



**Nafta College**  
Online Knowledge

# РЕТРОКУП

**Программа для интерактивного  
обучения управлению  
нефтегазодобывающими активами и  
нефтяному инжинирингу**

**PetroCup**

**Описание функциональных характеристик ПО**

**2021**

# Содержание

СОДЕРЖАНИЕ .....	2
ОПРЕДЕЛЕНИЯ .....	4
ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ .....	5
ВВЕДЕНИЕ .....	7
<b>1. МОДУЛЬ «РАСЧЕТ ПОВЕРХНОСТНОЙ ИНФРАСТРУКТУРЫ МЕСТОРОЖДЕНИЙ» .....</b>	<b>8</b>
1.1. НАЗНАЧЕНИЕ .....	8
1.2. ВХОДНЫЕ ДАННЫЕ (SOLVER INPUTS) .....	8
1.3. ВЫХОДНЫЕ ДАННЫЕ (SOLVER OUTPUTS) .....	9
1.4. ОБРАБОТКА ОШИБОК .....	9
1.5. СТРУКТУРНАЯ ДИАГРАММА ПРОГРАММНОГО МОДУЛЯ .....	11
1.6. РАЗРАБОТКА СХЕМЫ РАБОТЫ ПРОГРАММНОГО МОДУЛЯ И ЕЕ ОПИСАНИЕ .....	12
1.7. ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС .....	12
1.8. ОПИСАНИЕ ИСПОЛЬЗОВАННЫХ ОПЕРАТОРОВ И ФУНКЦИЙ .....	13
<b>2. МОДУЛЬ «БУРЕНИЕ СКВАЖИН» .....</b>	<b>15</b>
2.1. НАЗНАЧЕНИЕ .....	15
2.2. ВХОДНЫЕ ДАННЫЕ (SOLVER INPUTS) .....	15
2.3. ВЫХОДНЫЕ ДАННЫЕ (SOLVER OUTPUTS) .....	15
2.4. ОБРАБОТКА ОШИБОК .....	16
2.5. СТРУКТУРНАЯ ДИАГРАММА ПРОГРАММНОГО МОДУЛЯ .....	17
2.6. РАЗРАБОТКА СХЕМЫ РАБОТЫ ПРОГРАММНОГО МОДУЛЯ И ЕЕ ОПИСАНИЕ .....	18
2.7. ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС .....	18
2.8. ОПИСАНИЕ ИСПОЛЬЗОВАННЫХ ОПЕРАТОРОВ И ФУНКЦИЙ .....	19
<b>3. МОДУЛЬ «ГИДРОРАЗРЫВ ПЛАСТА» .....</b>	<b>21</b>
3.1. НАЗНАЧЕНИЕ .....	21
3.2. ВХОДНЫЕ ДАННЫЕ (SOLVER INPUTS) .....	21
3.3. ВЫХОДНЫЕ ДАННЫЕ (SOLVER OUTPUTS) .....	22
3.4. ОБРАБОТКА ОШИБОК .....	22
3.5. СТРУКТУРНАЯ ДИАГРАММА ПРОГРАММНОГО МОДУЛЯ .....	22
3.6. РАЗРАБОТКА СХЕМЫ РАБОТЫ ПРОГРАММНОГО МОДУЛЯ И ЕЕ ОПИСАНИЕ .....	23
3.7. ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС .....	23
3.8. ОПИСАНИЕ ИСПОЛЬЗОВАННЫХ ОПЕРАТОРОВ И ФУНКЦИЙ .....	24
<b>4. МОДУЛЬ «КОМПЛЕКС ГЕОФИЗИЧЕСКИХ ДАННЫХ» .....</b>	<b>26</b>
4.1. НАЗНАЧЕНИЕ .....	26
4.2. ВХОДНЫЕ ДАННЫЕ (SOLVER INPUTS) .....	26

4.3. ВЫХОДНЫЕ ДАННЫЕ .....	26
4.4. ОБРАБОТКА ОШИБОК .....	27
4.5. СТРУКТУРНАЯ ДИАГРАММА ПРОГРАММНОГО МОДУЛЯ .....	28
4.6. РАЗРАБОТКА СХЕМЫ РАБОТЫ ПРОГРАММНОГО МОДУЛЯ И ЕЕ ОПИСАНИЕ .....	29
4.7. ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС .....	29
4.9. ОПИСАНИЕ ИСПОЛЬЗОВАННЫХ ОПЕРАТОРОВ И ФУНКЦИЙ .....	30
<b>5. МОДУЛЬ «ИНТЕРФЕЙСНАЯ ЧАСТЬ «RETROCUP WEB-INTERFACE»» .....</b>	<b>32</b>
5.1. НАЗНАЧЕНИЕ .....	32
5.2. ПРОВЕРКА ВВОДА ПОЛЬЗОВАТЕЛЯ .....	32
1.5.3. СТРУКТУРНАЯ ДИАГРАММА ПРОГРАММНОГО МОДУЛЯ .....	34
5.4. ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС .....	35
5.5. ОПИСАНИЕ ИСПОЛЬЗОВАННЫХ ОПЕРАТОРОВ И ФУНКЦИЙ .....	36

## ОПРЕДЕЛЕНИЯ

- **Виртуальное синтетическое месторождение** – это компьютерная модель, максимально реалистично имитирующая геологические и технологические условия и процессы нефтегазового месторождения;
- **Добыча и управление разработкой** – управление разработкой добывающего актива на основе истории разработки;
- **Жизненный цикл разработки месторождения** – это стадии разработки углеводородных месторождений, которые описываются как правило пятью стадиями: поиск и разведка углеводородных запасов, интенсивное освоение месторождения, характеризующийся ростом объемов добычи, стабилизация темпов роста и выхода на максимальный уровень добычи, равномерное снижение добычи и завершающая (поздняя) стадия - заметное сокращение добычи, приводящее к потере рентабельности;
- **Математическая модель** - точное представление реальных объектов, процессов или систем, выраженное в физических терминах и сохраняющее существенные черты оригинала;
- **Проектирование разработки** - составление генерального проекта разработки на основе первичных данных геологоразведочных работ и пилотной добычи;
- **ПРОГРАММНЫЙ ЛОГ** – это файлы, содержащие системную информацию работы сервера, в которые заносятся определенные действия пользователя или программы;
- **Тестирование автоматизированных систем управления разработкой** – тестирование программных решений на основе мультивариантных расчетов и систем искусственного интеллекта;
- **Цифровой двойник** - единая модель, достоверно описывающая все процессы и взаимосвязи как на отдельном объекте, так и в рамках целого производственного актива в виде виртуальных установок и имитационных моделей;
- **Цифровое месторождение** — это программное обеспечение, включающее набор приложений, которые позволяют описывать поведение месторождения на компьютере;
- **ОН ЛОГ** – лог открытого ствола скважины;
- **PetroCup** — тренажер в области анализа и проектирования разработки цифровых месторождений нефти и газа.

## ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

– **ГДИС** (Гидродинамические исследования скважин) - совокупность различных мероприятий, направленных на измерение определенных параметров (давление, температура, уровень жидкости, дебит и др.) и отбор проб пластовых флюидов;

– **ГДК** (Гидродинамический каротаж) – определение фильтрационных характеристик пластов-коллекторов и параметров, характеризующих проницаемость пластов. Измерения выполняют с помощью скважинного прибора;

– **ГИС** (Геофизические исследования скважин) — комплекс методов разведочной геофизики, используемых для изучения свойств горных пород в околоскважинном и межскважинном пространствах;

– **ГРП** (Гидравлический разрыв пласта) – популярный метод, интенсивной добычи нефти при разработке низкопроницаемых нефтегазовых коллекторов;



– **ГРР** (Геолого-разведочные работы) — это совокупность взаимосвязанных, применяемых в определённой последовательности работ по изучению недр с целью поиска и подготовки к эксплуатации залежей углеводородов;

– **ГТМ** (Геолого-технические мероприятия) — это работы, проводимые на скважинах с целью регулирования разработки месторождений и поддержания целевых уровней добычи нефти. С помощью геолого-технических мероприятий нефтедобывающие предприятия обеспечивают выполнение проектных показателей разработки месторождений;

– **ЗКЦ** (Заколонная или затрубная циркуляция) – это движение флюида по стволу скважины за обсадной колонной (заколонный переток пластовых флюидов), который как правило, возникает по причине некачественного цементирования ствола скважины.

– **МСГРП** (Многостадийный гидроразрыв пласта) – одна из самых передовых технологий в нефтяной отрасли промышленности, наиболее эффективная для горизонтальных скважин;

– **НИОКР** – научно-исследовательские и опытно-конструкторские разработки;



– **ПГИ** (Промыслово-геофизические исследования) – являются стандартными методами анализа добычи или закачки флюида для любой скважины. Данная информация является важной, поскольку она объясняет распределение добычи или закачки флюида по зонам перфорации в стволе скважины или в интервале притока в стволе скважины;

– **ПД** (Пластовое давление) – это давление, под которым находятся жидкость (нефть, вода) и газ, насыщающие поровое пространство и (или) трещины коллекторов нефтяных и газовых месторождений. Оно является существенным показателем для характеристики режима эксплуатации нефтяных и газовых месторождений;

– **ПО** (Программное Обеспечение)

– **РПП** (Резервуарный профиль потока) – мультифазный профиль потока по пластам на основе термосимуляций и спектральной шумометрии (профиль потока за колонной);

– **СПП** (Скважинный профиль потока) – мультифазный профиль потока по стволу скважины на основе механической расходомерии и датчиков состава.

## ВВЕДЕНИЕ

Цифровые месторождения (ЦМ) набирают популярность и применение в разных сферах нефтегазовой промышленности. Важное место занимает концепция ЦМ в образовательном процессе. С помощью них можно обучать и тестировать:

- отдельных специалистов (геологов, петрофизиков, гидродинамиков, технологов отдела добычи, интерпретаторов ГДИ, ГИС и ПГИ, разработчиков, проектировщиков);
- комплексному анализу промысловых данных;
- принятию решений по ГТМ или выбор кандидатов для исследований.

Предлагаемый продукт ПО PetroCup предназначен для обучения и тестирования в следующих областях разработки нефтегазовых месторождений: поиск и оценка запасов, проектирование разработки, добыча и управление разработкой, тестирование автоматизированных систем управления разработкой, интерпретация результатов ГИС в открытом стволе, интерпретация результатов ПГИ и ГДИ, управление режимами работы скважин и управление портфелем активов.

Таким образом, основная цель выполнения НИОКР – разработка ПО PetroCup до состояния коммерческого (рыночного) продукта в соответствии с актуальными предложениями и рекомендациями пользователей ПО – нефтегазодобывающих и нефтесервисных компаний, а также профильных высших учебных заведений, который объединит в себе стадии жизненного цикла разработки месторождения и выполнение сопутствующих геолого-технических мероприятий в единый автоматизированный вычислительный процесс.

# 1. МОДУЛЬ «РАСЧЕТ ПОВЕРХНОСТНОЙ ИНФРАСТРУКТУРЫ МЕСТОРОЖДЕНИЙ»

## 1.1. Назначение

Главной задачей модуля «Расчет поверхностной инфраструктуры месторождений» является обучение пользователей и приобретение ими опыта на основании чего, будут сокращены затраты на обустройство промысла, будет упрощена автоматизация процессов добычи и обслуживания, а также будет обеспечено охрану окружающей среды при освоении нефтяных и газовых месторождений.

Разработанный программный модуль позволяет выполнять расчеты с учетом всех основных элементов поверхностной промысловой инфраструктуры и позволяет выполнять расчеты с учетом взаимосвязи различных объектов:

- расчеты гидравлических процессов трубопроводов;
- балансов жидкости и газа;
- экономической эффективности вариантов обустройств.

Модуль позволяет:

- строить и рассчитывать поверхностную сеть сбора как саму по себе, так и полностью интегрированную с фильтрационной моделью пласта;
- создавать и редактировать сеть сбора;
- создавать библиотеки элементов: скважина, нагнетательная скважина, труба, насос, штуцер, сепаратор.

Для разработки программного модуля использованы языки программирования JavaScript, Python.

## 1.2. Входные данные (Solver Inputs)

К входным данным относятся:

$p_{in}$	входное давление на питающем насосе
$G(p)$	модель подачи питательного насоса в виде функции таблицы
$K(p)$	модель расхода потока из трубопровода в виде табличной функции
$p$	входное давление



$q_i(p), \forall i \in [1 \dots N]$	модель потока на $i$ -ой стоковой трубе
$Z_{FP}$	абсолютная вертикальная отметка источника питательной трубы
$Z_{SP}$	абсолютная вертикальная отметка разветвителя
$L_{FP}$	расстояние между разветвителем и источником питающей трубы
$b_{(FP_w)}$	коэффициент трения питающей трубы
$p$	давление на устье нагнетательной скважины в точке питания
$p_{max}$	максимальное давление в точке питания
$q_{max}$	максимальный расход в точке питания

### 1.3. Выходные данные (Solver Outputs)

К выходным данным относятся:

$q^*$	стабилизированный расход потока (рабочая точка)
$p^*$	стабилизированное давление соединения (рабочая точка)
$q$	стабилизированный входной дебит
$p_i, \forall i \in [1 \dots N]$	стабилизированный расход давления разветвителя
$q = const$	расход потока на поверхности в точке питания

### 1.4. Обработка ошибок

Интерфейс спроектирован так, чтобы исключить некорректные вводимые данные.

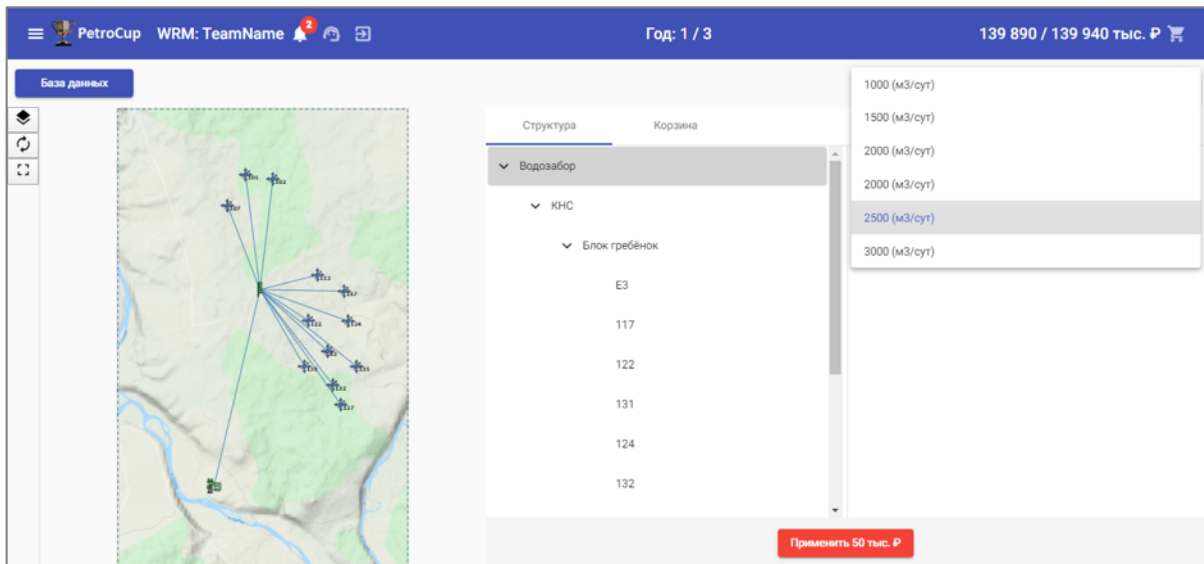


Рис. 1.1.4.1. Пример ввода значений при выборе параметров устьевой инфраструктуры

В случае возникновения непредвиденных ошибок пользователь получает уведомление:

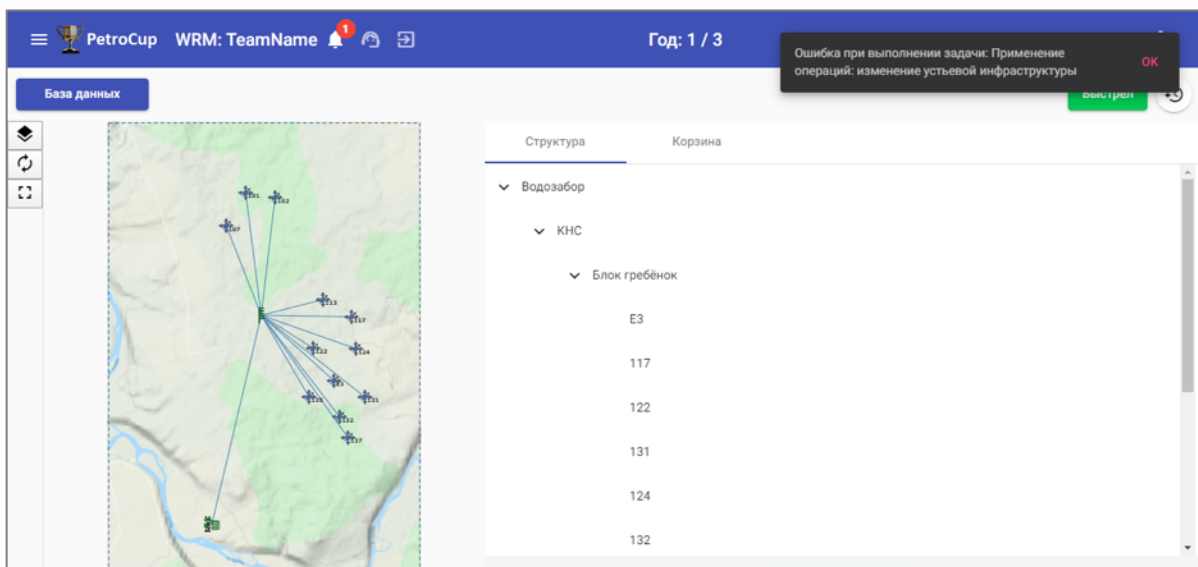


Рис. 1.1.4.2. Сообщение об ошибке, в случае возникновения ошибки при редактировании устьевой инфраструктуры

Также пользователю доступна форма оперативной связи с тех. поддержкой в случае непредвиденных ошибок.

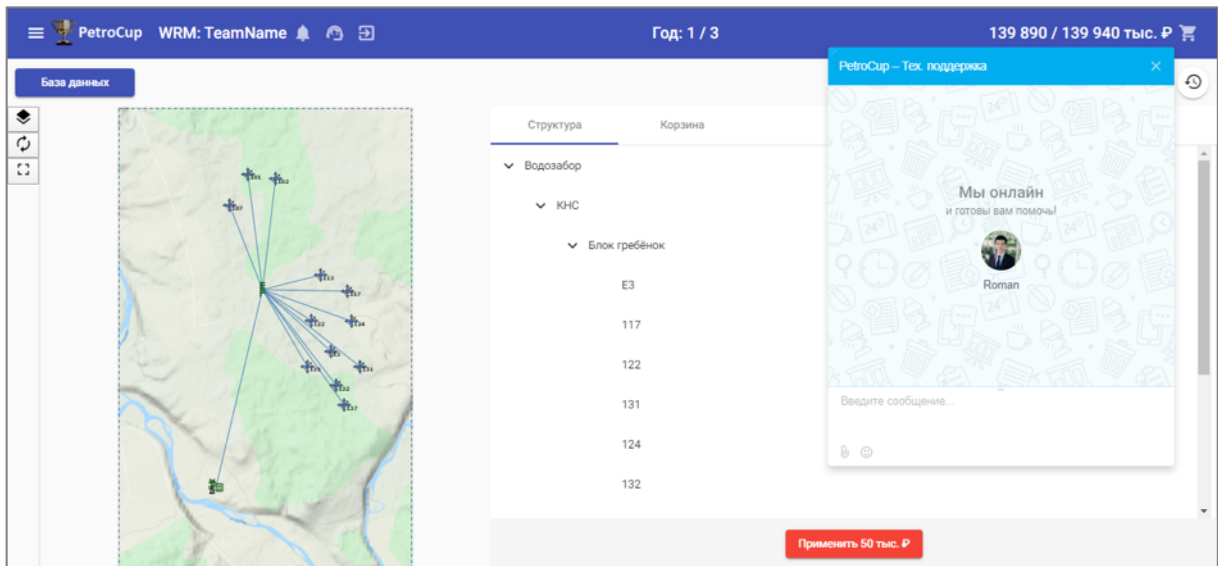


Рис. 1.1.4.3. Форма оперативной связи с тех. поддержкой

## 1.5. Структурная диаграмма программного модуля

Модуль имеет структуру, представленную на рисунке 1.1.5.1:

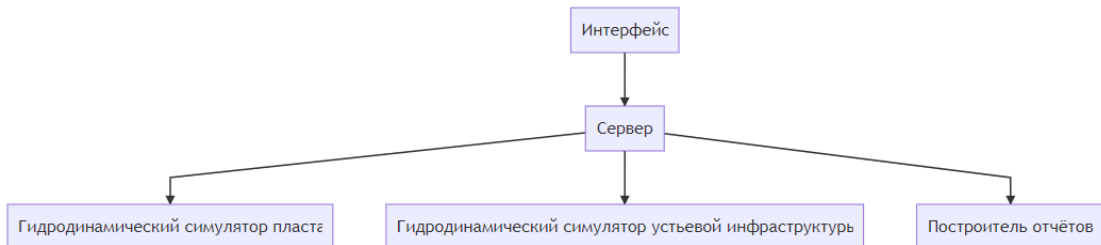


Рис. 1.1.5.1. Структурная диаграмма программного модуля

Структурные элементы:

- интерфейс - составляющая интерфейса, отвечающая за ввод параметров пользователем (подробнее о работе интерфейса в пункте 1.5);
- сервер - диспетчер для сообщений клиента;
- гидродинамический симулятор пласта - процедура, проводящая расчёт параметров пласта необходимых для поиска режима работы устьевой инфраструктуры [6];
- построитель отчётов - процедура для представления результатов работы модуля в читаемый человеком формат.

## 1.6. Разработка схемы работы программного модуля и ее описание

Модуль выполняет две задачи:

1. Редактирование параметров устьевой инфраструктуры [1];
2. Симуляция работы устьевой инфраструктуры.

Блок-схема работы модуля устьевой инфраструктуры представлена на рисунке 1.1.6.1:



Рис. 1.1.6.1. Схема программного модуля устьевой инфраструктуры

В данном модуле данные, введённые пользователем в интерфейсе, поступают на сервер, который последовательно вызывает вспомогательные процедуры, указанные на схеме.

После этого, результаты симуляции становятся доступными для просмотра в интерфейсе.

## 1.7. Пользовательский интерфейс

Пользовательский интерфейс (ПИ) – это набор программных элементов, обеспечивающих взаимодействие пользователя и компьютера для управления работой ПО и получения требуемого результата.

Интерфейс для программного модуля «Расчет поверхностной инфраструктуры месторождений» представлен на рисунке 1.1.7.1:



Рис. 1.1.7.1. Вид пользовательской формы

Описание пользовательской формы:

1. поле выбора типа поверхностной инфраструктуры;
2. карта;
3. поле выбора объекта инфраструктуры;
4. поле изменения параметров выбранного объекта;
5. кнопка применения изменений.

## 1.8. Описание использованных операторов и функций

Выше представлен фрагмент кода модуля «Расчет поверхностной инфраструктуры месторождений» выполняющий расчёт установившегося режима по части системы поддержания пластового давления.

Описание используемых в коде конструкций:

- ... *if условие else* ... - тернарный оператор, возвращающий левую часть выражения, если условие логически верно и правую в противном случае;
- *for значение in коллекция* - конструкция, позволяющая перебрать все значения в некоторой коллекции;
- [*выражение(значение) for значение in коллекция*] - быстрое создание новой коллекции по значениям другой коллекции с помощью вычисления некоторого выражения.

Описание используемых в коде переменных:

- *self* - структура, содержащая необходимую для решения задачи информацию;
- *result* - табличная функция с соответствием входных давлений и режима работы участка устьевой инфраструктуры.

Описание используемых в коде функций и методов:

- `converter.from_bar_to_Pa` - конвертер единиц измерений;
- `converter.from_day_to_sec` - конвертер единиц измерений;
- `np.linspace (a, b, n)` - создание массива из n чисел в интервале от a до b;
- `self.get_rate_by_choke` - функция для расчёт режима одной скважины по заданным параметрам;
- `any` - проверяет, что хотя бы один из аргументов является логически истинным;
- `sum` - вычисляет сумму;
- `reverse_simple_pipe_pressure_loss` - функция для пересчёта давления по направлению, обратному направлению потока.

## 2. МОДУЛЬ «БУРЕНИЕ СКВАЖИН»

### 2.1. Назначение

Главной задачей разрабатываемого модуля «Бурение скважин» является обучение пользователей проектировать траектории ствола скважин для корректного вскрытия объектов разработки, что позволит увеличить эффективность разработки нефтяных и газовых месторождений.

Разработанный программный модуль позволит спроектировать пользователю профиль траектории скважины и корректно перевести этот профиль в программный язык, понятный гидродинамическому симулятору, который решает задачу об изменении насыщенности резервуара нефтью, газом и водой в терминах ячеек. Таким образом, пользователю нет необходимости дополнительно изучать алгоритмы ввода данных в гидродинамический симулятор.

Для разработки программного модуля использованы языки программирования JavaScript, Python.

### 2.2. Входные данные (Solver Inputs)

К входным данным относятся:

$P_{head}$	координаты начала (устья) скважины в формате $(X, Y, Z)$ , где $(X, Y)$ – топографические координаты в метрах, а $(Z)$ – глубина в метрах
$P_{hill}$	пятка скважины в формате аналогичном устью
$P_{toe}$	носок скважины в формате аналогичном устью

Пользователь может задавать координаты  $(X, Y)$  скважины как с помощью формы для ввода (пример ввода: Рис. 1.2.4.1), так и с помощью клика по карте месторождения (в интерфейсе), после чего значения координат автоматически заносятся в форму.

### 2.3. Выходные данные (Solver Outputs)

К выходным данным относятся:

$I$	Индекс пересекаемой ячейки по оси $x$
$J$	Индекс пересекаемой ячейки по оси $y$

$K$	Индекс пересекаемой ячейки по оси $z$
$MD_{top}$	Глубина первой точки ячейки, пересекаемой траекторией
$MD_{bot}$	Глубина последней точки пересечения траектории и ячейки
$X_{top}$	Абсолютная координата первой точки пересечения ячейки и траектории
$Y_{top}$	Абсолютная координата первой точки пересечения ячейки и траектории
$Z_{top}$	Абсолютная координата первой точки пересечения ячейки и траектории
$X_{bot}$	Абсолютная координата последней точки пересечения ячейки и траектории
$Y_{bot}$	Абсолютная координата последней точки пересечения ячейки и траектории
$Z_{bot}$	Абсолютная координата последней точки пересечения ячейки и траектории

## 2.4. Обработка ошибок

В логике интерфейса предусмотрена защита от очевидно неверных действий человека при использовании ПО, что позволяет исключить некорректно вводимые данные: интерфейс указывает пользователю на незаполненные поля формы и не даёт сохранить изменения с некорректными значениями.

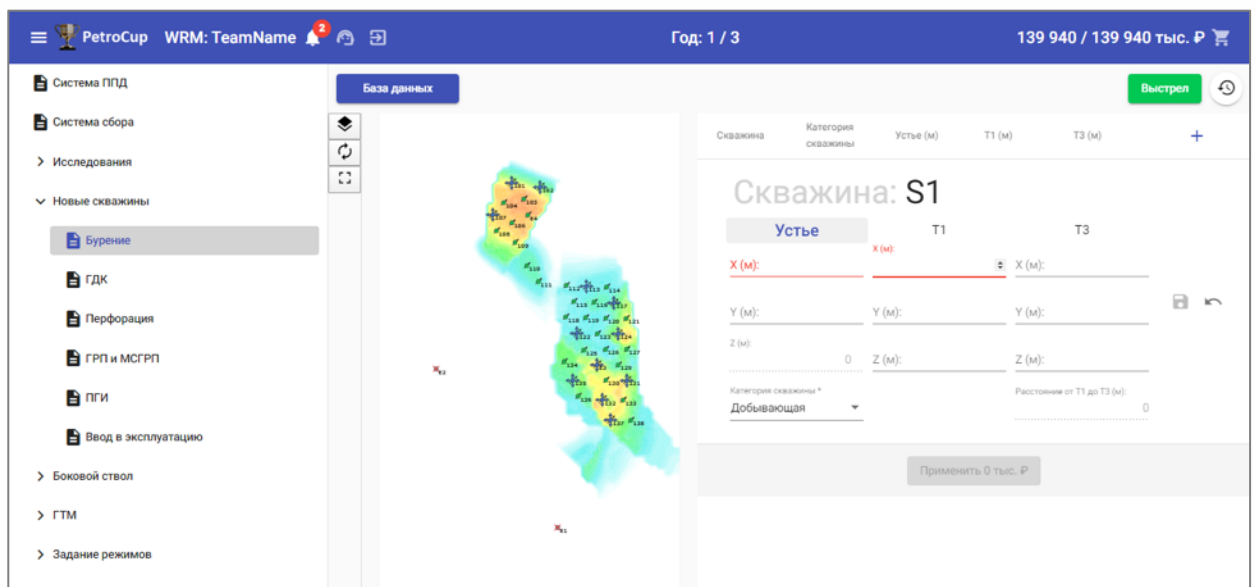


Рис. 1.2.4.1. Подсветка пропущенных при вводе полей

В случае возникновения непредвиденных ошибок пользователь получает уведомление:



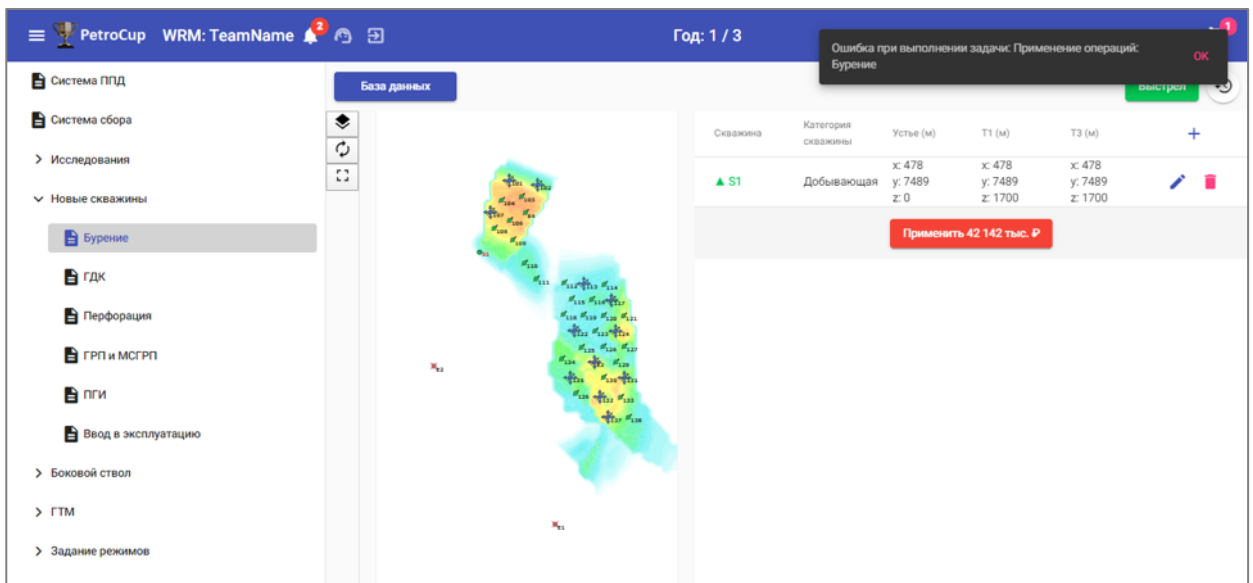


Рис. 1.2.4.2. Сообщение об ошибке, в случае возникновения ошибки при бурении новой скважины

Также пользователю доступна форма оперативной связи с технической поддержкой в случае возникновения ошибок / вопросов.

## 2.5. Структурная диаграмма программного модуля

Модуль имеет структуру, представленную на рисунке 1.2.5.1:

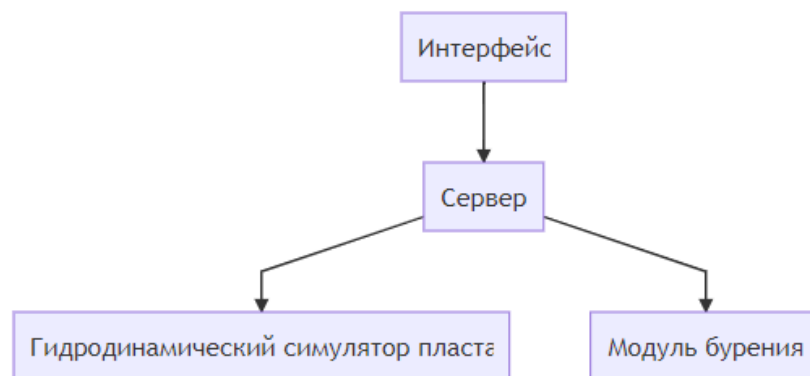


Рис. 1.2.5.1. Структурная диаграмма программного модуля

Структурные элементы:

- интерфейс - составляющая интерфейса, отвечающая за ввод параметров пользователем (подробнее о работе интерфейса в пункте 1.5);
- сервер - диспетчер для сообщений клиента;

- гидродинамический симулятор пласта - процедура, выдающая параметры сетки гидродинамического расчёта, чтобы выполнить построение траектории [6];
- модуль бурения - процедура, проводящая перевод данных, введённых пользователем в язык гидродинамического симулятора [6].

## 2.6. Разработка схемы работы программного модуля и ее описание

Модуль выполняет две задачи:

1. ввод параметров траектории новой скважины;
2. построение траектории новой скважины.

Блок-схема процедуры построения траектории скважины представлена на рисунке 1.2.6.1:

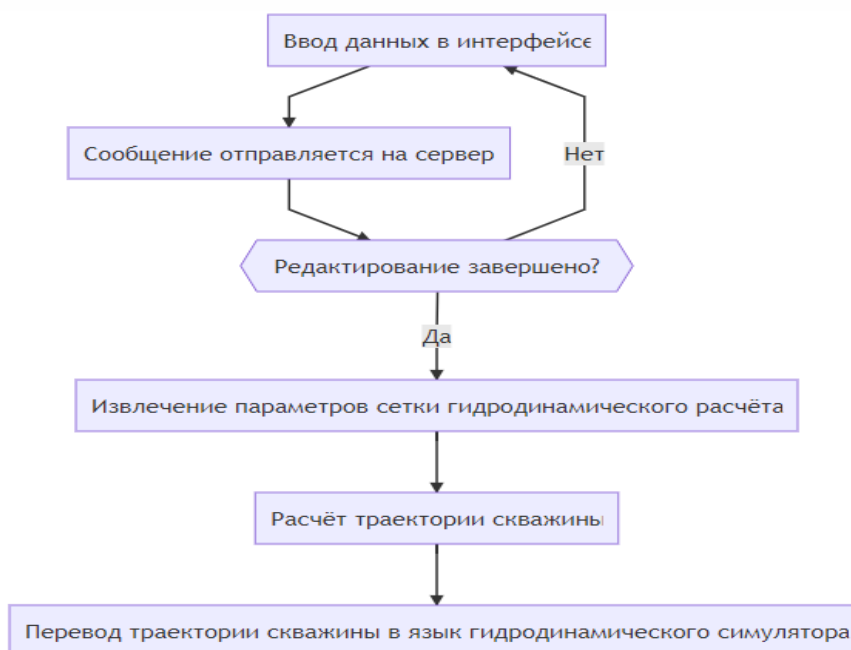


Рис. 1.2.6.1. Схема программного модуля "Бурение скважин"

В данном модуле данные, введённые пользователем в интерфейсе, поступают на сервер, который последовательно вызывает вспомогательные процедуры, указанные на схеме.

Полученные результаты используются в дальнейшем гидродинамическим симулятором при симуляции разработки для оценки эффективности пробуренной скважины.

## 2.7. Пользовательский интерфейс

Интерфейс для программного модуля "Бурение скважин" представлен на рисунке 1.2.7.1:



Рис. 1.2.7.1. Вид пользовательской формы

Описание пользовательской формы:

1. карта;
2. поле ввода координат устья скважины;
3. поле ввода координат входа скважины в пласт;
4. поле ввода координат забоя скважины;
5. кнопка добавления ещё одной скважины;
6. кнопка сохранения введенных координат;
7. кнопка отмены;
8. поле выбора типа скважины;
9. кнопка применения изменений.

## 2.8. Описание использованных операторов и функций

Выше представлен фрагмент модуля бурения скважин с построением отклика по бурению.

Описание используемых в коде конструкций:

- **if условие:** ... - условный оператор, выполняющий блок кода только при выполнении некоторого условия;

- **with** выражение **as** переменная: - конструкция, позволяющая автоматически освобождать ресурсы, занятые выражением при создании переменной после того, как блок кода закончит своё выполнение;
- **try:** блок **Кода1** **except:** тип **Ошибки** **as** переменная: блок **Кода2** **else:** блок **Кода3** - попытка выполнить блок **Кода1**. При ошибке типа **тип Ошибки** в блок **Кода1** вызывается блок **Кода2**. В случае работы без ошибки после блок **Кода1** выполняется блок **Кода3**;
- **переменная[ключ]** - получение значения из коллекции по заданному ключу.

Описание используемых в коде переменных:

- **data** - структура, содержащая необходимую для решения задачи информацию:
  - **oh\_model** - объект для моделирования датчиков открытого ствола по пробуренной скважине;
  - **lws\_data** - переменная, хранящая конструкцию новой скважины;
  - **store** - объект, ответственный за хранение данных;
  - **log\_plotter** - построитель графиков для пользователя;
  - **with\_img** - логическая переменная для указания необходимости сохранить результаты симуляции в виде изображения;
  - **assets** - набор неизменяемых параметров месторождения;
  - **field** - набор текущих временных параметров моделируемого месторождения.

Описание используемых в коде функций и методов:

- **write\_las** - сохранить результаты в текстовом формате las;
- **new\_open\_hole\_model\_from\_data** - создание нового объекта для моделирования отклика по бурению новой скважины;
- **store.save\_ON\_las**, **store.save\_ON\_zones**, **store.save\_ON\_sensors**, **store.save\_ON\_img** - методы для сохранения различных составляющих отклика по бурению скважины;
- **logging.error** - запись информации об ошибке.

### 3. МОДУЛЬ «ГИДРОРАЗРЫВ ПЛАСТА»

#### 3.1. Назначение

Главной задачей модуля "Гидроразрыв пласта" является обучить пользователей эффективно производить гидроразрыв, чтобы увеличить интенсивность нефтеотдачи за счет большего охвата продуктивной зоны, сделать рентабельной добычу из неудобных коллекторов с низкими фильтрационно-емкостными свойствами и при этом не получить заколонную, или затрубную циркуляцию (ЗКЦ).

Разработанный программный модуль позволит повысить продуктивность пласта путем увеличения эффективного радиуса дренирования скважины. В пластах с относительно низкой проницаемостью гидроразрыв – лучший способ повышения продуктивности. Также, гидроразрыв пласта помогает создавать канал притока в приствольной зоне нарушенной проницаемости. Нарушение проницаемости продуктивного пласта – важное для понимания понятие, поскольку тип и масштаб процесса разрыва проектируется именно с целью исправления этого нарушения. Если есть возможность создать проходящую сквозь зону повреждения трещину, заполненную проппантом, и привести падение давления до нормальной величины градиента гидродинамического давлений, то продуктивность скважины возрастет.

Для разработки программного модуля использованы языки программирования Python, JavaScript.

#### 3.2. Входные данные (Solver Inputs)

К входным данным относятся:

<b><math>md</math></b>	Глубина точки инициации гидроразрыва пласта относительно траектории скважины
<b><math>h</math></b>	Полувысота трещины
<b><math>w</math></b>	Полудлина трещины
<b><math>\alpha</math></b>	Азимутальный угол
<b><math>\theta</math></b>	Зенитный угол

### 3.3. Выходные данные (Solver Outputs)

К выходным данным относятся:

$q$	Мультифазный поток флюида по ГРП
-----	----------------------------------

### 3.4. Обработка ошибок

Интерфейс спроектирован так, чтобы исключить некорректные вводимые данные.

В случае возникновения непредвиденных ошибок пользователь получает уведомление:

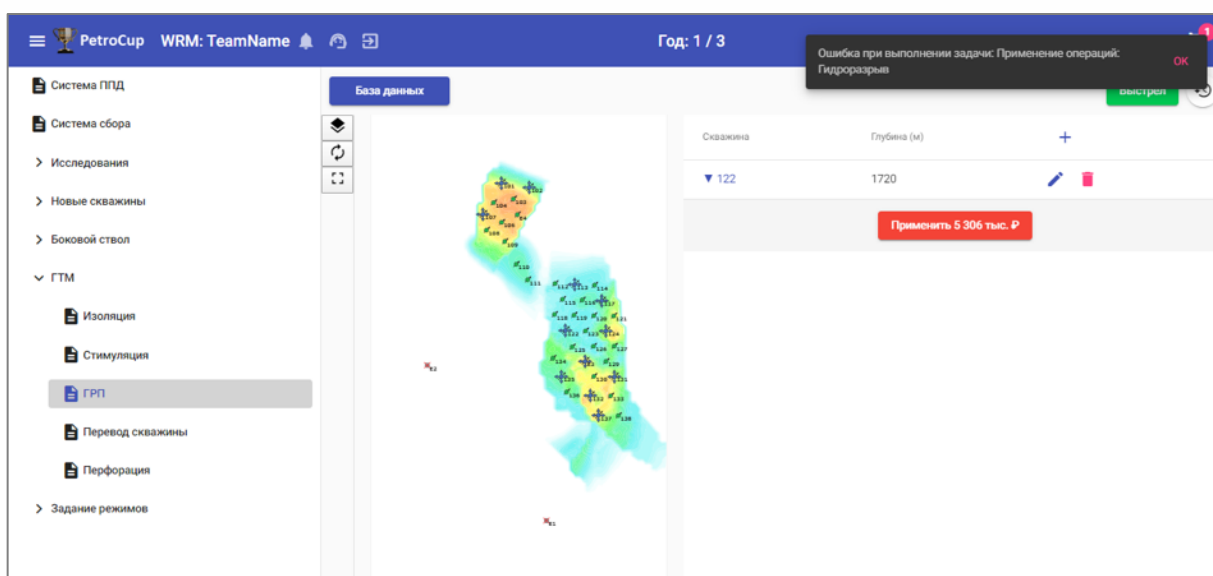


Рис. 1.3.4.1. Сообщение в случае возникновения непредвиденной ошибки при проведении ГРП

### 3.5. Структурная диаграмма программного модуля

Модуль имеет структуру, представленную на рисунке 1.3.5.1:

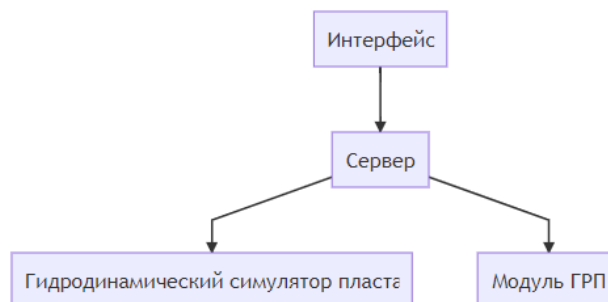


Рис. 1.3.5.1. Структурная диаграмма программного модуля

Структурные элементы:

- интерфейс - составляющая интерфейса, отвечающая за ввод параметров пользователем (подробнее о работе интерфейса в пункте 1.5);
- сервер - диспетчер для сообщений клиента;
- гидродинамический симулятор пласта - процедура, выдающая параметры сетки гидродинамического расчёта, чтобы выполнить построение траектории;
- модуль ГРП - процедура, проводящая перевод данных, введённых пользователем в язык гидродинамического симулятора.

### 3.6. Разработка схемы работы программного модуля и ее описание

Модуль выполняет две задачи:

1. редактирование параметров гидроразрыва пласта;
2. симуляция работы гидроразрыва пласта в гидродинамической модели.

Блок-схема процедуры моделирования ГРП представлена на рисунке 1.3.6.1:

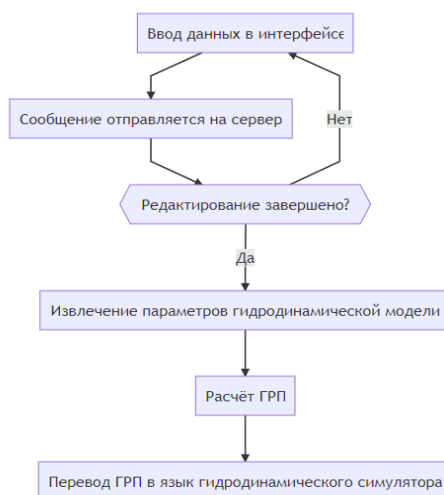


Рис. 1.3.6.1. Схема программного модуля Гидроразрыв пласта

В данном модуле данные, введённые пользователем в интерфейсе, поступают на сервер, который последовательно вызывает вспомогательные процедуры, указанные на схеме.

Полученные результаты используются в дальнейшем гидродинамическим симулятором при симуляции разработки для оценки эффективности выполненного гидроразрыва пласта.

### 3.7. Пользовательский интерфейс

Интерфейс для программного модуля "Гидроразрыв пласта" представлен на рис. 1.3.7.1.

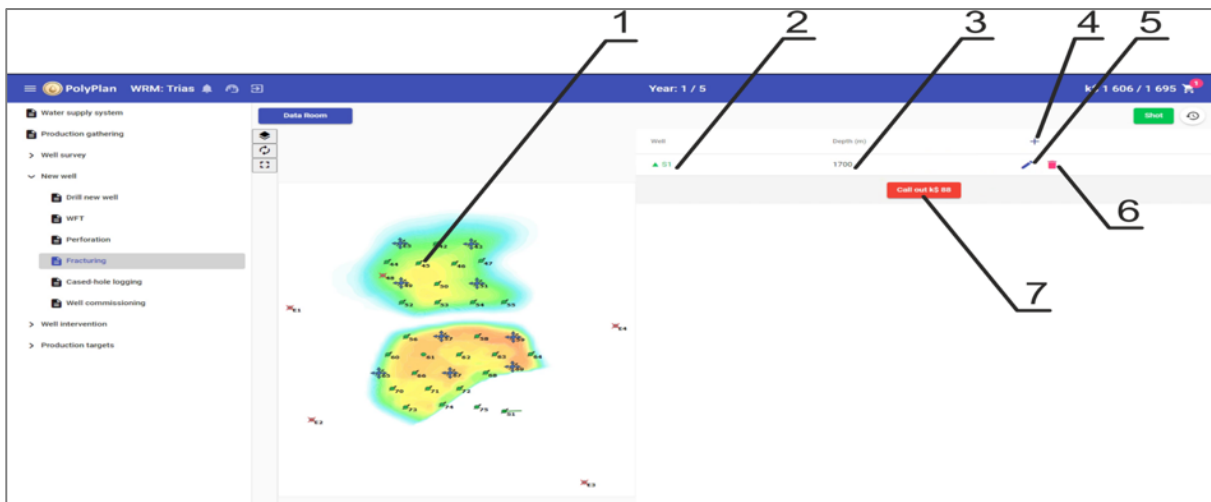


Рис. 1.3.7.1. Вид пользовательской формы

Описание пользовательской формы:

1. карта;
2. поле выбора скважины;
3. поле ввода глубины интервала ГРП;
4. кнопка добавления ещё одного интервала ГРП;
5. кнопка редактирования глубины интервала;
6. кнопка удаления интервала;
7. кнопка применения изменений.

### 3.8. Описание использованных операторов и функций

В коде выше производится расчёт притока по трещине ГРП.

Описание используемых в коде конструкций:

- **[выражение(значение) for значение in коллекция]** - быстрое создание новой коллекции по значениям другой коллекции с помощью вычисления некоторого выражения;
- **for значение in коллекция** - конструкция, позволяющая перебрать все значения в некоторой коллекции;
- **if условие: ...** - условный оператор, выполняющий блок кода только при выполнении некоторого условия.

Описание используемых в коде переменных:

- fractures - набор ГРП по скважине;
- dense\_plt\_data - приток флюида по скважине;
- mask - интервал притока по ГРП;



- `oil_rate` - поток нефти по скважине не из ГРП;
- `water_rate` - поток воды по скважине не из ГРП.

Описание используемых в коде функций и методов:

- `sum` - функция, вычисляющая сумму элементов;
- `get_data_by_cell` (индекс, ключ, `default=0`) - получение параметра по ключу и индексу ячейки;
- `logging.warning` - функция для записи предупреждения в программный лог;
- `calculate_fracture_flow` - функция расчёта потока по ГРП.

## 4. МОДУЛЬ «КОМПЛЕКС ГЕОФИЗИЧЕСКИХ ДАННЫХ»

### 4.1. Назначение

Главной задачей разрабатываемого модуля «Комплекс геофизических данных» является обучить пользователей интерпретировать сырые данные промыслово-геофизических исследований и следить за поддержанием пластового давления.

Разработанный программный модуль позволит участникам заказывать исследования, которые позволяют снять ряд существенных неопределенностей в районе скважины:

- скважинный профиль потока (СПП);
- резервуарный профиль потока (РПП);
- замер пластового давления.

Для разработки программного модуля использованы языки программирования JavaScript, Python.

### 4.2. Входные данные (Solver Inputs)

К входным данным относятся:

<b>well</b>	модель скважины для теста (траектория, конструкция)
<b>top, bottom</b>	интервал теста по глубине скважины
<b>q_z</b>	матричный приток по активным ячейкам без учета трещин ГРП
<b>wells</b>	названия скважин (набор скважин)
<b>units</b>	единица измерений
<b>path</b>	путь до файла в формате Eclipse 100

### 4.3. Выходные данные

К выходным данным относятся:

<b>q_zi</b>	поток жидкости в пласте вдоль скважины
<b>result</b>	значения пластовых давлений по скважинам

#### 4.4. Обработка ошибок

Интерфейс спроектирован так, чтобы исключить некорректно вводимые данные.



Рис. 1.4.4.1. Выбор имени скважины для исследования



Рис. 1.4.4.2. Выбор типа исследования

В случае возникновения непредвиденных ошибок пользователь получает уведомление:

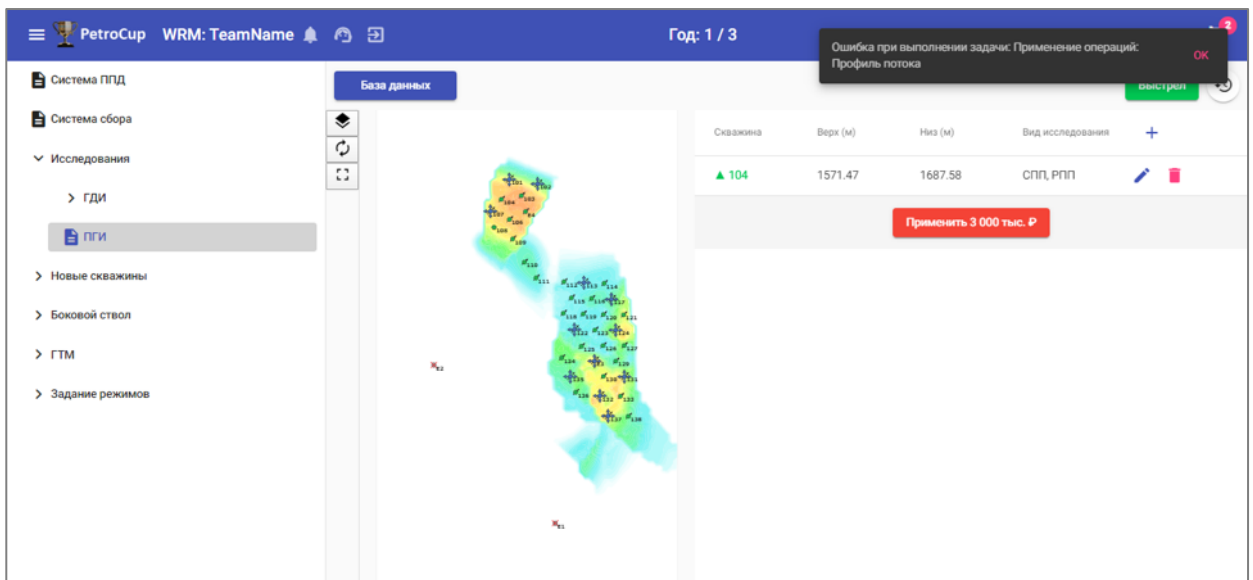


Рис. 1.4.4.3. Сообщение в случае возникновения ошибки в модуле "Комплекс геофизических данных"

Также пользователю доступна форма оперативной связи с тех. поддержкой в случае непредвиденных ошибок.

#### 4.5. Структурная диаграмма программного модуля

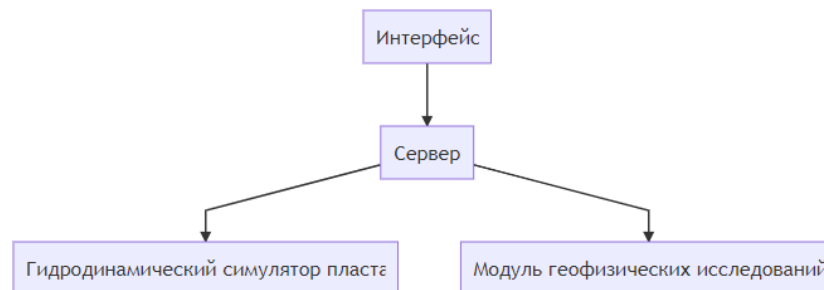


Рис. 1.4.5.1. Структурная диаграмма программного модуля

Структурные элементы:

- интерфейс - составляющая интерфейса, отвечающая за ввод параметров пользователем (подробнее о работе интерфейса в пункте 1.5);
- сервер - диспетчер для сообщений клиента;
- гидродинамический симулятор пласта - процедура, выдающая параметры сетки гидродинамического расчёта, чтобы выполнить построение траектории;

- модуль геофизических исследований - процедура, проводящая перевод данных, введённых пользователем в язык симулятора генерации комплекса открытого ствола скважины.

#### 4.6. Разработка схемы работы программного модуля и ее описание

Блок-схема процедуры моделирования комплекса геофизических исследований представлена на рисунке 1.4.6.1:



Рис. 1.4.6.1. Схема программного модуля геофизических данных

В данном модуле данные, введённые пользователем в интерфейс, поступают на сервер, который последовательно вызывает вспомогательные процедуры, указанные на схеме. После этого результаты симуляции становятся доступными для просмотра в интерфейсе.

#### 4.7. Пользовательский интерфейс

Интерфейс для программного модуля "Комплекс геофизических данных" представлен на рисунке 1.4.7.1:

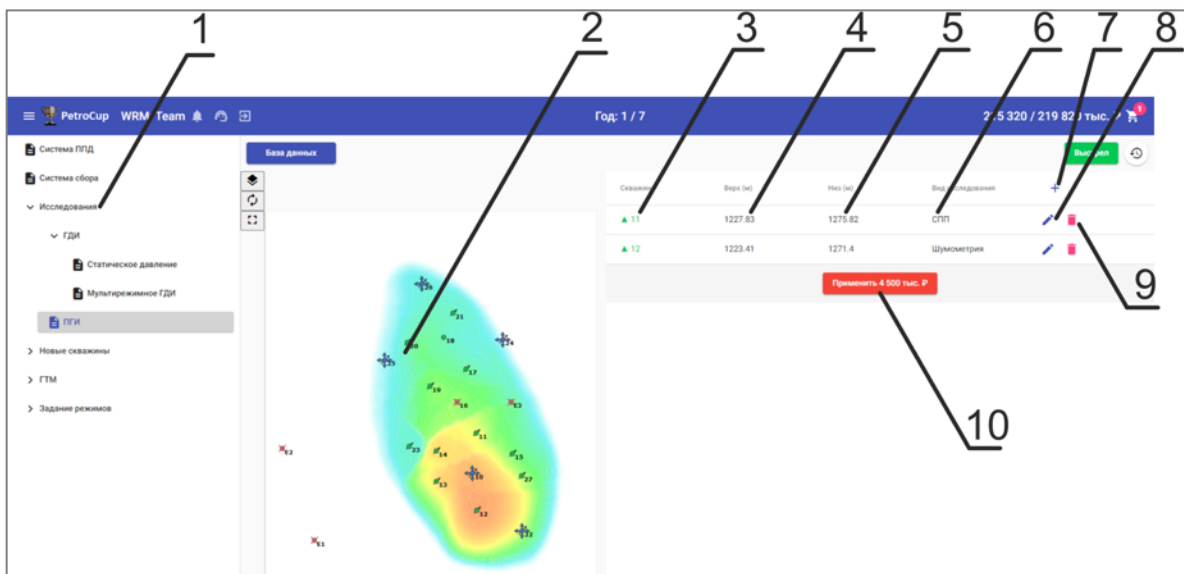


Рис. 1.4.7.1. Вид пользовательской формы

#### Описание пользовательской формы:

1. поле выбора типа исследований;
2. карта;
3. поле выбора скважины;
4. поле ввода интервала исследования;
5. поле низа интервала исследования;
6. поле выбора вида исследования;
7. кнопка добавления ещё одного исследования;
8. кнопка редактирования глубины интервала;
9. кнопка удаления интервала;
10. кнопка применения изменений.

#### 4.9. Описание использованных операторов и функций

Выше приведён фрагмент модуля расчёта скважинного профиля притока.

Описание используемых в коде конструкций:

- ... **or** ... - возвращает левое значение, если его логическое значение - правда. Иначе возвращает правое значение;
- **for значение in коллекция** - конструкция, позволяющая перебрать все значения в некоторой коллекции;
- **if условие: ... else: ...** - условный оператор, выполняющий блок кода только при выполнении некоторого условия. Иначе выполняет второй блок кода.

Описание используемых в коде переменных:

- well - набор параметров скважины;
- bfp - параметры геофизического исследования СПП;
- plt\_data - предобработанные данные потока по скважине;
- store - объект, ответственный за хранение данных;
- ws - объект, хранящий данные конструкции скважины;
- plt - модуль, содержащий функции для моделирования потоков для разных типов скважин;
- qzi - результат моделирования геофизических датчиков в виде массивов чисел;
- units - используемые единицы измерения.

Описание используемых в коде функций и методов:

- preprocess\_plt\_data - предобработка данных гидродинамического симулятора для геофизического модуля;
- plt.normalize\_columns - функция переименования имен по принятому соглашению;
- plt.make\_bfp\_producer, plt.make\_bfp\_injector - функции для моделирования геофизических данных для разных типов скважин;
- log\_plotter.plot\_bfp\_producer, log\_plotter.plot\_bfp\_injector - функции построения геофизических данных в виде изображений;
- store.save\_bfp\_simulation, store.save\_bfp\_img, store.save\_user\_bfp\_simulation, store.save\_initial\_bfp\_data - функции для сохранения геофизических данных в различных форматах.

## 5. МОДУЛЬ «ИНТЕРФЕЙСНАЯ ЧАСТЬ «PETROCUP WEB-INTERFACE»»

### 5.1. Назначение

Главной задачей разрабатываемого модуля «Интерфейсная часть «PETROCUP Web-Interface»» является возможность манипулировать программным и аппаратным обеспечением и управлять им с помощью пользовательского интерфейса. Пользовательский интерфейс спроектирован таким образом, чтобы обеспечить понимание пользователем программного обеспечения.

Разработанный программный модуль позволит обеспечить фундаментальную платформу для взаимодействия всех дополнительных модулей в одном месте.

Для разработки программного модуля использован язык программирования JavaScript и использованы фреймворки Angular, Apollo.

### 5.2. Проверка ввода пользователя

При выполнении программного модуля была предусмотрена и реализована проверка вводимых пользователем данных на корректность. Также были предусмотрены варианты, при которых позволяющие пользователю не забыть или корректно завершить редактирование форм.

#### 1. Предупреждение о потенциальных невыполненных действиях

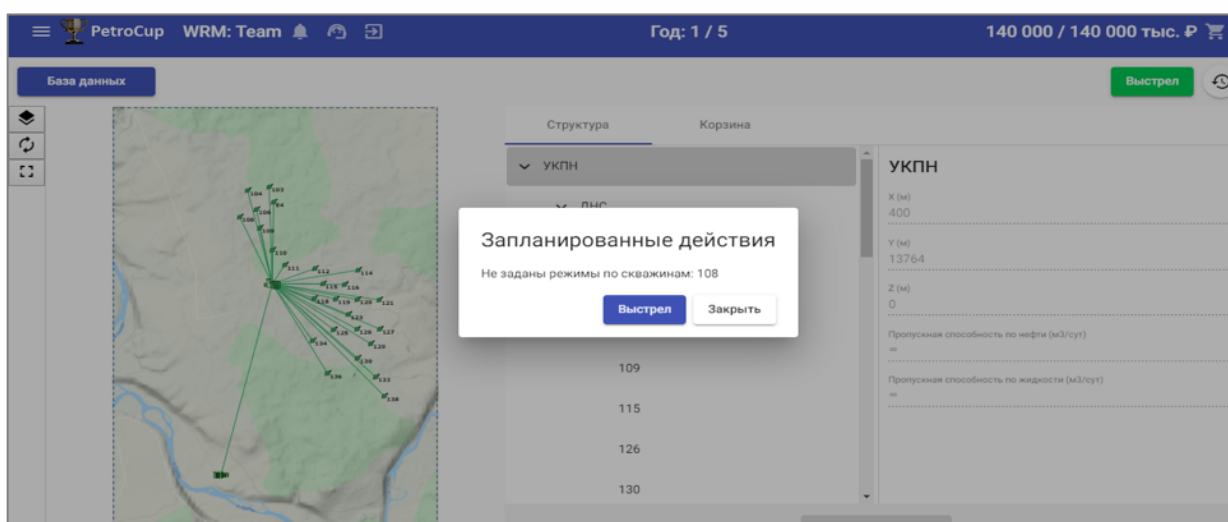


Рис. 1.5.2.1. Предупреждение о не заданных режимах при запуске расчёта

#### 2. Подсветка пропущенных полей при вводе



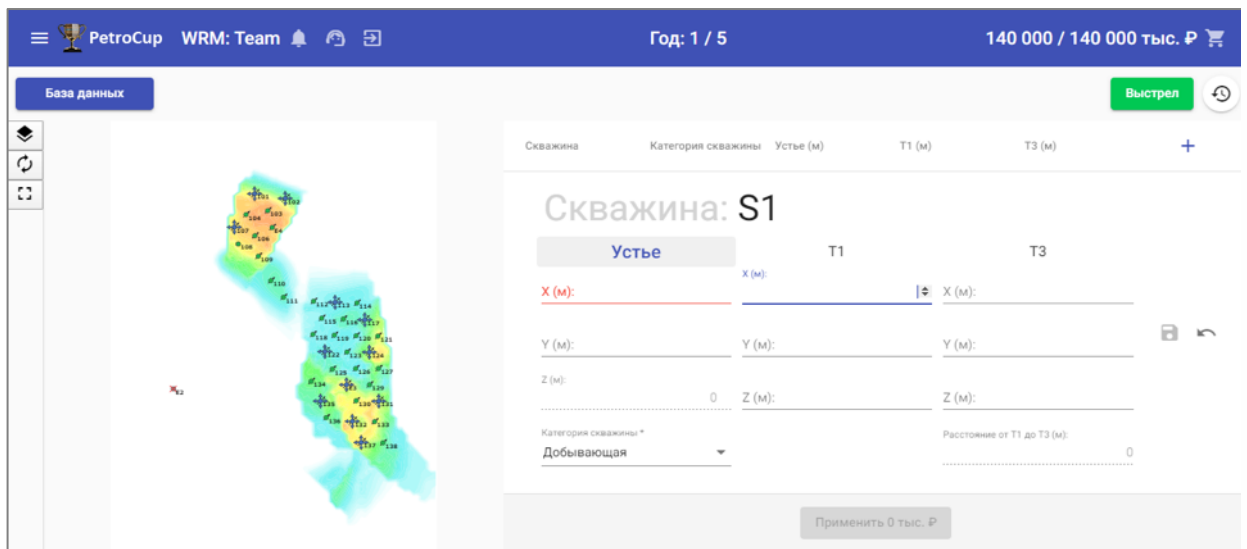


Рис. 1.5.2.2. Подсветка полей пропущенного поля ввода при задании координат для бурения скважины

### 3. Предупреждение о незавершённом редактировании формы

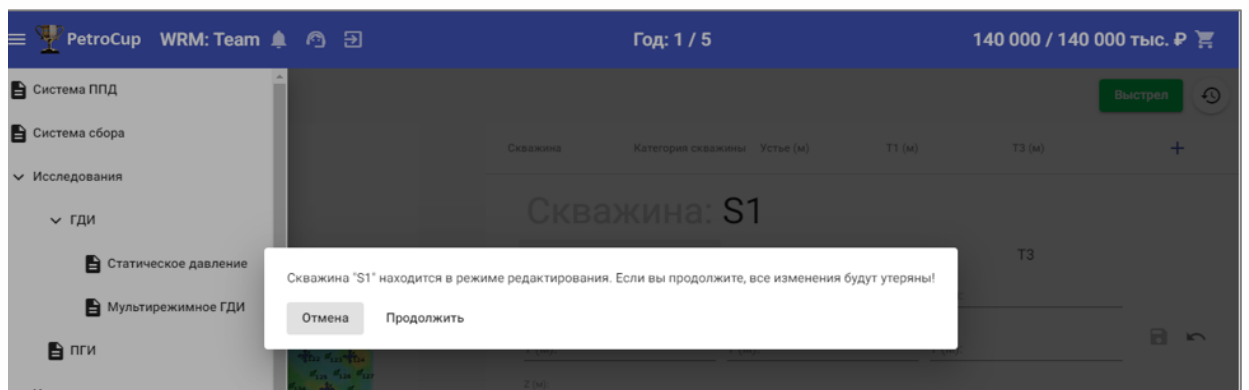


Рис. 1.5.2.3. Предупреждение о незавершённом редактировании формы бурения при переходе на другую страницу

### 4. Ограничение интервалов корректных значений



Рис. 1.5.2.4. Ограничение сверху для интервала времени исследования скважины

## 5. Обработка ошибок соединения

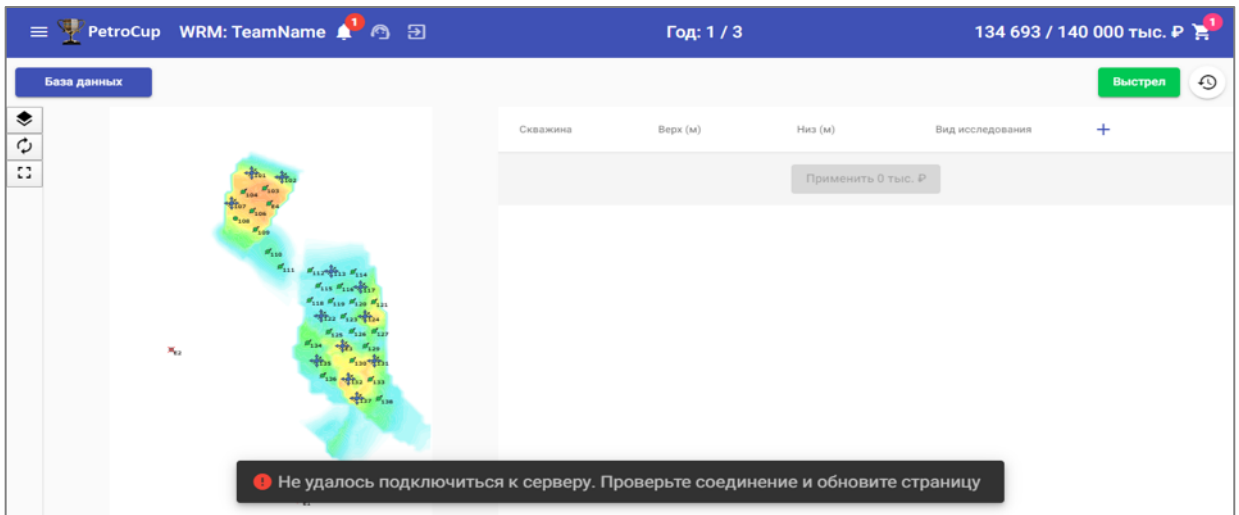


Рис. 1.5.2.5. Уведомление о потере соединения с сервером

## 1.5.3. Структурная диаграмма программного модуля

Модуль имеет структуру, представленную на рисунке 1.5.5.1:

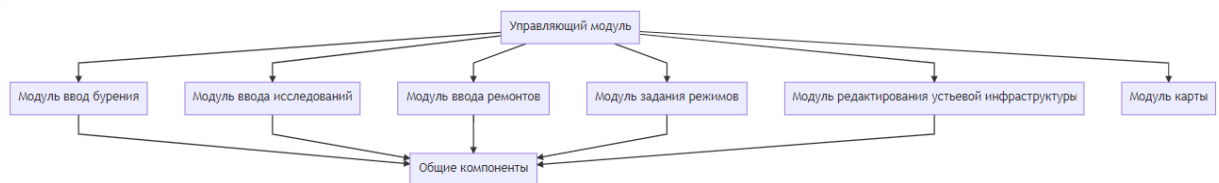


Рис. 1.5.3.1. Структурная диаграмма программного модуля

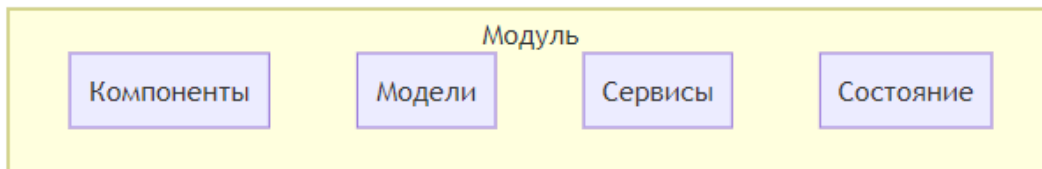


Рис. 1.5.3.2 Схема модулей ввода

Управляющий модуль отвечает за связь с серверной частью по транспортным протоколам HTTP и WebSocket и обновление состояния внутри модулей по мере поступления изменений с сервера. Такой подход позволяет обеспечить реактивность интерфейса - состояние форм отображается одинаковым на нескольких клиентах одновременно (например, из разных вкладок браузера или при доступе нескольких людей из разных браузеров).

Модули ввода отвечают за отображение, ввод и проверку значений для пользователя по заранее определённым правилам.

#### 5.4. Пользовательский интерфейс

Интерфейс для программного модуля «Интерфейсная часть «PETROCUP Web-Interface»» представлен на рисунке 1.5.7.1:

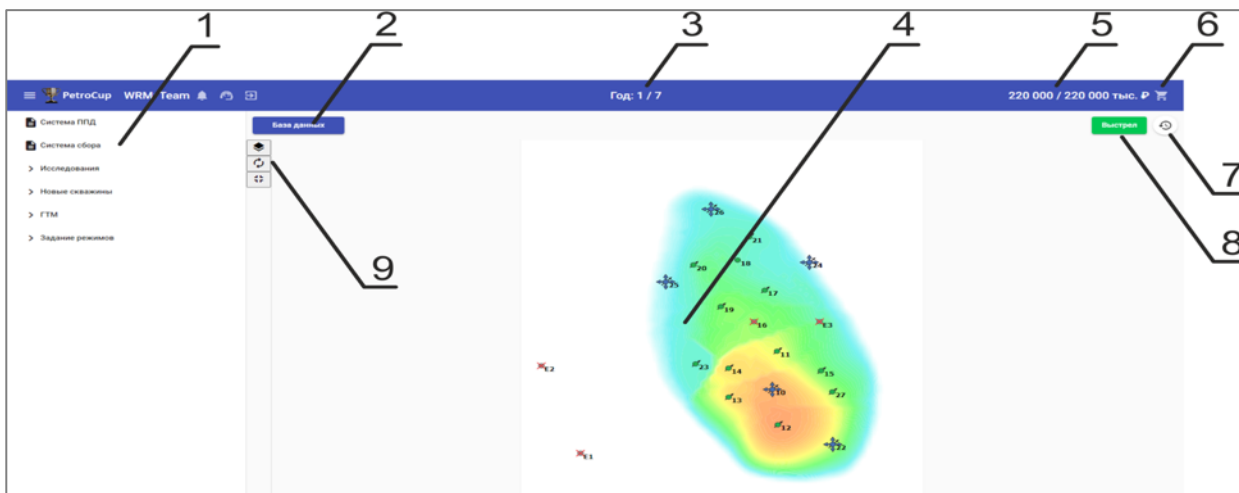


Рис. 1.5.4.1. Вид пользовательской формы

Описание пользовательской формы:



1. поле выбора модуля;
2. кнопка открытия DATAROOM;
3. номер текущего выстрела;
4. карта;
5. текущий бюджета;
6. кнопка просмотра заказанных операций;
7. кнопка перезапуска;
8. кнопка выстрела;
9. кнопки управления картой.

## 5.5. Описание использованных операторов и функций

Выше приведён фрагмент front-end части отвечающий за установку и поддержание соединение с сервером по протоколу WebSocket (ws).

Описание используемых в коде конструкций:

- `@Injectable()` - маркер показывающий фреймворку Angular, что данный объект может быть переиспользован в других компонентах;
- `private` - указатель на то что метод или данные будут доступны только из этого класса;
- `async` - указатель на то что функция или метод имеет возможность коммуницировать с серверами или делать иного рода асинхронные запросы;
- `pipe` - так называемая "труба", это метод из библиотеки rxjs, позволяющий использовать данные в режиме онлайн.

Описание используемых в коде переменных:

- `event` - в эту переменную записываются данные, которые приходят от сервера для дальнейшего применения;
- `state$` - в эту переменную записываются данные об изменениях состояния соединения;
- `reconnecting$` - в эту переменную записываются данные - события о том что веб-страница пытается опять наладить соединение;
- `noconnection$` - в эту переменную записываются данные об отсутствии соединения;
- `attempts` - при повторных попытках переподключения, здесь отмечается запись увеличивая значение на единицу;
- `maxAttempts` - максимальное значение количества переподключений.

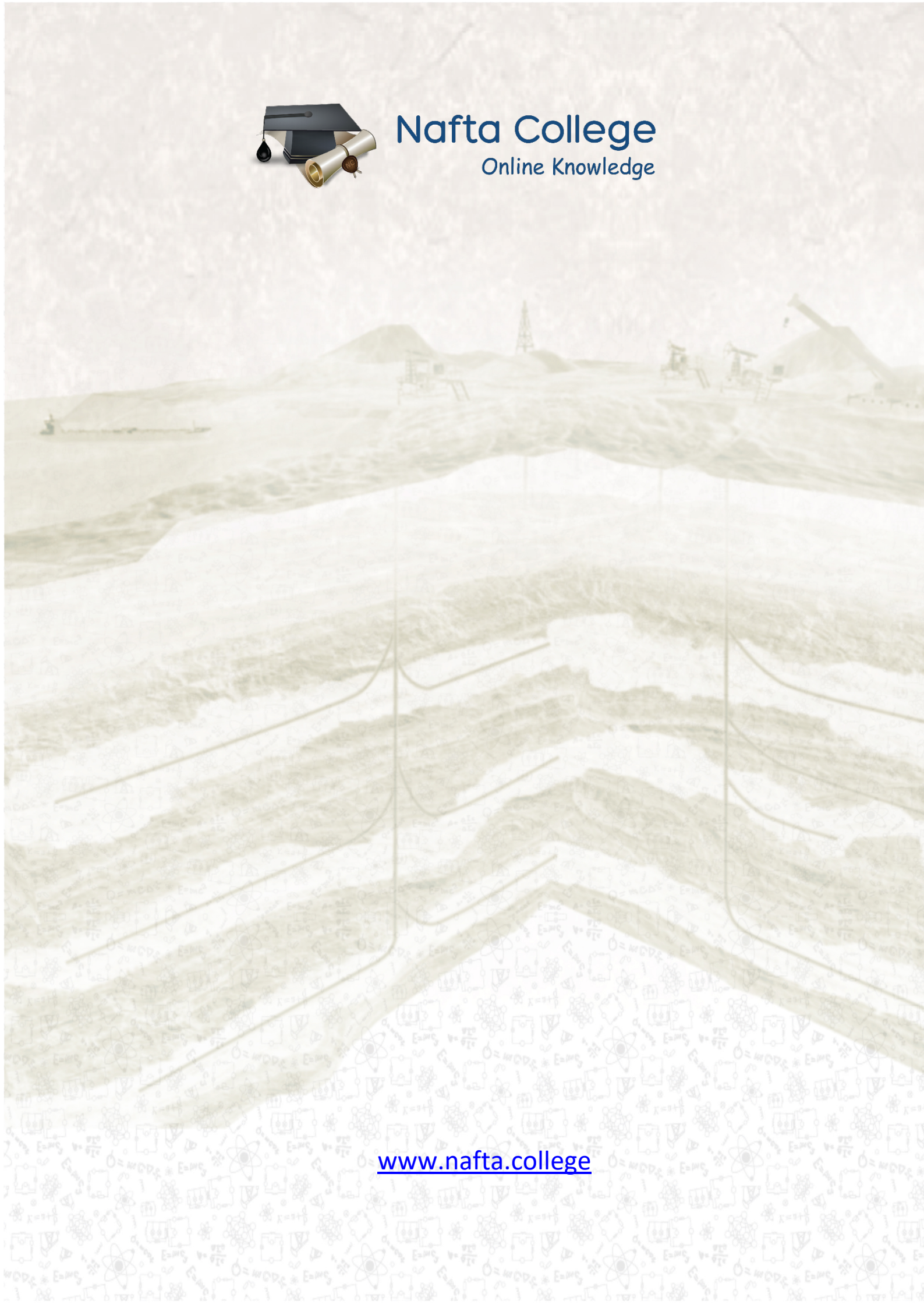
Описание используемых в коде функций и методов:

- `async connect()` - метод который при вызове начинает устанавливать соединение;
- `private connected()` - метод вызываемый при установке соединения, который обнуляет количество попыток подключения и устанавливает статус соединения;
- `private reconnect()` - метод который вызывается для повторной попытки подключения, через некоторый промежуток времени задержки.



# Nafta College

Online Knowledge



[www.nafta.college](http://www.nafta.college)