

## **МОДУЛЬ КОНСТРУИРОВАНИЯ ВИДЖЕТОВ И ВИЗУАЛИЗАЦИИ ДАННЫХ В СИСТЕМЕ «UFAM МОНИТОРИНГ»**

**Аннотация.** В статье представлена реализация модуля конструирования виджетов и визуализации данных для клиентской части информационной системы оперативного контроля производственно-экономических показателей предприятий нефтегазового сектора «UFAM Мониторинг». Результатом разработки является комплекс инструментов для гибкой интерактивной настройки компонентов визуализации статистических данных. Разработка выполнена на основе технологий создания Web-приложений.

**Ключевые слова:** визуализация данных, мониторинг производственных показателей, интерактивная настройка, виджет.

**Введение.** Нефтегазовая отрасль является одной из ключевых отраслей мировой экономики, где сбор, анализ и визуализация данных играют важнейшую роль при принятии стратегических решений. В постоянно меняющейся рыночной среде и сложной геологической среде нефтегазовых месторождений правильное и эффективное использование данных имеет первостепенное значение [1]. Однако огромный объем и сложность данных, подлежащих анализу, создают дополнительные трудности при их обработке. Для успешного анализа и обоснованного принятия решений необходимы инструменты и методы, обеспечивающие эффективную обработку и наглядное визуальное представление информации [2].

Управление бизнес-процессами в реалиях цифровой экономики имеет свои особенности, которые требуют наличия определенной технологической среды, учитывающей специфику отрасли [3]. В том числе, необходима реализация оперативного доступа к показателям, отражающим состояние компании в режиме реального времени [4]. Использование стандартных продуктов построения отчетности (MS Excel, DBeaver и т. п.) в отсутствие единой платформы имеет следствиями

- низкий уровень интерактивности графических элементов для представления данных;
- низкий уровень оперативности получаемой информации;
- низкий уровень производительности при больших объемах данных;
- отсутствие специализированных графических элементов.

Кроме того, подобные решения предполагают использование программного обеспечения, произведенного компаниями недружественных стран. В соответствии с национальной программой «Цифровая экономика РФ» [5], одной из целей которой является перевод объектов государственного значения на отечественные ИТ-решения, создание собственных цифровых платформ для обработки бизнес-данных в нефтегазовой отрасли становится особенно актуальным.

Ответом на комплекс описанных выше проблем стала разработка информационной системы мониторинга производственно-экономической деятельности нефтегазовой компании и оперативного контроля ключевых производственных показателей под названием «UFAM Мониторинг». Данная система находится в активной фазе разработки и на данный момент в ней отсутствуют необходимые компоненты визуализации.

**Проблема исследования.** Система «UFAM Мониторинг» включает клиентскую и серверную части; представляемый в данной работе модуль конструирования виджетов и визуализации данных является компонентом клиентской части.

Клиентская часть «UFAM Мониторинг» состоит из трех web-приложений:

- «Редактор виджетов» — для создания и настройки интерактивных шаблонов виджетов, а также графических элементов, связанных с источниками данных;
- «Дашборды» — для создания и настройки дашбордов — на основе шаблонов виджетов, сконструированных в приложении «Редактор виджетов»;
- «Настройки источников» — для настройки перечня таблиц доступных для построения виджетов (доступно пользователям с ролью Администратор).

На рис. 1 представлена диаграмма последовательности, иллюстрирующая взаимодействие компонентов клиентской части системы UFAM Мониторинг.

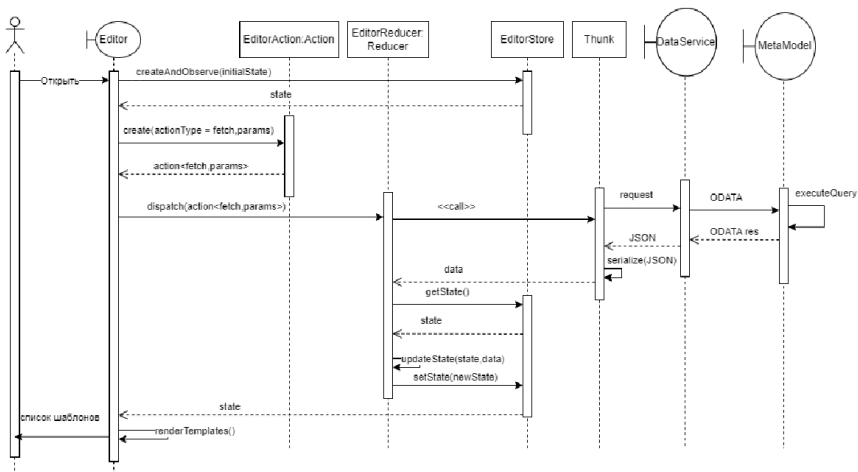


Рис. 1. Диаграмма последовательности клиентской части UFAM Мониторинг

Описание основных компонентов.

- Editor — компонент, представляющий интерфейс приложения «Редактор виджета», с которым взаимодействует пользователь. При старте приложения инициализирует состояние приложения, отвечает за создание действий (action) и их передачу в редуктор (reducer) для обновления состояния приложения.
- EditorAction — структура, сохраняющая информацию о действии пользователя.
- EditorReducer — структура, отвечающая за изменение состояния приложения в зависимости от действия пользователя.
- EditorStore — структура, отвечающая за хранение текущего состояния приложения, состоящего из срезов хранилища данных.
- DataService — сервис, находящийся на стороне сервера, на клиентской части приложения получают следующую информацию:

Шаблоны виджетов с настройками, Источники данных и описание метайнформации, Настройки виджетов и Данные из источников.

- **Metamodel** — сервис метамодели. Метамодель предназначена для выполнения функций: обеспечение подключения к нескольким базам данных, предоставление API для взаимодействия приложений платформы с базами данных и получения метайнформации, получение информации из базы данных.

В данной работе была поставлена задача разработки инструментария (компонентов и функций) для визуализации и настройки графических элементов: «Таблица», «Линейный график», «Гистограмма» и «Телеметрия». Основное функциональное требование — реализация предоставления информации в наглядном виде с возможностью гибкой интерактивной настройки элементов.

### **Проектирование и реализация модуля**

Для решения поставленных задач были использованы следующие технологии: Стандартизированный язык разметки документов HTML, препроцессор CSS — Less [6], язык программирования TypeScript [7], и библиотеки: компонентов Ant Design, Highcharts [8], React [9], Redux [10], расширение для библиотеки Redux — Redux-toolkit. Выбор такого стека технологий обусловлен по ряду причин:

- в компании OIS, занимающейся разработкой системы UFAM Мониторинг используется именно такой стек;
- все вышеперечисленные технологии являются open-source, что минимизирует санкционные риски.

В соответствии с выбранными технологиями, в первую очередь были спроектированы структуры/типы, описывающие перечень настроек для каждого типа виджетов.

Задача, связанная с необходимостью гибкой настройки пользователем визуальных компонентов, решалась следующим путем. На стороне клиентской части были созданы «срезы» (slice) хранилища состояния приложения (store), для хранения настроек виджетов различных типов в виде массивов, а также методы для манипуляции этими настройками:

- **setSettings** — установить новый массив настроек виджетов,

- `addSettings` — добавить новый экземпляр настроек виджета,
- `removeSettingsById` — удалить настройки по ключу виджета,
- `editSettingsById` — изменить настройки виджета по ключу.

Задача, связанная с сохранением и получением данных с сервера, решалась при помощи инструментария AXIOS [11] — библиотеки JavaScript, предоставляющей простой интерфейс для выполнения HTTP-запросов из браузера или с помощью Node.js. Она является альтернативой встроенному API Fetch [12] и предлагает более удобный способ работы с сетевыми запросами и обработки асинхронных действий.

После получения данных и сохранения их в состоянии приложения, необходимо с помощью созданных ранее срезов и методов преобразовать данные в формат, понятный пользователю. Все методы для преобразования были реализованы при помощи библиотек Highcharts, React и Ant Design; визуализация данных в виде графиков и диаграмм — на основе Highcharts; пользовательский интерфейс для взаимодействия — на основе React и Ant Design.

Для реализации модуля использовался язык программирования TypeScript. Необходимо было сформировать типы, которые описывают структуры для хранения настроек каждого из типов виджетов, а также выделить и переиспользовать общие настройки. На рис. 2-5 представлены диаграммы, которые описывают взаимосвязь настроек для виджетов и их состав.

`HistogramSettings` — тип, отвечающий за описание настроек всей гистограммы: `AxisX` — настройки оси x, `AxisY` — настройки оси y, `FormatingMasks` — маски для форматирования данных, отображаемых на диаграмме и во всплывающих подсказках, `seriesSettings` — настройки серий гистограммы, `Legend` — настройки легенды диаграммы, `Title` — настройки заголовка диаграммы, `Group` — настройки поля группировки.

`ColumnSettings` — тип, отвечающий за описание информации о колонках таблицы: `title` — русскоязычное название, `dataIndex` — код поля в метамодели, `propertyType` — тип данных поля.

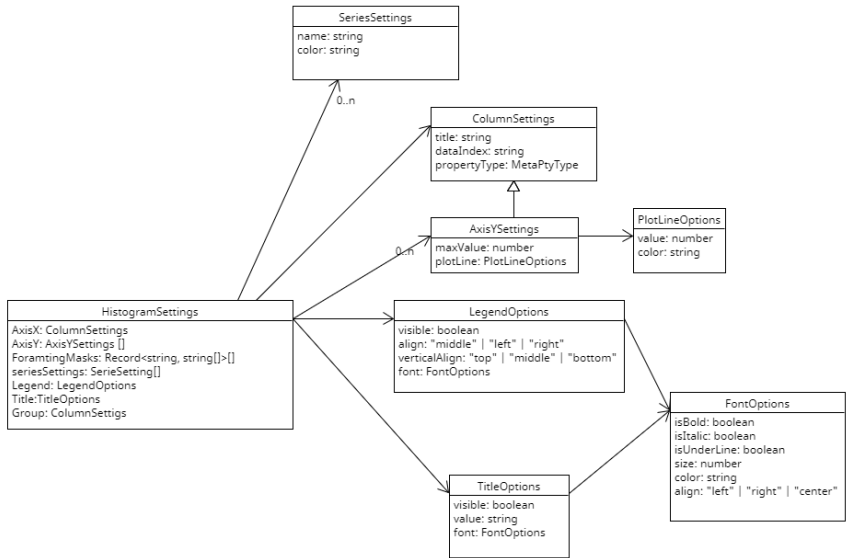


Рис. 2. Типы, описывающие настройки виджета «Гистограмма»

**AxisYSettings** — тип, описывающий настройки для оси y; расширяет структуру **ColumnSettings**, а также описывает: **maxValue** — максимальное значение допустимое на график, **plotline** — опции горизонтальной линии, отрисовываемой на графике.

**PlotLineOptions** — тип описывающий, опции для построения горизонтальной линии: **value** — значение по оси y, по которому нужно провести линию, **color**— цвет линии.

**LegendOptions** — тип, описывающий опции для отрисовки легенды на графике: **visible** — отображение или скрытие легенды, **align** — выравнивание легенды по горизонтали, **verticalAlign** — выравнивание легенды по вертикали, **font** — опции текста легенды.

**TitleOptions** — тип, описывающий опции заголовка, отображаемого для текста: **visible** — скрыть показать заголовок, **value** — содержимое заголовка.

**FontOptions** — тип, описывающий настройки текста: **isBold** — жирный, **isItalic** — курсив, **isUnderline** — подчеркнутый, **size** — размер, **color** — цвет, **align** — выравнивание текста.

FormatingMasks — массив, описывающий маски, по которым нужно форматировать данные при отображении, и поля, к которым нужно применить маску.

SeriesSettings —настройки серий гистограммы: color- цвет, name — название серии.

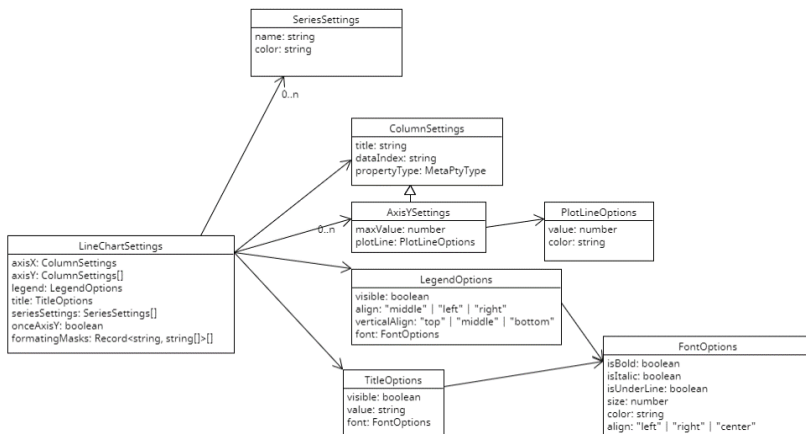


Рис. 3. Типы описывающие настройки виджета «Линейный график»

Так как большая часть настроек линейного графика совпадет по структуре с настройками виджета «Гистограмма», было принято решение использовать для этих двух виджетов одни и те же формы и структуры для хранения настроек. Отличие — только в поле Group.

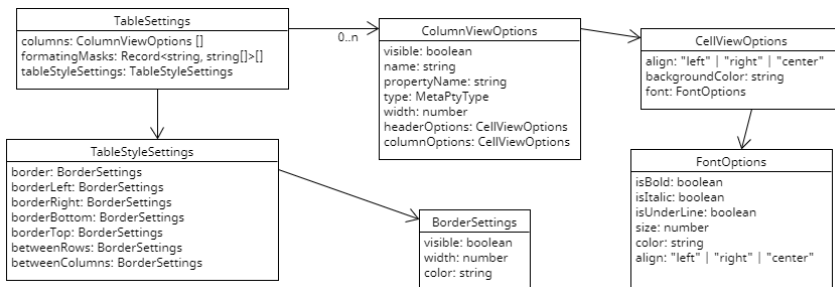


Рис. 4. Типы описывающие настройки виджета «Таблица»

**TableSettings** — тип для описания настроек таблицы: **columns** — перечень колонок таблицы, используемых при отображении, и их стилистическое оформление, **formattingMasks** — маски для форматирования данных отображаемых в таблице, **tableStyleSettings** — структура, отвечающая за хранение визуального оформления таблицы: внешние границы, границы между строками и столбцами.

**BorderSettings** — тип для описания настроек границ таблицы: **visible** — видимость границы, **width** — толщина границы, **color** — цвет границы.

**ColumnViewOptions** — тип для описания настроек каждой из колонок таблицы: **visible** — видимость колонки на виджете, **name** — название в заголовке, **propertyName** — код поля из таблицы метамодели, по которому необходимо брать данные из таблицы, **type** — тип данных поля из метамодели, **width** — ширина колонки, **headOptions** и **columnOptions** — структуры, описывающие стилистическое оформление ячеек, соответственно, в заголовке таблицы и в ячейках с данными.

**CellViewOptions** — тип для описания настроек ячеек таблицы: **align** — выравнивание текста, **backgroundColor** — цвет фона, **font** — настройки шрифта аналогичные ранее описанным одноименным структурам **FontOptions**.

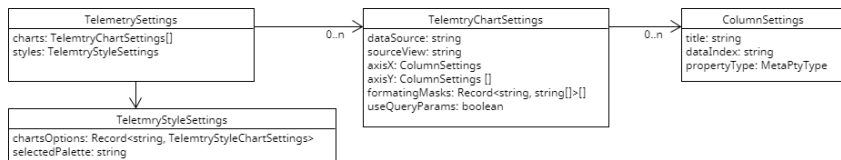


Рис. 5. Типы описывающие настройки виджета «Телеметрия»

**TelemetrySettings** — структура, описывающая настройки для виджета телеметрии, где **charts** — список настроек графиков, **styles** — стилистическое оформление для графиков виджета «Телеметрия».

**TelemetryStylesSettings** — структура, описывающая стилистическое оформление каждого графика; на данный момент — это только цвета осей.



TelelemetryChartSettings — структура, описывающая настройки каждого графика телеметрии: dataSource — название источника данных из метамодели, sourceView — название представления источника данных из метамодели, axisX и axisY — опции, аналогичные опциям виджета «Линейный график», useQueryParameters — использовать ли параметры «Навигатора» при выполнении запроса для каждого из графиков телеметрии.

ColumnSettings — структура, аналогичная одноименному типу в реализации виджета «Гистограмма».

**Пользовательский интерфейс.** После создания срезов были разработаны компоненты пользовательского интерфейса. В соответствии с выбранными технологиями, для реализации пользовательских форм и компонентов использовались библиотека React и библиотека компонентов Ant Design.

Для каждого из описанных выше графических элементов доступны следующие настройки:

- dataSource — название таблицы из метамодели;
- view — представление таблицы из метамодели.
- Для гистограммы доступны настройки:
- AxisX — наименование поля, по которому строиться ось x;
- AxisY — перечень полей, по которым строятся столбцы гистограммы;
- FormatingMasks — маски для форматирования данных, отображаемых на диаграмме и в всплывающих подсказках;
- Legend — настройки легенды, отображаемой на диаграмме: visible — отображение или скрытие легенды, align — выравнивание легенды по горизонтали, verticalAlign — выравнивание легенды по вертикали, font — опции текста легенды;
- Title — настройки заголовка, отображаемого на диаграмме: visible — скрыть/показать заголовок, value — содержимое заголовка;
- Group — поле, по которому объединяются колонки;
- SeriesSettings — настройки серий гистограммы.

- PlotLine — настройки линии, которую нужно провести по диаграмме: value — значение по оси y, lineStyle — стиль линии (прерывистая, сплошная), visible — скрыть и отобразить линию на диаграмме.

Для настроек Title и Legend также доступны настройки для оформления текста FontOptions: isBold — жирный, isItalic — курсив, isUnderline — подчеркнутый, size — размер в пикселях, color — цвет, align — выравнивание текста. Гистограмма в режиме редактирования показана на рис. 6 и 7.

Для линейного графика почти все настройки аналогичны настройкам гистограмме. Исключение составляет AxisY — перечень полей, который отвечает за построение нескольких линий графика; также имеется настройка, отвечающая за отображение одной оси y или всего выбранного множества. Линейный график в режиме редактирования с настройками источника данных представлен на рис. 8.

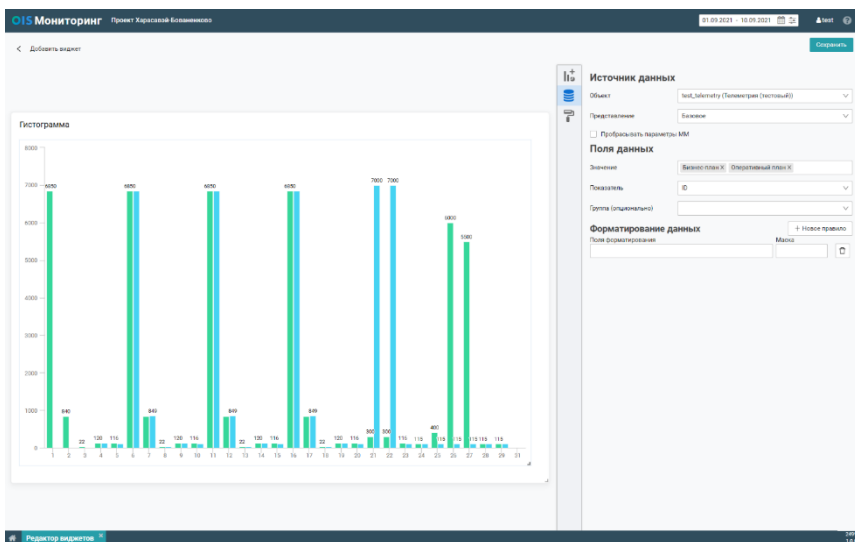


Рис. 6. Гистограмма режиме редактирования с настройками источников данных

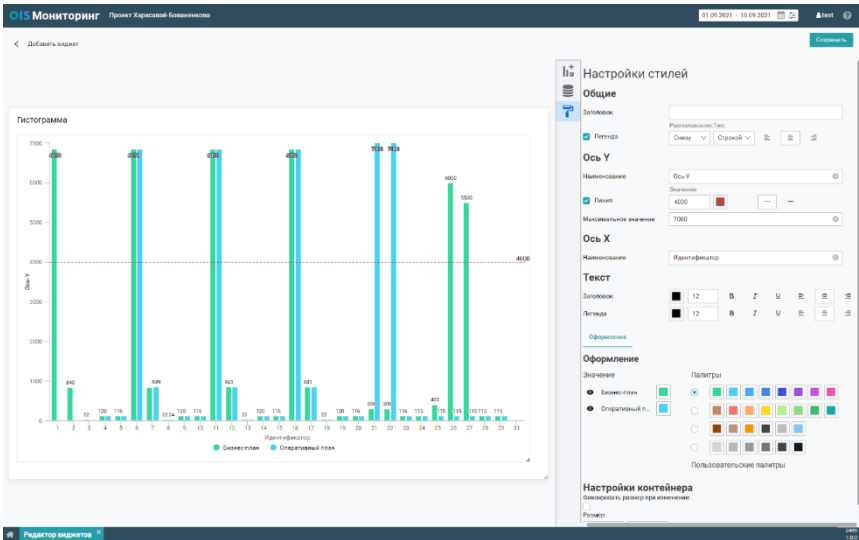


Рис. 7. Гистограмма в режиме редактирования с настройками стилистического оформления

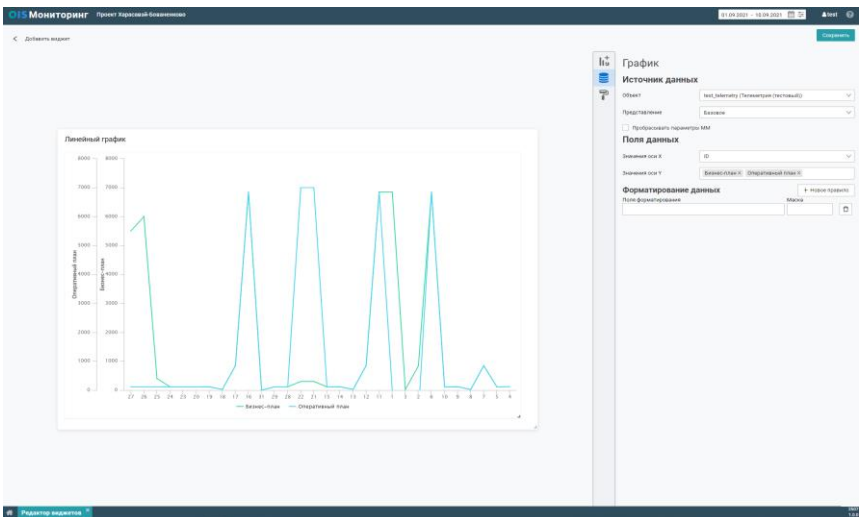


Рис. 8. Линейный график с настройками источника данных

Для таблицы доступны следующие опции настройки:

- Columns — перечень колонок, отображаемых в таблице;
- FormattingMasks — маски для форматирования данных, отображаемых в виджете;
- TableStyle — стилистического оформление границ таблицы;
- Border — настройки границ таблицы: visible — видимость границы, width — толщина границы, color — цвет границы;
- ColumnView — настройки каждой из колонок таблицы: visible — видимость колонки, name — название в заголовке, width — ширина колонки, headerOptions и columnOptions — стилистическое оформление ячеек в заголовке таблицы и соответственно в ячейках с данными;
- CellViewOptions — настройки ячеек таблицы: align — выравнивание текста, backgroundColor — цвет фона, font — настройки шрифта аналогичные ранее описанной ранее опции FontOptions.

Таблица в режиме редактирования с настройками источника данных показана на рис. 9.



Рис. 9. Таблица в режиме редактирования с настройками источника данных

Настройка ширины колонок таблицы происходит при помощи drag&drop растягивания колонки за заголовок таблицы в отображении таблицы. Изменение порядка отображаемых колонок происходит при помощи drag&drop перетаскивания заголовка в форме с настройками.

На рис. 10 представлена таблица с настройками стилистического оформления границ.

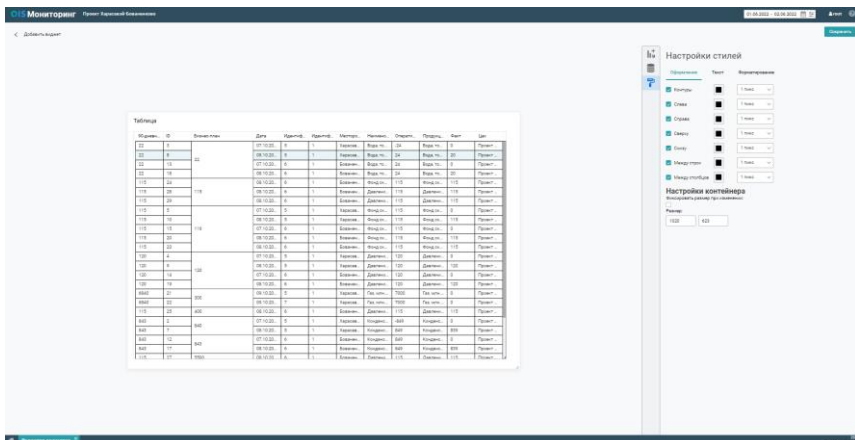


Рис. 10. Таблица в режиме редактирования с настройками стилистического оформления границ

На рис. 11 представлена таблица и форма для стилистического оформления колонок таблицы.

Для телеметрии доступны следующие опции:

- Charts — список настроек, аналогичных настройкам линейных графиков;
- Styles — стилистическое оформление графиков;
- TelemetryStylesSettings — стилистическое оформление каждого графика; на данный момент это только цвета осей у;
- TelemetryChartSettings — настройки каждого графика телеметрии: dataSource — название источника данных из метамодели, sourceView — название представления источника данных из метамодели, axisX — опция аналогичная опции виджета «Линейный график», axisY — опция аналогичная опции виджета «Линейный график», useQueryParameters — использовать ли параметры дополнительные параметры при выполнении запроса для каждого из графиков телеметрии.



**Заключение.** Разработан модуль для клиентской части информационной системы мониторинга производственно-экономической деятельности нефтегазовой компании, оперативного контроля ключевых производственных показателей руководителя, представляющий собой набор компонентов и функций для визуализации данных в режиме реального времени с гибкой пользовательской настройкой отображения графических элементов.

## СПИСОК ЛИТЕРАТУРЫ

1. Лисин Ю. В. Мониторинг Магистральных нефтепроводов в сложных геологических условиях / Ю. В. Лисин, А. А. Александров. — Текст: электронный // Наука и технологии трубопроводного транспорта нефти и нефтепродуктов. — 2013. — № 2(10). — URL: [https://niitn.transneft.ru/u/section\\_file/246751/41.pdf](https://niitn.transneft.ru/u/section_file/246751/41.pdf) (дата обращения 28.05.2023).
2. Управление данными в нефтегазовой отрасли. Особенности комплексного управления и анализа данных для нефтегазового сегмента. — Текст: электронный // IBS: официальный сайт — 2021. — URL: <https://ibs.ru/media/media/upravlenie-dannymi-v-neftegazovoy-otrasli/?ysclid=li7xj326kg364621216> (дата обращения 28.05.2023).
3. Ермолкин О. Системы оперативного контроля производительности нефтегазовых скважин / О. Ермолкин, М. Гавшин, Е. Андреев. — Текст: электронный // Разработки/ Нефтегазовая промышленность. — 2001. — № 2. — URL: <https://www.cta.ru/cms/f/366711.pdf> (дата обращения 28.05.2023).
4. Абдрахманов Н.Х. Разработка технических требований к созданию системы оперативного мониторинга и управления промышленной и экологической безопасностью опасных производственных объектов на основе минимизации рисков / Н.Х. Абдрахманов, Н.В. Шутов, К.Н. Абдрахманова, В.В. Ворохобко, Р.А. Шайбаков. — Текст: непосредственный // Нефтегазовое дело. — 2015. — № 4. — С. 497-511.
5. Национальная программа «Цифровая экономика Российской Федерации», утверждена протоколом заседания президиума Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам от 4 июня 2019 г. № 7. — Текст: электронный // Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации: официальный сайт — 2023. — URL: [https://digital.gov.ru/ru/activity/directions/858/?utm\\_referrer=https%3a%2f%2fwww.google.com%2f](https://digital.gov.ru/ru/activity/directions/858/?utm_referrer=https%3a%2f%2fwww.google.com%2f) (дата обращения 28.05.2023).

6. Less Documentation: Документация языка стилей Less: [сайт]. — URL: <https://lesscss.org/> (дата обращения 28.05.2023). — Текст электронный.
7. TypeScript Handbook: Документация языка TypeScript: [сайт]. — URL: <https://www.typescriptlang.org/docs/handbook/intro.html> (дата обращения 28.05.2023). — Текст электронный.
8. Highcharts API Reference: Документация библиотеки Highcharts: [сайт]. — URL: <https://www.highcharts.com/> (дата обращения 28.05.2023). — Текст электронный.
9. React Documentation: Документация библиотеки React: [сайт]. — URL: <https://react.dev/> (дата обращения 28.05.2023). — Текст электронный.
10. Redux Documentation: Документация библиотеки Redux: [сайт]. — URL: <https://redux.js.org/> (дата обращения 28.05.2023). — Текст электронный.
11. Axios Documentation: Документация библиотеки Axios: [сайт]. — URL: <https://axios-http.com/docs/> (дата обращения 28.05.2023). — Текст электронный.
12. MDN Fetch API Documentation: Официальная документация разработчиков Mozilla Documentation Network: [сайт]. — URL: [https://developer.mozilla.org/en-US/docs/Web/API/Fetch\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API) (дата обращения 28.05.2023). — Текст электронный.