских данных» на промышленных серверах БД (хранение копии – не менее 7 дней).

При выполнении резервного копирования должен осуществляться контроль выполнения процессов резервного копирования баз данных.

В случае возникновения аварии или сбоя в процессе выполнения пользовательских задач должна быть предусмотрена возможность восстановления БД до состояния на момент последней завершённой транзакции.

В случае повреждения журналов транзакций СУБД должно быть обеспечено восстановление состояния ПО «Платформа визуализации медицинских данных» на момент создания последней резервной копии данных.

9. Контактные лица для консультации по развертыванию ПО «Платформа визуализации медицинских данных»

ФИО		Должность	Реквизиты
Сорокин Максимович	Кирилл	Директор по разработке	Мобильный телефон: +7 921 883 99 43 E-mail: ksorokin@teh-lab.ru
Боровиков Юрьевич	Максим	Инженер по автоматизации ИТ-инфраструктуры	Мобильный телефон: +7 925 032 57 00 E-mail: mborovikov@teh-lab.ru



**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ «ПЛАТФОРМА ВИЗУАЛИЗАЦИИ
МЕДИЦИНСКИХ ДАННЫХ»**

ИНСТРУКЦИЯ ПО ИСПОЛЬЗОВАНИЮ СИСТЕМЫ

Аннотация

В настоящей инструкции по использованию системы дано описание и работа с программным обеспечением «Платформа визуализации медицинских данных» (далее – система).

Изображения и цвет элементов системы, используемые в документе, могут незначительно отличаться от оригинального отображения.

1. Авторизация в системе	443
2. Описание интерфейса системы	443
3. Раздел «Дашборды»	443
3.1. Особенности отображения дашбордов в табличном виде.	44
3.2. Особенности отображения дашбордов в виде блоков.	44
3.3. Создание нового Дашборда	44
3.4. Работа с дашбордом.	45
3.5. Экспорт результатов дашборда	47
4. Раздел «Графики»	47
4.1. Создание нового графика	47
5. Раздел «SQL»	49
6. Раздел «Данные»	49
6.1. Создание новой базы данных	50
6.2. Создание нового датасета.	51

1. Авторизация в системе

Для авторизации в системе необходимо ввести в поля «Логин» и «Пароль» данные для входа, затем нажать кнопку «Войти».

Данные для входа предоставляет администратор системы.

Обратите внимание, пароль чувствителен к регистру букв, т.е. password и PASSWORD — это разные пароли.

2. Описание интерфейса системы

Пользователю доступно несколько разделов:

- Дашборды;
- Графики;
- SQL;
- Данные;
- Настройки.

Чтобы перейти в нужный раздел используйте панель меню в верхней части экрана.

Чтобы добавить новый элемент в раздел используйте кнопку в виде плюса. При наведении на плюс откроется список доступных для добавления элементов. Так же добавить элемент можно в соответствующем разделе.

Кнопка в виде флага позволяет сменить язык интерфейса системы.

Кнопка в виде знака вопроса открывает пользователю документацию на систему.

Кнопка в виде жука позволяет отправить сообщение об ошибке.

3. Раздел «Дашборды»

Дашборд – это информационная панель, состоящая из одного или нескольких графиков.


Главный экран раздела отображает список доступных дашбордов. Над списком дашбордов расположен фильтр, для формирования выборки по следующим параметрам:

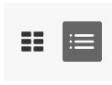
- Владелец;
- Кем создано;
- Статус;
- Избранное;
- Утверждено.

Доступен поиск по ключевым словам.

Чтобы выбрать сразу несколько дашбордов в списке используйте кнопку «Множественный выбор».

Чтобы добавить новый дашборд используйте кнопку «+Дашборд». Так же вы

можете импортировать готовый дашборд с помощью кнопки  .

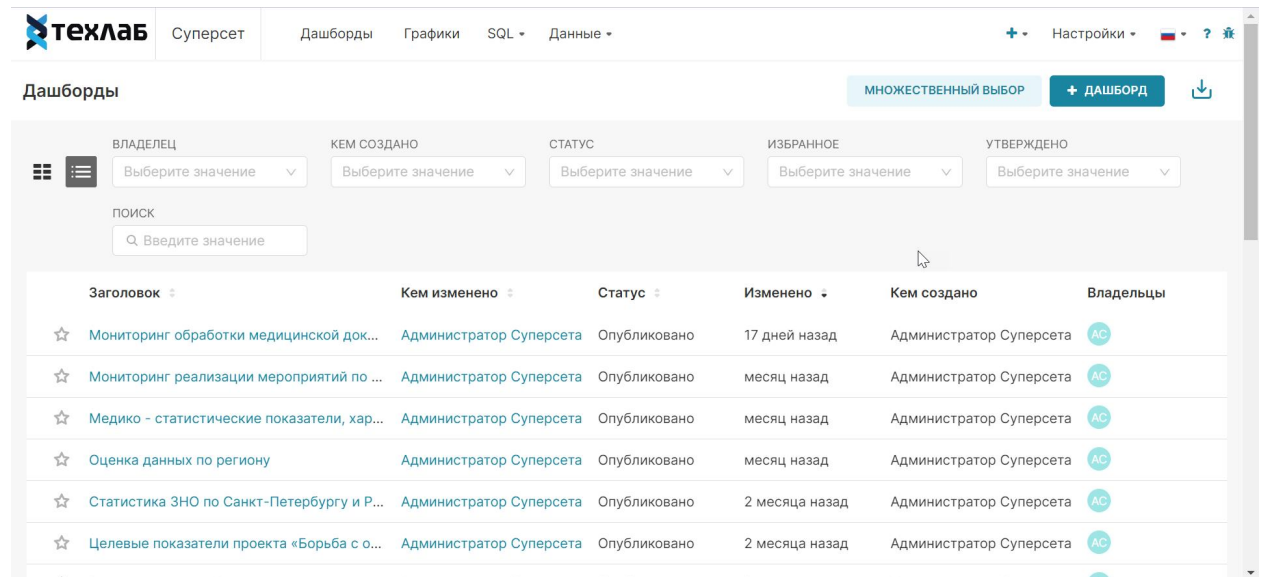
Доступно два варианта отображения списка дашбордов. Чтобы выбрать отображение используйте кнопки  . Где слева доступен табличный вид, а справа отображение дашбордов в виде блоков. В зависимости от отображения набор доступных действий со списком дашбордов будет отличаться.

3.1. Особенности отображения дашбордов в табличном виде.

Помимо названия дашборда отображает данные: кем изменено, статус, изменено, кем создан, владельцы, действия.

Столбцы таблицы можно сортировать. Страницы таблицы можно листать. Под таблицей отображено количество дашбордов в сформированной выборке.

Чтобы добавить дашборд в избранное, нажмите на звездочку в строке с нужным дашбордом.



3.2. Особенности отображения дашбордов в виде блоков.


Каждый блок – это отдельный дашборд с названием и датой последнего изменения.

Дашборд можно редактировать, экспортировать или удалить с помощью меню, которое расположено в правом нижнем углу блока.

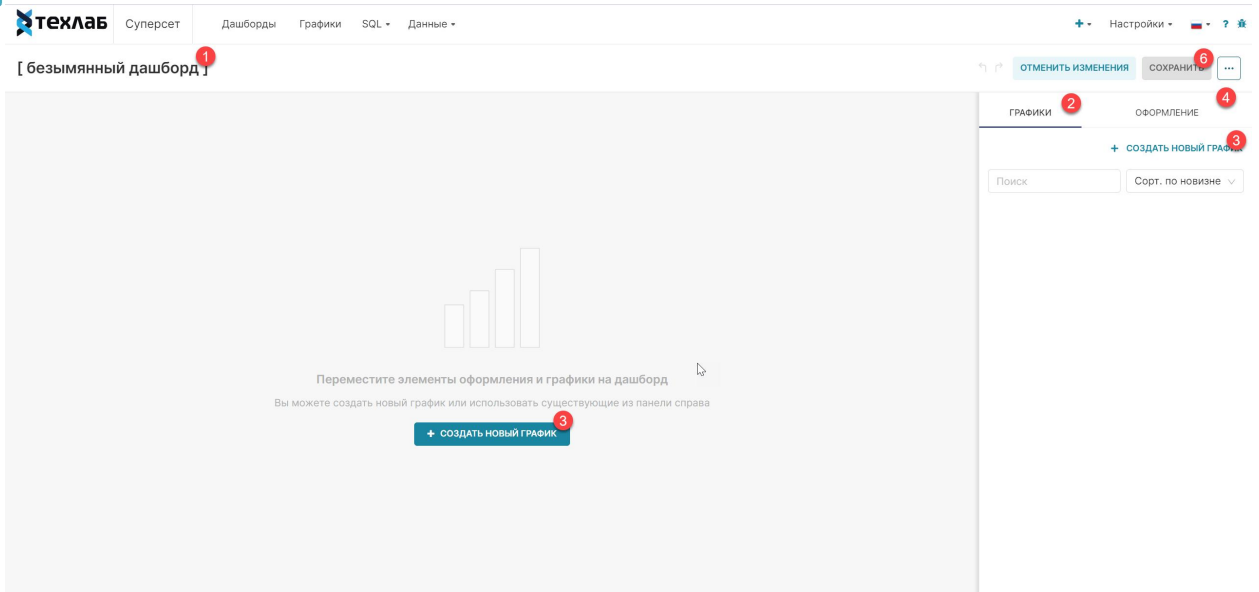
Чтобы добавить дашборд в избранное, нажмите на звездочку в блоке с нужным дашбордом.

3.3. Создание нового Дашборда

Создание Дашборда доступно только после создания графиков, так как дашборд состоит из набора графиков.

Чтобы создать новый дашборд, нажмите на кнопку «+Дашборд» в разделе «Дашборды» или на кнопку  в верхней панели системы, а затем «Дашборд».

Откроется окно создания дашборда. Задайте название (1), выберите графики, которые будут содержаться в этом дашборде (2). Так же вы можете перейти в окно создания нового графика (3). Выберите оформление дашборда (4). После настройки не забудьте сохранить изменения (6).



3.4. Работа с дашбордом.

Чтобы перейти в нужный Дашборд, нажмите на название в списке в разделе «Дашборды».

Дашборд формируется автоматически с предустановленными параметрами. Чтобы изменить параметры Дашборда воспользуйтесь панелью «Фильтры» в левой части окна. Настройте необходимые параметры и нажмите «Применить фильтры».



Параметр дашборда представляет собой условие, которое учитывается при выборке пациентов для построения результатов. Таким образом, при выборе в параметрах «Пол» → «Женский», Дашборд будет строиться исключительно по пациентам женского пола.

Параметры дашбордов бывают нескольких форматов:

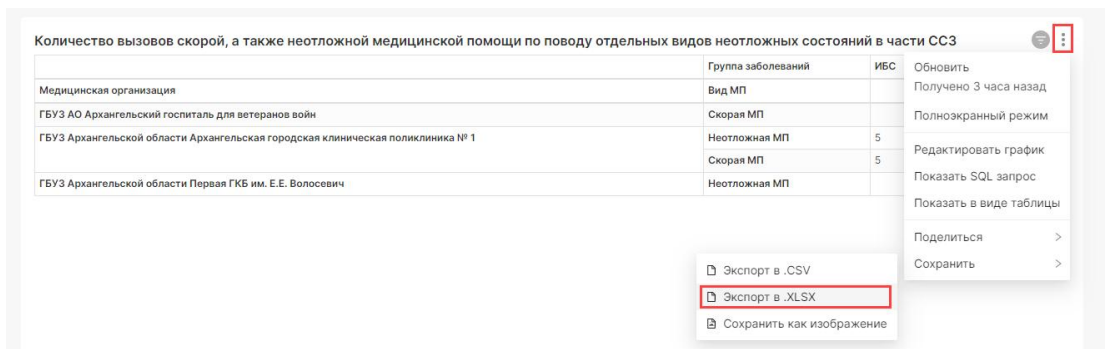
- дата (в формате: дд.мм.гг),
- число,
- выбор из списка значений (пол пациентов, МО, локализация опухоли и т. п.).

3.5. Экспорт результатов дашборда

Результаты построения можно выгрузить в файл формата XLSX:

1. Укажите необходимые параметры дашборда и нажмите «Применить фильтры».

Кнопка выгрузки расположена в правом углу таблицы с результатами построения.



2. Нажмите на кнопку выгрузки, чтобы скачать файл XLSX с результатами построения.

4. Раздел «Графики»

Раздел «Графики» содержит список графиков из которых состоят дашборды.

График — отдельный компонент визуализации данных, представляющий из себя график, таблицу, карту или фильтр.

График является визуальным представлением данных запроса. Он может быть представлен в множестве разных форм. Выбор типа диаграммы чаще всего

определяется типом используемых данных и тем, что необходимо донести до конечного пользователя отчета.

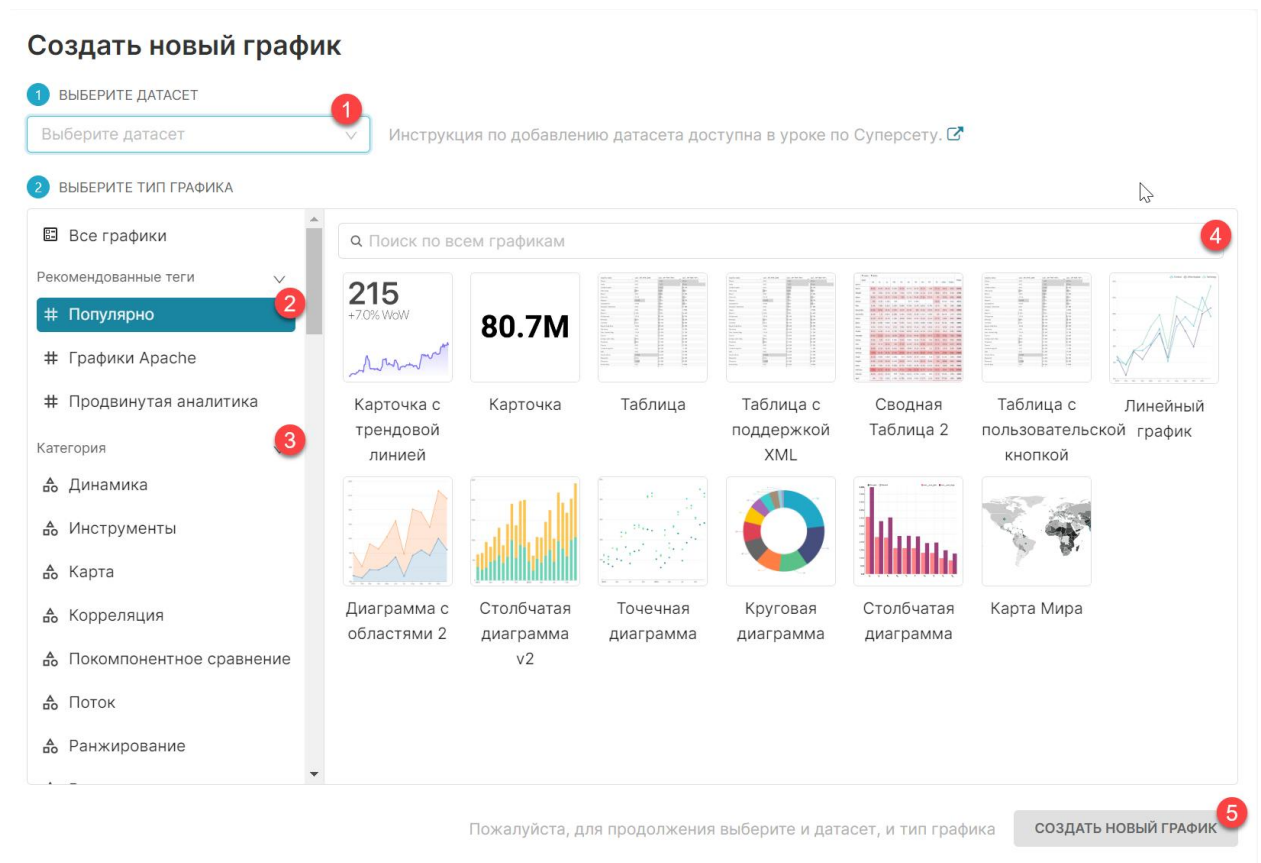
Раздел «Графики» имеет такой же функционал, как и раздел «Дашборды».

4.1. Создание нового графика

Чтобы создать новый график, нажмите на кнопку «+График» в разделе «Графики» или на кнопку **+** в верхней панели системы, затем «График».

В открывшемся окне выберите датасет из списка существующих (1). Выберите тип графика (тип визуализации). Для поиска нужного типа графика вы можете использовать теги (2), категории (3) и поиск по названию (4).

После выбора графика, нажмите на кнопку «Создать новый график»(5).

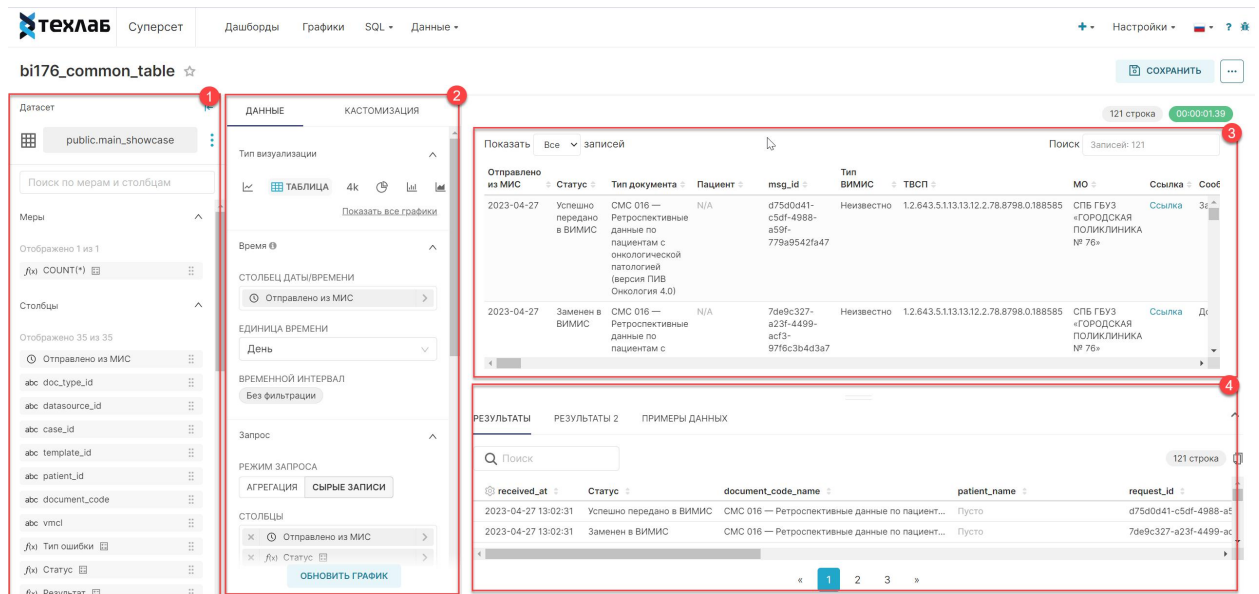


После создания, откроется окно редактирования графика. Окно состоит из 4 блоков:

1. Список мер и столбцов из выбранного датасета.
2. Настройки отображения графика. Перечень настроек блока зависит от выбранного типа визуализации при создании графика.

3. Результат отображения графика для пользователя.

4. Результаты SQL запросов, созданных при настройке графика.



The screenshot shows the 'техлаб' interface with the 'би176_common_table' dataset selected. The interface is divided into several sections:

- Датасет:** Shows the selected dataset 'public.main_showcase' and a search bar for measures and columns.
- Меры:** A list of measures including 'COUNT(*)'.
- Столбцы:** A list of columns including 'doc_type_id', 'datasource_id', 'case_id', 'template_id', 'patient_id', 'document_code', 'vmsi', 'Тип ошибки', and 'Статус'.
- ДАННЫЕ (КАСТОМИЗАЦИЯ):** A configuration panel for the visualization, including options for 'Тип визуализации' (Table), 'Время' (Time), 'СТОЛБЕЦ ДАТЫ/ВРЕМЕНИ' (Date/Time Column), 'ЕДИНИЦА ВРЕМЕНИ' (Time Unit), 'ВРЕМЕННОЙ ИНТЕРВАЛ' (Time Interval), 'Запрос' (Query), 'РЕЖИМ ЗАПРОСА' (Query Mode), 'АГРЕГАЦИЯ' (Aggregation), and 'СТОЛБЦЫ' (Columns).
- Показать:** A table showing the results of the query, with columns for 'Отправлено из МИС', 'Статус', 'Тип документа', 'Пациент', 'msg_id', 'Тип ВИМИС', 'ТВСП', 'МО', 'Ссылка', and 'Сооб'. The table contains two rows of data.
- РЕЗУЛЬТАТЫ:** A section for the results of the SQL query, with a search bar and a table showing the results of the query, with columns for 'received_at', 'Статус', 'document_code_name', 'patient_name', and 'request_id'.

5. Раздел «SQL»

В разделе можно использовать язык SQL, чтобы создать запрос в созданные в системе базы данных.

Вкладки с запросами расположены в верхней части экрана. Слева расположено поля для выбора базы данных, схемы и таблицы из базы данных. Справа расположено поля для написания запроса и для результатов запроса. Нажмите кнопку «Выполнить» после того, как сформируете запрос, чтобы получить результаты.

Созданные запросы можно сохранить с помощью кнопки «Сохранить как» или отправить ссылку на запрос с помощью кнопки «Скопировать ссылку».

При создании запросов доступно использование предзаданных параметров. Для этого перейдите в дополнительное меню, нажав на кнопку **...**. Нажмите «Параметры» в открывшемся меню. И задайте набор параметров в формате JSON (пример: `{"my_table": "foo"}`).

6. Раздел «Данные»

В разделе «Данные» доступен список созданных баз данных, датасетов, сохраненных в разделе «SQL» запросов и история этих запросов.

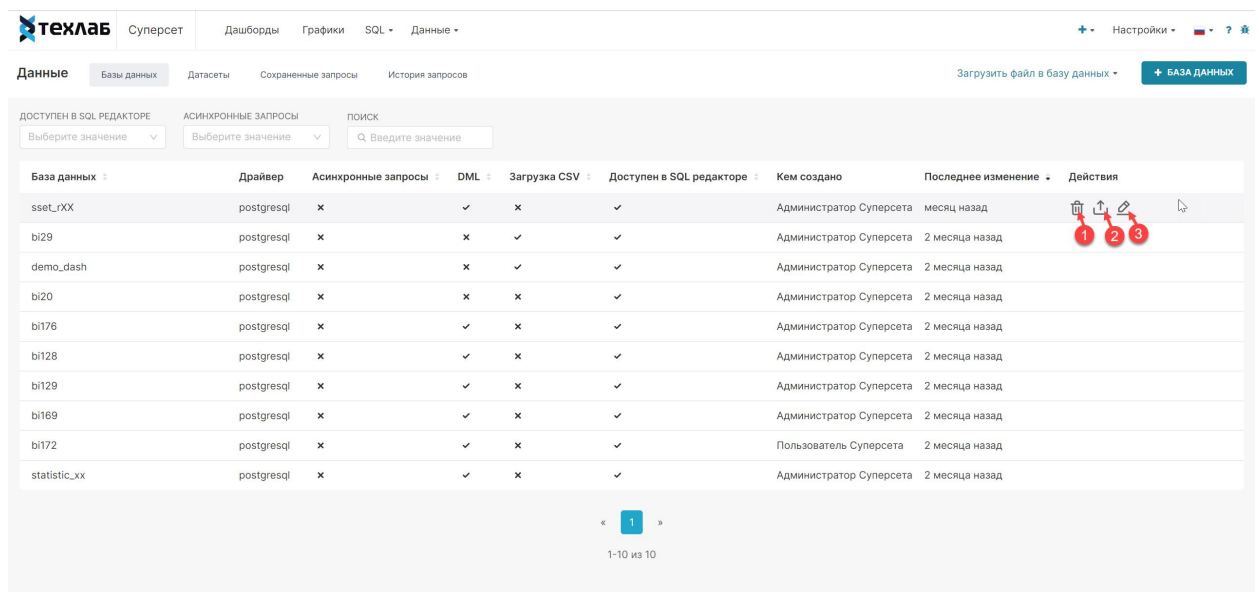
Список баз данных представляет из себя таблицу со столбцами:




- База данных;
- Драйвер;
- Асинхронные запросы;
- DML;
- Загрузка CSV;
- Доступен в SQL редакторе;
- Кем создано;
- Последнее изменение;
- Действия.

Над таблицей расположен фильтр, для создания выборки по параметрам:

- Доступен в SQL редакторе;
- Асинхронные запросы;
- Поиск по наименованию.


При наведении на строку с наименованием базы данных становятся доступными действия с базой данных: удаление, экспорт и редактирование.



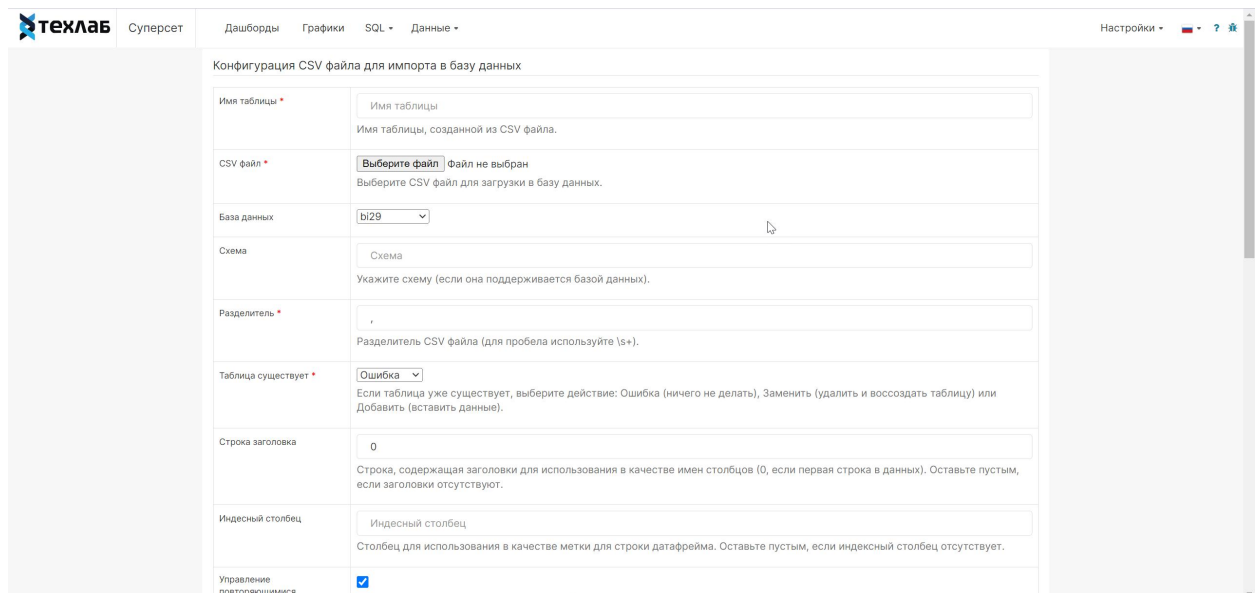
База данных	Драйвер	Асинхронные запросы	DML	Загрузка CSV	Доступен в SQL редакторе	Кем создано	Последнее изменение	Действия
sset_XX	postgresql	×	✓	×	✓	Администратор Суперсета	месяц назад	  
bi29	postgresql	×	×	✓	✓	Администратор Суперсета	2 месяца назад	1 2 3
demo_dash	postgresql	×	×	✓	✓	Администратор Суперсета	2 месяца назад	
bi20	postgresql	×	×	×	✓	Администратор Суперсета	2 месяца назад	
bi176	postgresql	×	✓	×	✓	Администратор Суперсета	2 месяца назад	
bi128	postgresql	×	✓	×	✓	Администратор Суперсета	2 месяца назад	
bi129	postgresql	×	✓	×	✓	Администратор Суперсета	2 месяца назад	
bi169	postgresql	×	✓	×	✓	Администратор Суперсета	2 месяца назад	
bi172	postgresql	×	✓	×	✓	Пользователь Суперсета	2 месяца назад	
statistic_xx	postgresql	×	✓	×	✓	Администратор Суперсета	2 месяца назад	

6.1. Создание новой базы данных

Вы можете импортировать существующую базу данных с помощью кнопки «Загрузить файл в базу данных» или подключиться к существующей базе данных с помощью кнопки «+База данных».

Такое же меню доступно при нажатии на кнопку  в верхней панели системы.

При загрузке базы данных из файла, необходимо заполнить конфигурацию файла для импорта в базу данных.



The screenshot shows a web interface for configuring a CSV file import. The title is 'Конфигурация CSV файла для импорта в базу данных'. The form contains several fields:

- Имя таблицы ***: Input field with placeholder 'Имя таблицы, созданной из CSV файла.'
- CSV файл ***: Button 'Выберите файл' and text 'Файл не выбран. Выберите CSV файл для загрузки в базу данных.'
- База данных**: Dropdown menu with 'b129' selected.
- Схема**: Input field with placeholder 'Укажите схему (если она поддерживается базой данных).'
- Разделитель ***: Input field with placeholder 'Разделитель CSV файла (для пробела используйте {s+}).'
- Таблица существует ***: Dropdown menu with 'Ошибка' selected. Text below: 'Если таблица уже существует, выберите действие: Ошибка (ничего не делать), Заменить (удалить и воссоздать таблицу) или Добавить (вставить данные).'
- Строка заголовка**: Input field with '0'. Text below: 'Строка, содержащая заголовки для использования в качестве имен столбцов (0, если первая строка в данных). Оставьте пустым, если заголовки отсутствуют.'
- Индесный столбец**: Input field with placeholder 'Индесный столбец'. Text below: 'Столбец для использования в качестве метки для строки датафрейма. Оставьте пустым, если индексный столбец отсутствует.'
- Управление повторяющимися**: Checked checkbox.

При подключении существующей базы данных настройки будут зависеть от типа базы данных. Нажмите кнопку «Проверить соединение», чтобы убедиться, что все работает от начала до конца. Если с подключением все в порядке, сохраните конфигурацию, нажав кнопку «Подключиться» в правом нижнем углу модального окна.

6.2. Создание нового датасета.

Создание датасета доступно только после создания базы данных, так как датасет создаётся на основе SQL запроса в базу данных.

Чтобы создать датасет перейдите в раздел «Данные», затем «Датасеты» и нажмите кнопку «+ Датасет» в правом верхнем углу.



Должно появиться модальное окно. Выберите базу данных, схему и таблицу, используя раскрывающиеся списки. Чтобы закончить, нажмите кнопку «Добавить» в правом нижнем углу. Теперь вы должны увидеть свой датасет в списке наборов данных.



Ценовая политика программного обеспечения «Платформа визуализации медицинских данных»

Минимальная стоимость развертывания решения составляет 500 000 рублей. Стоимость лицензии на компонент, группу компонентов или весь программный комплекс зависит от типа лицензии. Дополнительная информация предоставляется по запросу на адрес электронной почты: info@teh-lab.r