



ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ

«ПРОТЕЙ Технологии»

Инструкция по эксплуатации «PromUC.Портал»

Оглавление

1	Роли пользователей	3
2	Существующие разделы программы	4
3	Авторизация	5
4	Общая навигация.....	8
4.1	Меню.....	8
4.2	Область открытой вкладки	9
4.3	Выход из учетной записи.....	9
5	Меню.....	10
5.1	Уставки	10
5.2	Процесс SOP1	11
5.3	Экземпляры процессов.....	13
5.4	RuleEngine	14
6	Полный цикл инцидента	20

Технолог (администратор):

Это роль администратора в программе PromUC.Портал, в которой он настраивает работу системы, прописывает необходимые инструкции.

Функции раздела UI-технолога:

- Отслеживание работы пользователей;
- Доступ к подразделу «Уставки»;
- Доступ к подразделу «Экземпляры процессов»;
- Доступ и изменение (редактирование) инструкций в подразделе «SOP1»;
- Доступ и изменение (редактирование) цепочек в подразделе «RuleEngine».

Пользователя:

Это роль в программе PromUC.Портал, с помощью которой пользователь взаимодействует с настроенными администратором инцидентами.

Функции раздела UI-пользователя:

- Доступ к подразделу «Задачи» в UI-пользователя;
- В подразделе «Задачи»: отслеживание и подтверждение приходящих к пользователю инцидентов.

2

Существующие разделы программы

1) PromUC Control (интерфейс администратора) – это раздел в программе PromUC.Портал, в котором администратор с ролью UI-Технолога вносит изменения в инструкциях, процессе SOP1 и в цепочках RuleEngine. Также он предназначен для, чтобы отслеживать действия пользователей с ролью UI-пользователя.

Подразделы PromUC Control:

- «Уставки» - это заданное значение переменной, которое система стремится поддерживать.
- «Процесс SOP1» – процесс обработки инцидентов.
- «Экземпляры процессов» – таблица запущенных инцидентов.
- «Rule Engine» – шина данных.

2) ProScales (Интерфейс пользователя) – это раздел в программе PromUC.Портал, который выводит информацию об инцидентах для пользователей с ролью UI-пользователя.

Подразделы интерфейса пользователя:

- «Задачи» – в этом подразделе появляется информация об инцидентах и с ними может взаимодействовать пользователь.

3

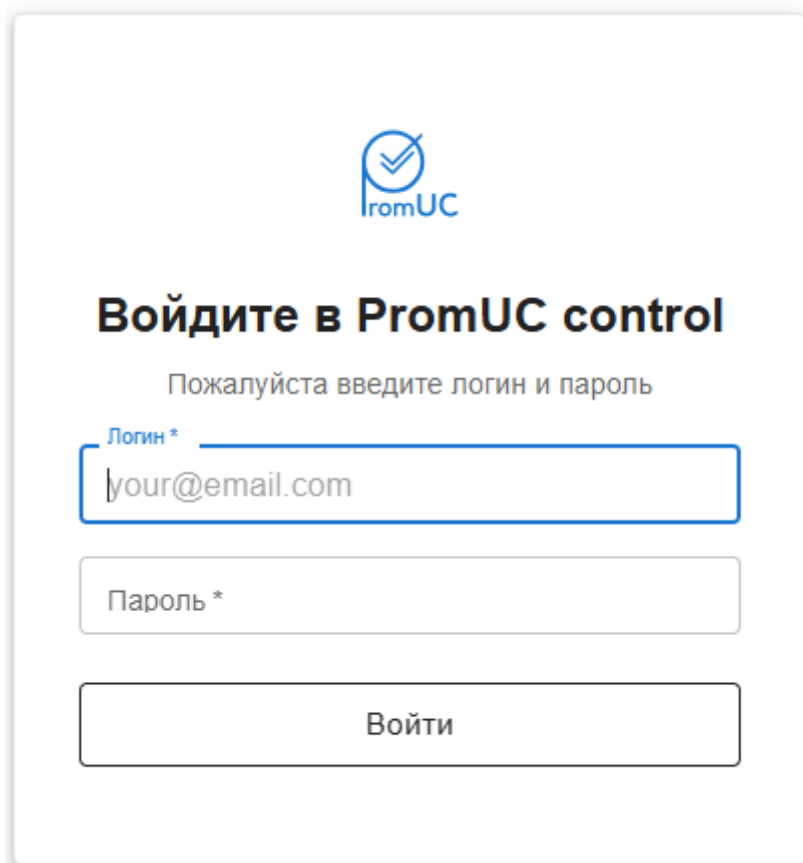
Авторизация

Что бы работать в системе обязательно необходимо пройти авторизацию. Для демонстрационной версии были заготовлены логины и пароли для определенных ролей.

UI-Технолога: логин (admin@protei.tl) и пароль (admin)

Каждая из ролей открывает свой доступ:

1.1 При входе на PromUC Control появляется окно для ввода логина и пароля для роли UI-Технолога



The image shows a login window for PromUC control. At the top is the PromUC logo, which consists of a blue circle with a checkmark and the text 'PromUC' below it. Below the logo is the title 'Войдите в PromUC control' in bold black text. Underneath the title is the instruction 'Пожалуйста введите логин и пароль' in a smaller, lighter font. There are two input fields: the first is labeled 'Логин *' in blue text and contains the placeholder 'your@email.com'; the second is labeled 'Пароль *' in gray text and is empty. Below the input fields is a large rectangular button with the text 'Войти' centered on it.

Если ввести неверный логин и пароль, то появится уведомление:



Войдите в PromUC control

Пожалуйста введите логин и пароль



Неверные логин или пароль

Логин *

admin@gmail.com

Пароль *

.....

Войти

После введения корректных данных и нажатии кнопки "Войти" вам откроется интерфейс Технолога.

Меню

 Уставки

 Процесс SOP1

 Экземпляры процессов

 RuleEngine

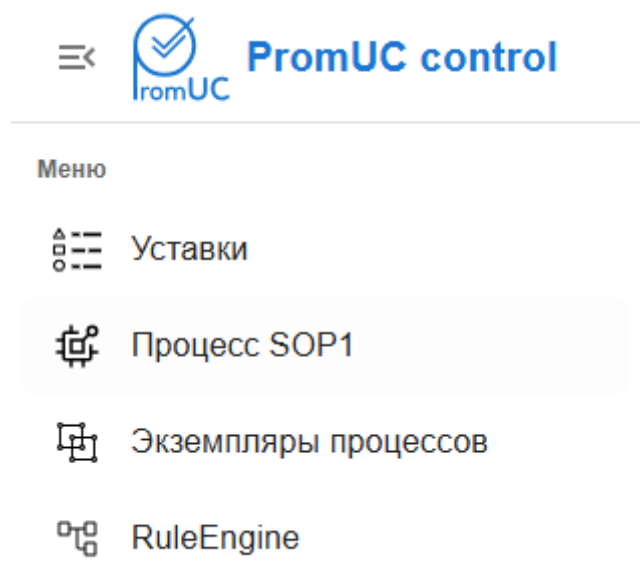
PromUC control

Добро пожаловать

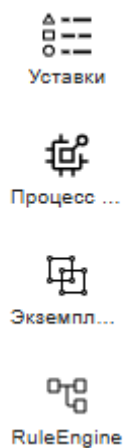
4.1 Меню

Меню находится в левой части экрана. Оно может быть в двух состояниях:

1. Развернутое



2. Свернутое

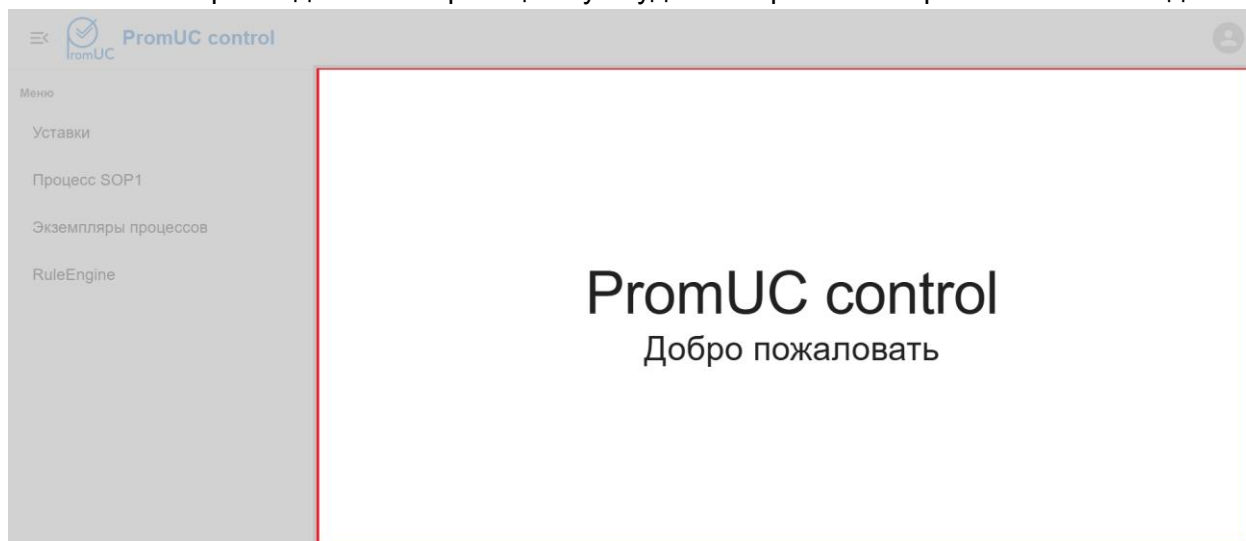


Вне зависимости от состояния меню вы можете легко открывать любой из его пунктов. Изменить состояние меню можно нажав:

 что бы свернуть меню, или  чтобы развернуть его.

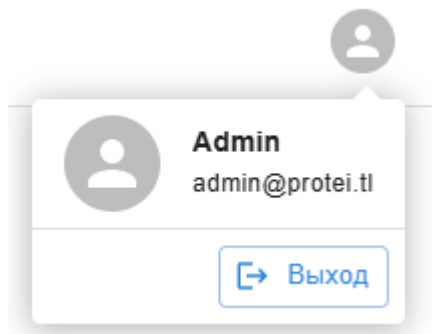
4.2 Область открытой вкладки

Основная часть экрана - именно в ней отражается содержимое первых трех пунктов меню. После прохождения авторизации тут будет отображаться приветственная надпись.



4.3 Выход из учетной записи

Для выхода из учетной записи нажмите на аватар технолога в верхнем правом углу



И нажмите на кнопку Выход. После этого на экране появиться окно входа в систему.

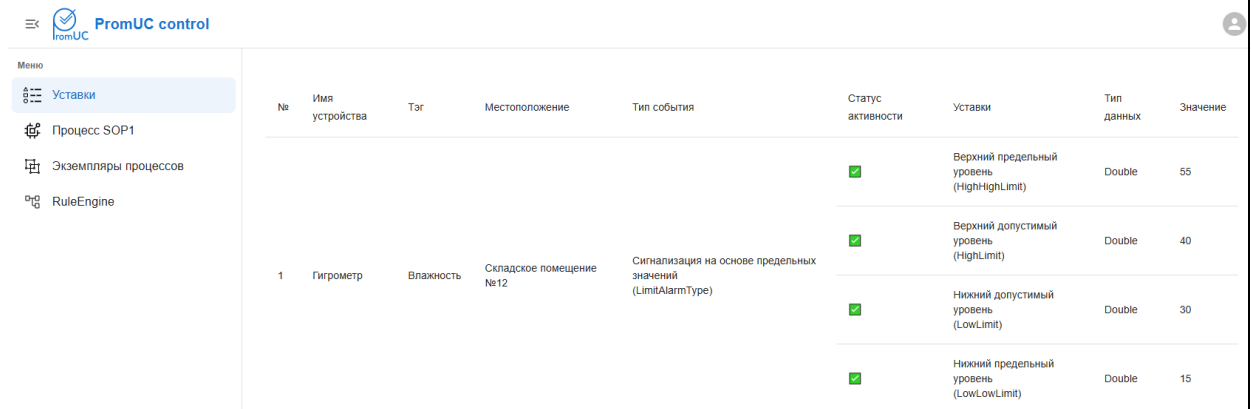
5

Меню

5.1 Уставки

Уставки — это заданное значение переменной, которое система стремится поддерживать. Это значение может быть установлено пользователем или другой системой и используется для управления процессом или оборудованием. Именно на основе сравнения уставок и приходящих данных система может понять, что что-то случилось.

В системе уже заведены уставки по уровню влажности в помещении:



№	Имя устройства	Тэг	Местоположение	Тип события	Статус активности	Уставки	Тип данных	Значение
1	Гигрометр	Влажность	Складское помещение №12	Сигнализация на основе предельных значений (LimitAlarmType)	☒	Верхний предельный уровень (HighHighLimit)	Double	55
					☒	Верхний допустимый уровень (HighLimit)	Double	40
					☒	Нижний допустимый уровень (LowLimit)	Double	30
					☒	Нижний предельный уровень (LowLowLimit)	Double	15

Расшифровка колонок в таблице:

№ - порядковый номер

Имя устройства - отображает к какому именно устройству относятся введенные уставки

Тэг - упрощает поиск и объединение уставок по типам

Местоположение - где расположено устройство

Тип события - тип по протоколу OPC UA. Например, у LimitAlarmType задается 4 граничных значения

Статус активности - Включена ли данная уставка, и сравнивает ли система с ней приходящие значения

Уставки - Наименование конкретной уставки
Промежуток между HighLimit и LowLimit считается нормой.

- Верхний предельный уровень (HighHighLimit)

Если значение приходящих данных выше верхнего предельного уровня - то срабатывает тревога

- Верхний допустимый уровень (HighLimit)

Если значение приходящих данных выше верхнего допустимого уровня - то срабатывает уведомление

- Нижний допустимый уровень (LowLimit)

Если значение приходящих данных ниже нижнего допустимого уровня - то срабатывает уведомление

- Нижний предельный уровень (LowLowLimit)

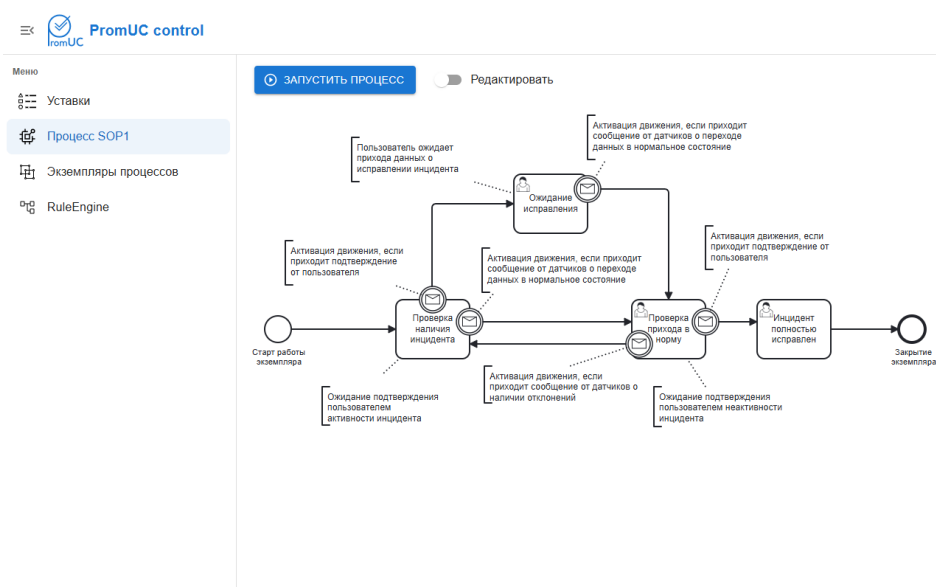
Если значение приходящих данных ниже нижнего предельного уровня - то срабатывает тревога

Тип данных - Double - какой тип данных сравнивается и вносится

Значение - конкретное значение уставки, с которым проводится сравнение.

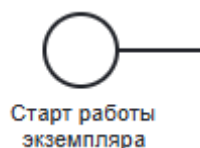
5.2 Процесс SOP1

Здесь отражается заранее настроенный процесс по обработке инцидента, возникающего в случае тревоги - модель процесса. При этом система берет модель процесса, и на её основе запускает отдельный экземпляр процесса - таким образом система может параллельно работать с несколькими инцидентами, по отдельности отслеживая и меняя состояние каждого отдельно.



Пояснения к BPMN диаграмме:

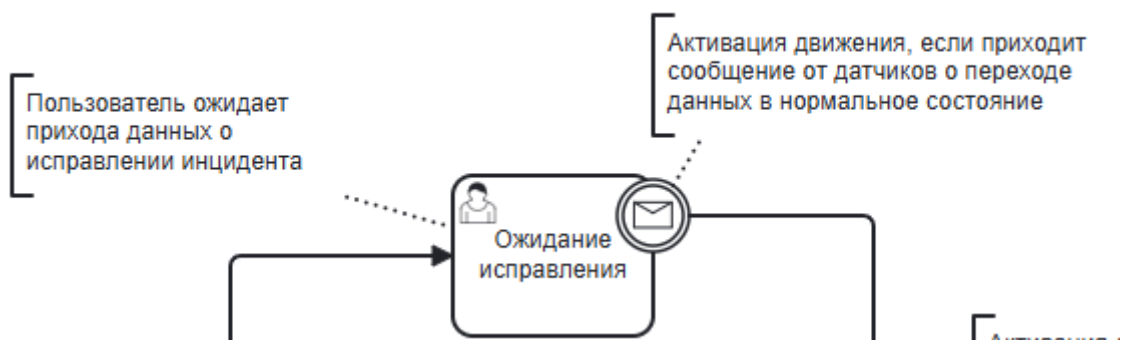
Обязательно в каждой модели есть стартовый блок - именно с него начинается всё движение



и блок окончания - когда закрывается экземпляр, и инцидент становится полностью отработанным.



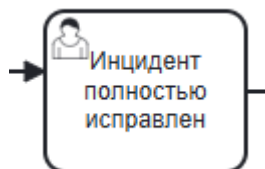
Стрелки показывают направление движения по модели от блока к блоку. Пунктирами к каждому блоку привязаны комментарии:



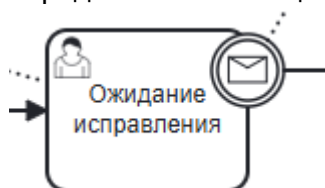
Слева комментарий к блоку (Ожидание исправления), а справа комментарий к переходу в следующий блок. Наименования блоков задач кратко сообщают о текущем состоянии, это очень поможет, когда мы запустим процесс и будем смотреть в экземпляры процесса.

Есть блоки с задачами – они выполняют на Backend ранее запрограммированные действия в системе. В данной модели есть два типа таких блоков:

1. Блок выполняет функции на стороне Backend и сразу идет в следующий блок



2. Блок выполняет функции на стороне Backend и ждет определенное сообщение, для перехода в следующий блок



Доступны две кнопки:



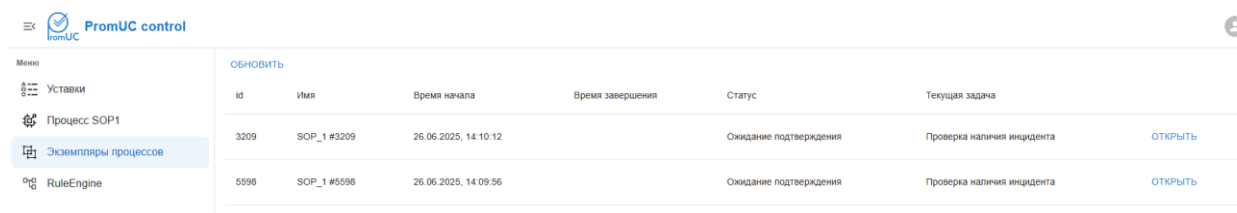
1. Запустить процесс
2. режим Редактирования

Как пользоваться этими кнопками описано в пункте 4 данного руководства.

5.3 Экземпляры процессов

При переходе в этот пункт открывается таблица запущенных экземпляров. Если процесс еще ни разу не запускался, то таблица будет пустой.

Кнопка **ОБНОВИТЬ** предназначена для обновления информации на странице Экземпляры процессов.



ОБНОВИТЬ					
id	Имя	Время начала	Время завершения	Статус	Текущая задача
3209	SOP_1 #3209	26.06.2025, 14:10:12		Ожидание подтверждения	Проверка наличия инцидента
5598	SOP_1 #5598	26.06.2025, 14:09:56		Ожидание подтверждения	Проверка наличия инцидента

Пояснение к столбцам таблицы:

id – уникальный идентификатор события, именно по нему можно найти инцидент во всей системе. Не может быть два инцидента с одинаковым id.

Имя – имя экземпляра состоит из названия модели процесса, и после # пишется id

Время начала – фиксируется дата и время (до секунды), когда экземпляр процесса был запущен

Время завершения - фиксируется дата и время (до секунды), когда экземпляр процесса был завершен. Будет пустым, пока экземпляр находится в процессе выполнения

Статус – выводит статус текущего экземпляра

Текущая задача – выводит, какая текущая задача выполняется

Кнопка **ОТКРЫТЬ** – позволяет провалиться в конкретный экземпляр процесса.

В открытом определенном процессе мы видим информацию:

Id – номер экземпляра

Статус – текущий статус экземпляра

Текущие инструкции – инструкции, которые были отправлены пользователю

Кнопка **ОБНОВИТЬ** – для обновления информации и обновления текущей задачи

Ниже расположена модель процесса, на основе которой был запущен экземпляр. На ней выделена блок-задача, на которой сейчас находится процесс – Зеленый контур и желтая заливка блока.

Id: 5598

Статус: Выполняется

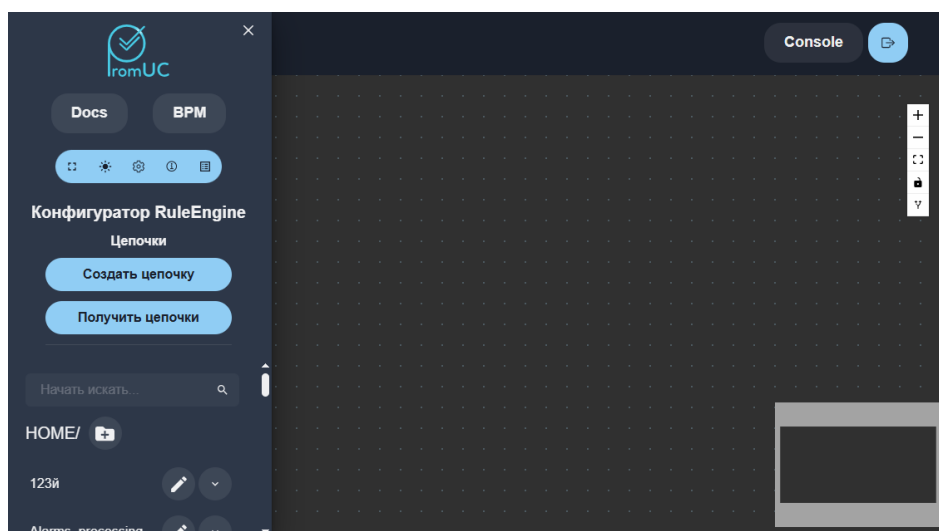
Текущие инструкции: Перейдите в окно Весовой, и визуально (по датчикам и / или камерам) и подтвердите наличие данной проблемы

[ОБНОВИТЬ](#)

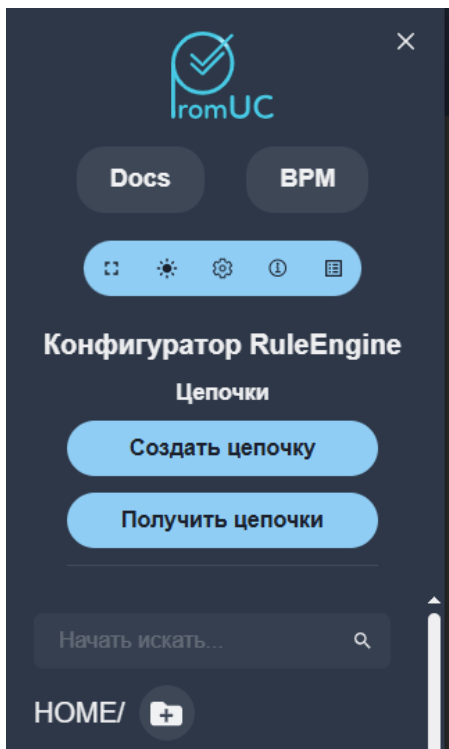


5.4 RuleEngine

RuleEngine позволяет фильтровать и частично обрабатывать всю входящую информацию, благодаря построению цепочек из блоков. В блоках необходимо будет минимальное знание Python для написания или корректировки скриптов внутри блоков. В данном руководстве мы познакомимся с интерфейсом и даже немного задействуем цепочки, в п.4 будут небольшие корректировки и отправка данных от цепочек в систему.

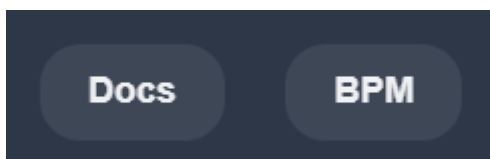


Меню по работе с цепочками расположено в левой части:



При необходимости его можно свернуть, нажав сверху «X», а чтобы вновь

развернуть нажать  в левом верхнем углу.



Кнопка Docs – переводит на подробную документацию по каждому блоку – для чего предназначен блок, как работает, какие входящие и выходящие данные должны быть и т.п.

Кнопка BPM – переводит нас в привычный вид UI Технолога.



Меню настроек интерфейса:



сделать во весь экран



смена темы – есть две – темная и светлая



настройки – позволяют увидеть и изменить ряд настроек работы редактора цепочек:

Настройки

URI API Ядра

URI Дебаг-сокета

URI BPM

Блокировать отключенные элементы ☒

Язык

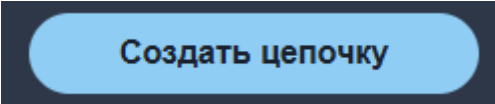
Русский

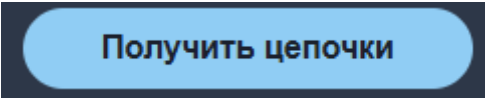
Сохранить

RuleEngine версия:
web: v2.2 dev b899437 06.08.2024, 16:07:44
core: v2.3.2-orm.2-101-g8594479 25.06.2025, 15:50:16

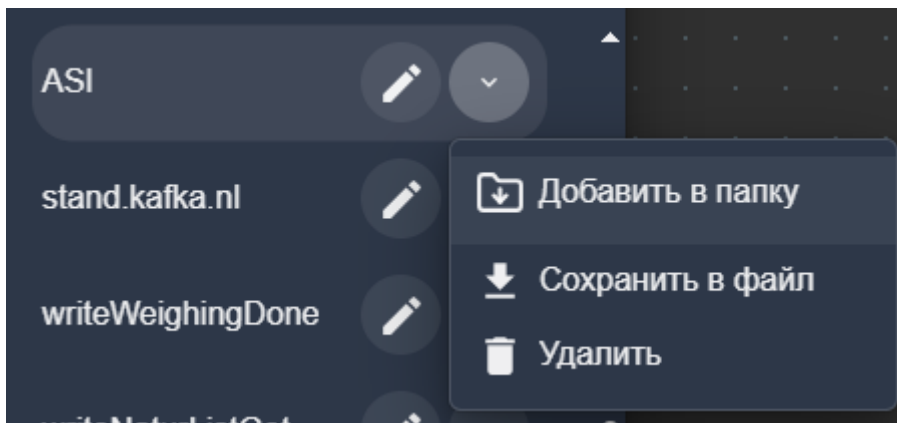
 – запускает краткий гайд по работе с цепочками

 – показывает журнал логов

 Позволяет с нуля создать новую цепочку

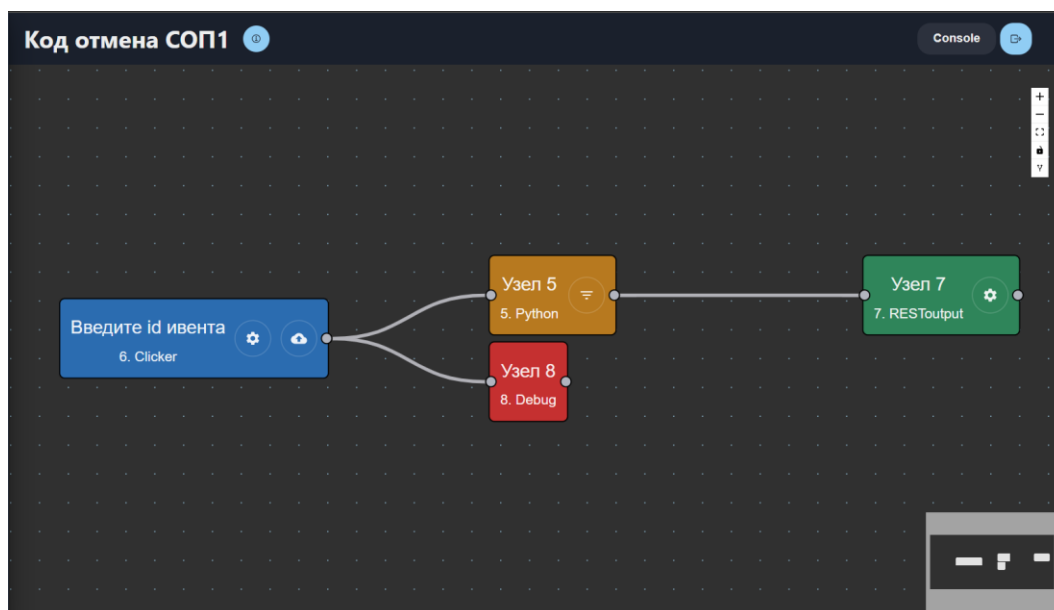
 Подгружает все доступные на сервере цепочки

Ниже идет окно для поиска определенной цепочки по названию. Ниже расположены все цепочки. Рядом с названием цепочки есть две кнопки:



1. Карандаш – для перехода в режим редактирования конкретной цепочки
2. Выбор действий с цепочкой:
 - а. Добавить в папку (можно использовать папки для компоновки цепочек)
 - б. Сохранить в файл – чтобы можно было загрузить на физический носитель или переслать цепочку как вложение
 - с. Удалить – вообще удалить цепочку – восстановить её после этого будет нельзя

При выборе цепочки она отобразится в области справа:



Справа сверху есть кнопки для управления этой областью:



Позволяют приблизить или отдалить цепочку



– максимальное приближение

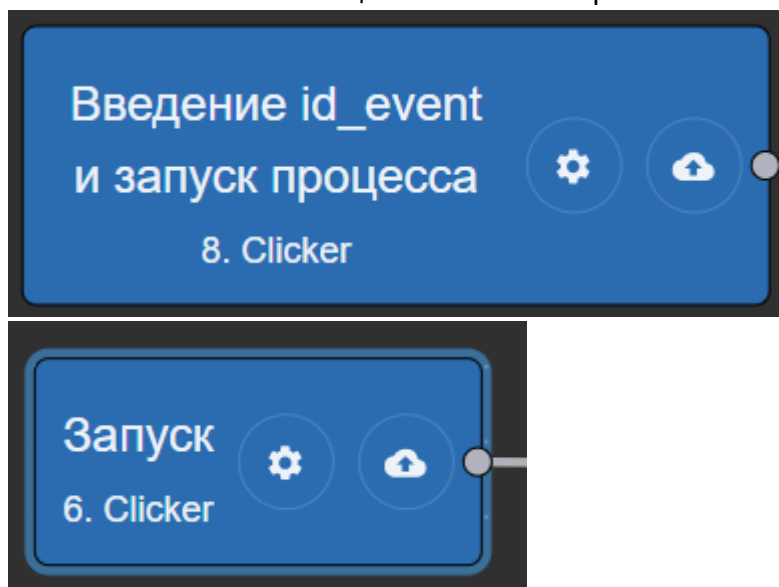


– расправляет все соединения. Любой из блоков цепочки вы можете передвигать в любую часть области. Это не изменит работу, но может сделать цепочку менее удобно читаемой. Для быстрого приведения цепочки в более удобный вид – просто нажимаете эту кнопку, и все блоки расправляются.

В самой цепочке можно увидеть цветные блоки:

Входные блоки

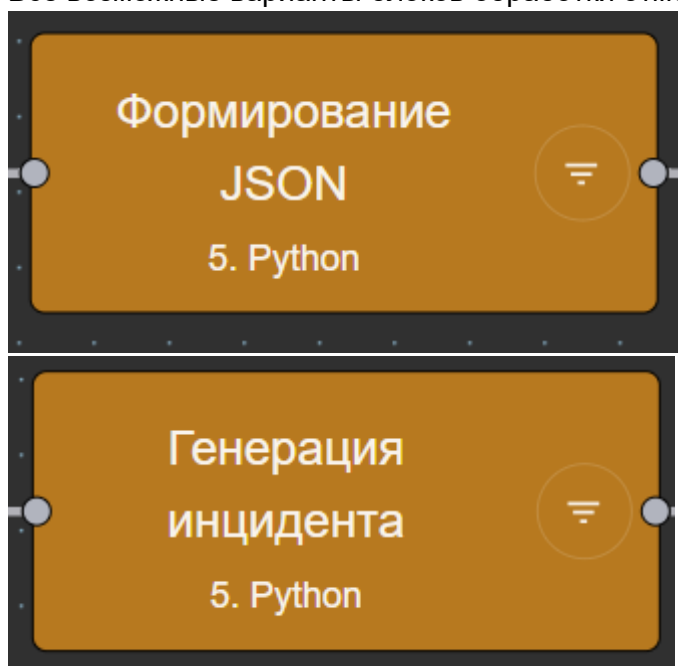
Входной блок цепочки может быть только один. Все возможные варианты входящих блоков отмечены синим цветом – вы выбираете только один на цепочку



Блоки обработки

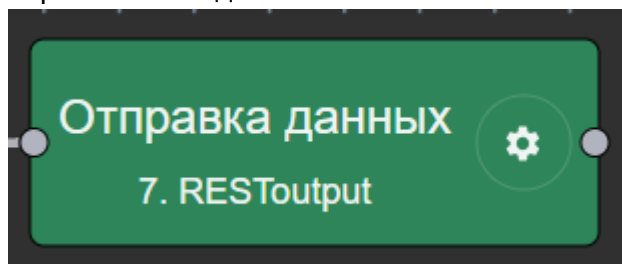
Их может любое количество, они предназначены именно для обработки приходящей информации – добавление, удаление и изменение.

Все возможные варианты блоков обработки отмечены красным или оранжевым цветом.



Выходные блоки

Таких блоков может быть любое количество — они отправляют результаты работ предыдущих блоков в другие компоненты или внешние источники. Все возможные варианты выходных блоков отмечены зеленым цветом.

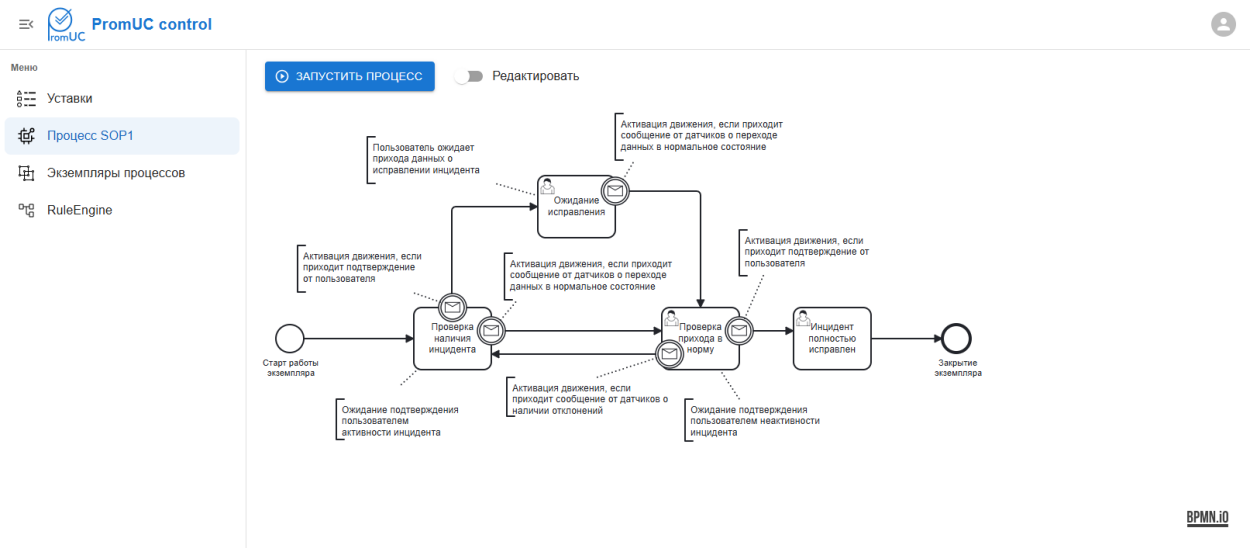


В каждом из блоков есть также дополнительные настройки. В некоторые необходимо прописывать скрипты, в некоторые - только внести определенные данные. Более подробно по каждому из блоков можно почитать в дополнительном приложении по RuleEngine. Для проверки взаимодействия всех модулей системы нам не понадобится большое погружение, так как цепочки будут преднастроены, необходимо будет только внести ряд данных — это будет описано в соответствующем пункте прохождения.

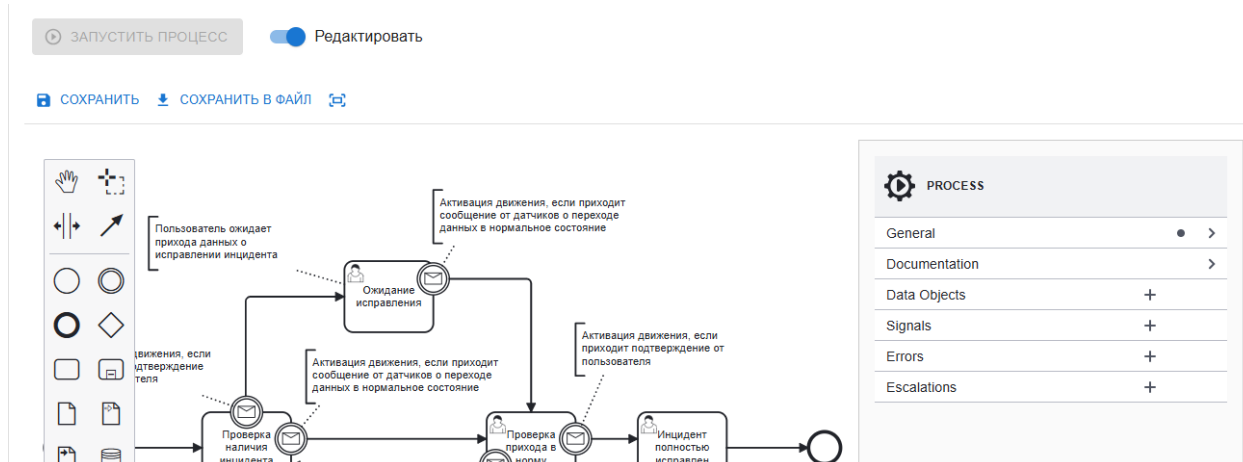
6 Полный цикл инцидента

Познакомившись с интерфейсом, теперь мы можем посмотреть, как возникший инцидент отображается и обрабатывается в системе. Для того, чтобы увидеть изменения в состоянии экземпляра в режиме онлайн, необходимо иногда выходить из режима администратора и переходить в режим пользователя и за него выполнять действия — это необходимо только при работе в демонстрационном режиме работы приложения. В стандартном режиме администратор может просто наблюдать за действиями пользователей. Далее в инструкции, при необходимости смены одной роли на другую будет написано под какой ролью надо зайти - пользователь или администратор.

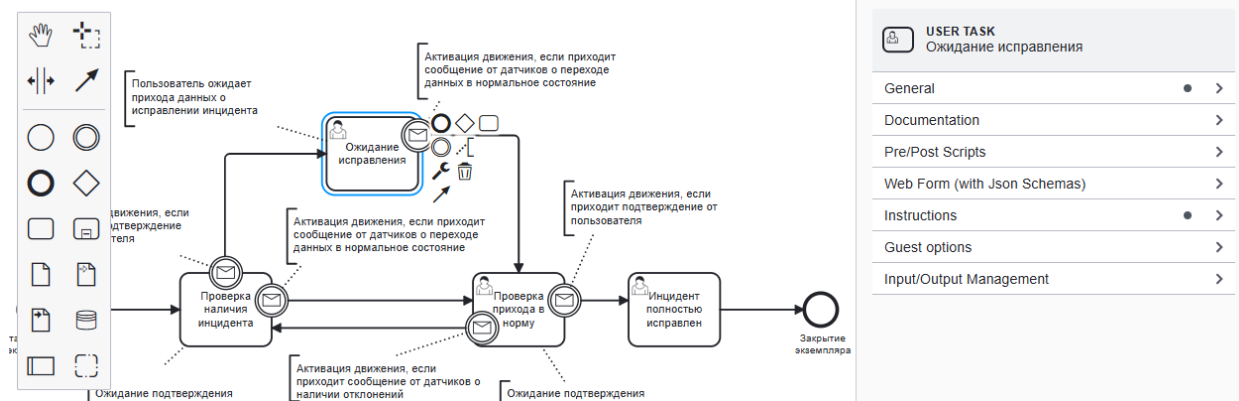
Зайдите в PromUC Control, в раздел Процесс SOP1



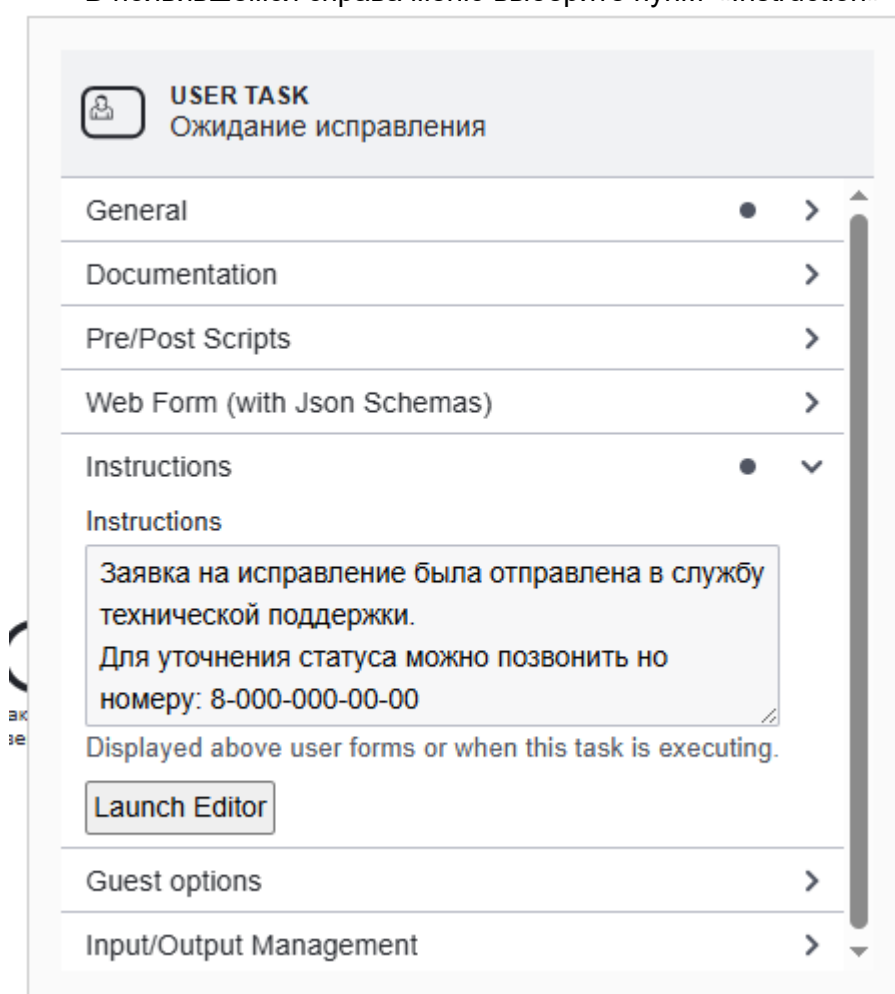
Включите режим редактирования



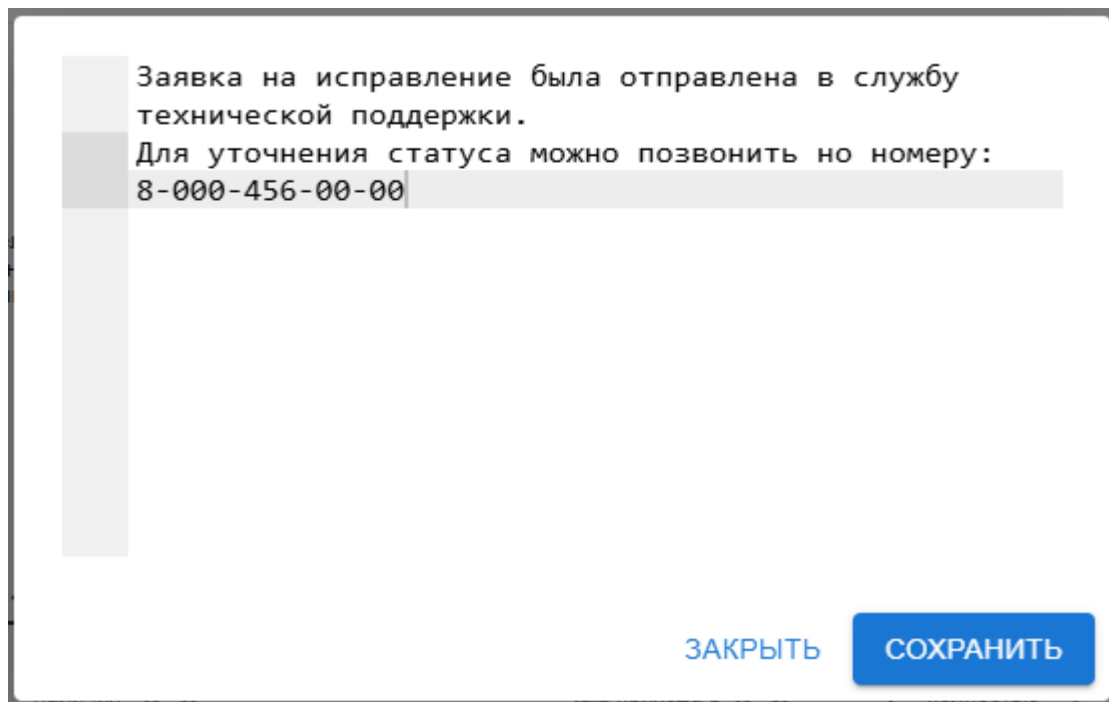
Выберите задачу «Ожидает исправления»



В появившемся справа меню выберите пункт «Instruction»

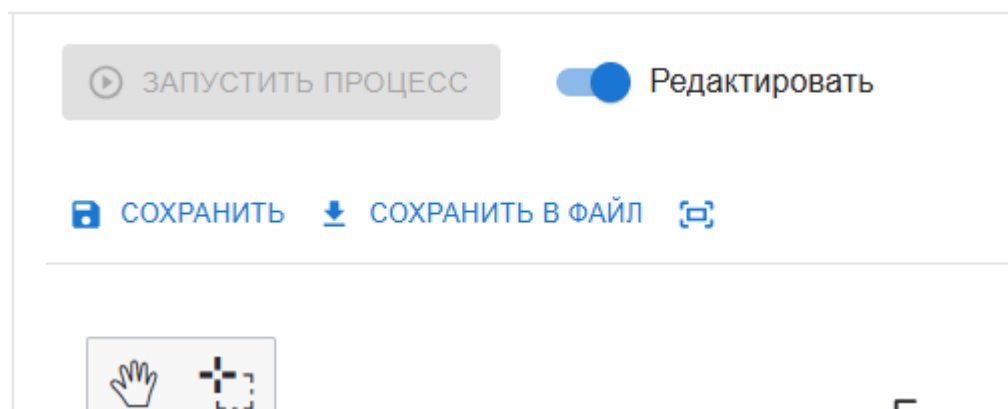


Нажмите кнопку «Launch Editor» и измените номер телефона в тексте – так вы сможете после увидеть именно тот номер, который вы укажете. Номер может быть изменен не точно как в примере – главное запомните внесенные изменения.

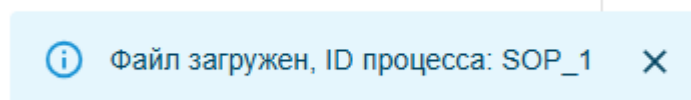


И нажмите кнопку «СОХРАНИТЬ»

Чтобы изменения сохранились на сервере, и при запуске мы точно увидели измененную инструкцию, необходимо нажать на кнопку «СОХРАНИТЬ»



4.7 При корректной работе в левом нижнем углу появится уведомление:



После просмотра уведомления его можно просто закрыть, нажав на «X» в самом уведомлении.

Теперь нам надо выйти из режима редактирования:

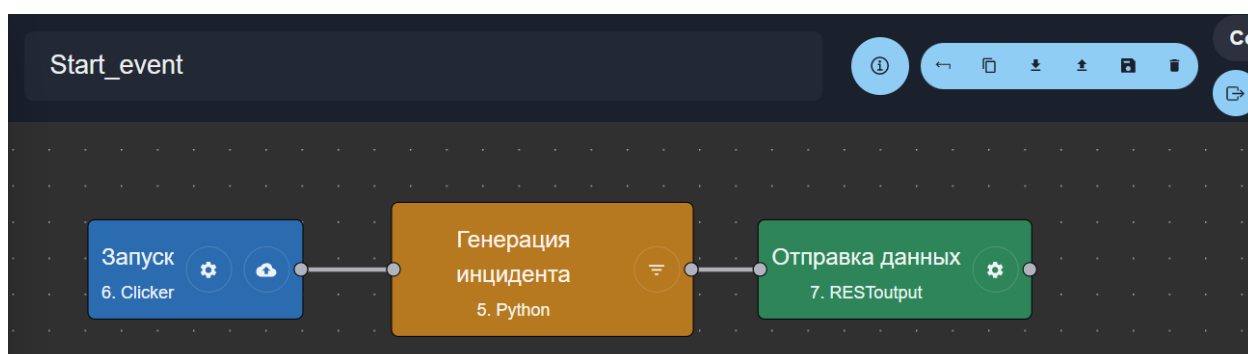


И можно запускать обновленный процесс.
Это можно сделать двумя способами:

- отправив данные из RuleEngine, имитируя реально работающую систему. За исключением того факта, что сигналы будут поступать не от датчиков, а от модуля, который получает данные от датчиков, обрабатывает и передает их дальше.

Для первого раза рекомендуется запускать инициализацию инцидента именно из RuleEngine.

Для этого необходимо перейти во вкладку RuleEngine и выбрать цепочку start_event. Эта цепочка имитирует получение сигналов от датчиков, которые выходят за рамки указанных значений в уставках. Цепочка start_event состоит из 3 блоков:



Входящий блок – Запуск, 6. Clicker

Является обычным кликером - при нажатии на кнопку с облаком  запускается работа цепочки вручную. Автоматический запуск данной цепочки не имеет практического смысла.

Блок обработки - Генерация инцидента, 5. Python

Содержит в себе написанный скрипт для генерации id_event – уникального номера инцидента.

Выходной блок – Отправка данных, 7. RESToutput

Получает от предыдущих блоков данные и отправляет его на указанный в блоке адрес, где эти данные сможет прочитать система и соответствующие модули смогут отреагировать.

Перед запуском необходимо для начала изменить адрес отправки данных. Выбираем цепочку, нажимаем редактировать. Выбираем зеленый блок, и нажимаем на иконку шестеренки. Появляется всплывающее окно:

Настройки блока "7. RESToutput"

Динамическая настройка

☐

Headers

#	KEY	VALUE	
0	Content-Type	application/json	

Метод

POST

URI

http://dev.bpm.promuc.local:8000/bpmn/start-process/

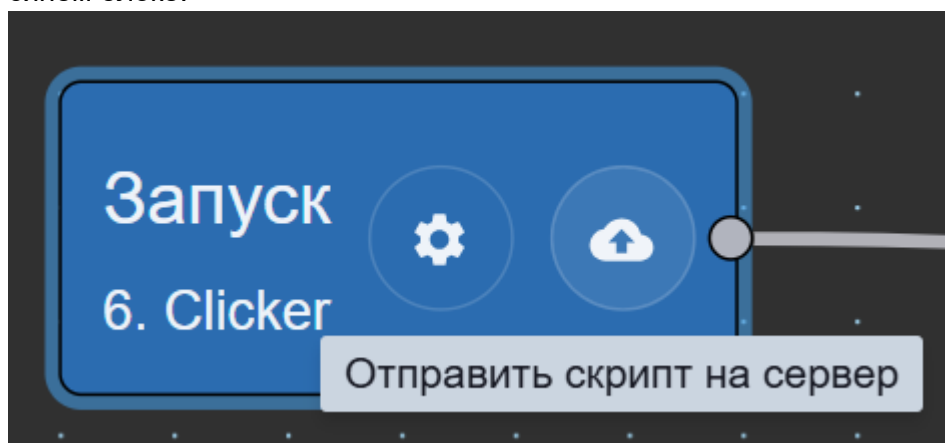
Сохранить Закреть

В URI необходимо заменить часть адреса (выделена жирным и зачеркнута):
[http://~~dev.bpm.promuc.local~~:8000/bpmn/start-process/](http://dev.bpm.promuc.local:8000/bpmn/start-process/)

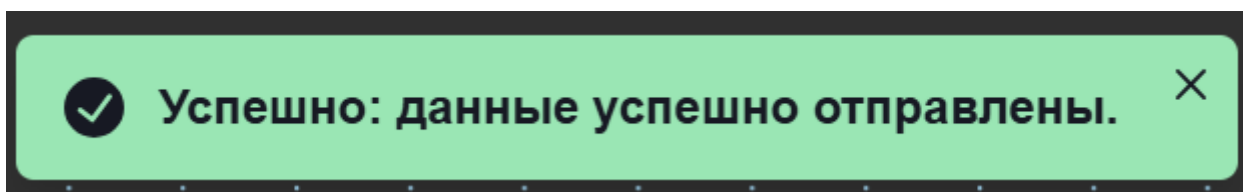
заменить необходимо на тот адрес и порт, который указан в инструкции по установке.

После внесения изменений нажать на кнопку сохранить.

Теперь можно запустить работу цепочки, для этого нажмите на кнопку с облаком на синем блоке:



При отправке данных внизу у вас появится сообщение об успешной отправке данных.

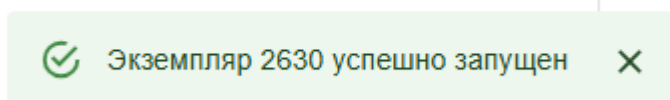


Чтобы узнать номер события необходимо нажать на кнопку BPM, далее зайти в Экземпляры процессов.

После выхода из режима редактирования модели процесса нажать на кнопку «ЗАПУСТИТЬ ПРОЦЕСС». Это можно использовать в тех случаях, когда вы проверяете лишь корректность сработки модели и вывода задачи в интерфейс пользователя. При выборе такого варианта у вас в интерфейсе появится уведомление.



При корректной работе в левом нижнем углу появится зеленое уведомление с номером экземпляра:



Этот номер нам в дальнейшем понадобится, чтобы смотреть состояние инцидента в разных интерфейсах.

ИТОГО:

При запуске инцидента через RuleEngine:

- Вы сразу проверяете взаимодействие модулей системы
- Вы можете проверить, что полученные системой данные пройдут нужный путь
- id_event – номер события можно увидеть в Экземплярах процесса

При запуске инцидента через Процесс SOP1:

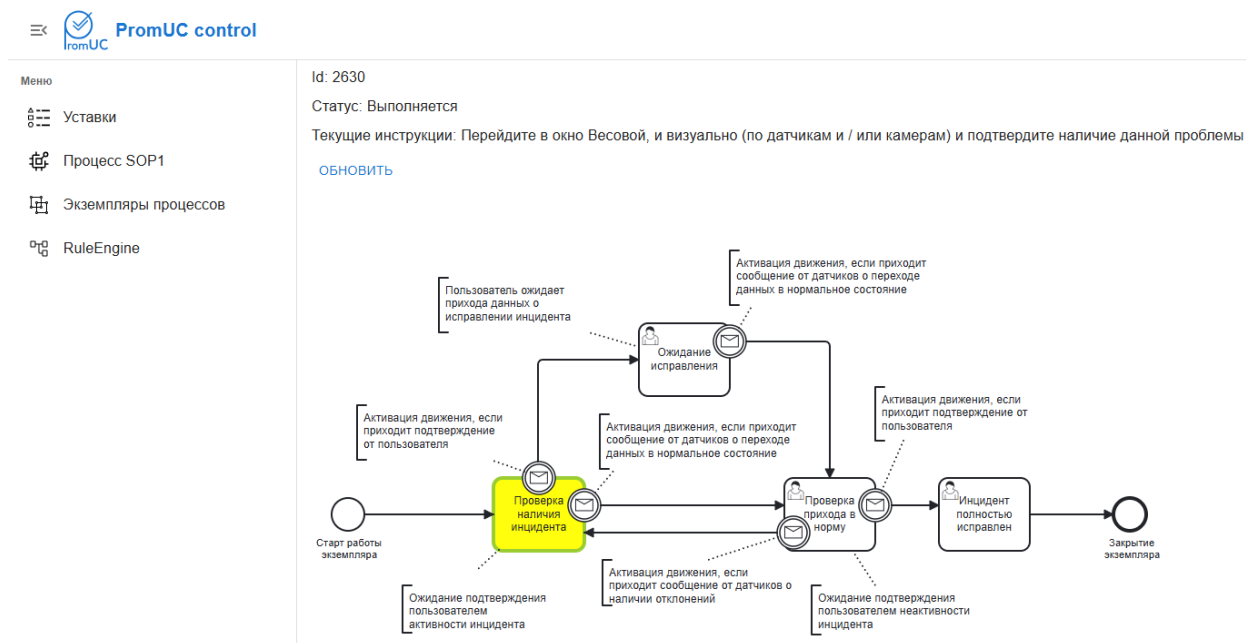
- Вы сразу увидите id_event – номер события (всплывающее сообщение)
- Удобно для проверки возможности вообще запустить процесс (может помочь при отладке системы в целом и для локализации проблемы)

В любом из двух случаев запуска будет инициирован процесс на основе введенной уставки на превышение влажности.

Заходим в Экземпляры процессов и видим, что там появилась запись с нашим номером экземпляра

Меню Уставки Процесс SOP1 Экземпляры процессов RuleEngine	ОБНОВИТЬ					
	id	Имя	Время начала	Время завершения	Статус	Текущая задача
	2630	SOP_1 #2630	30.06.2025, 13:44:57		Ожидание подтверждения	Проверка наличия инцидента ОТКРЫТЬ

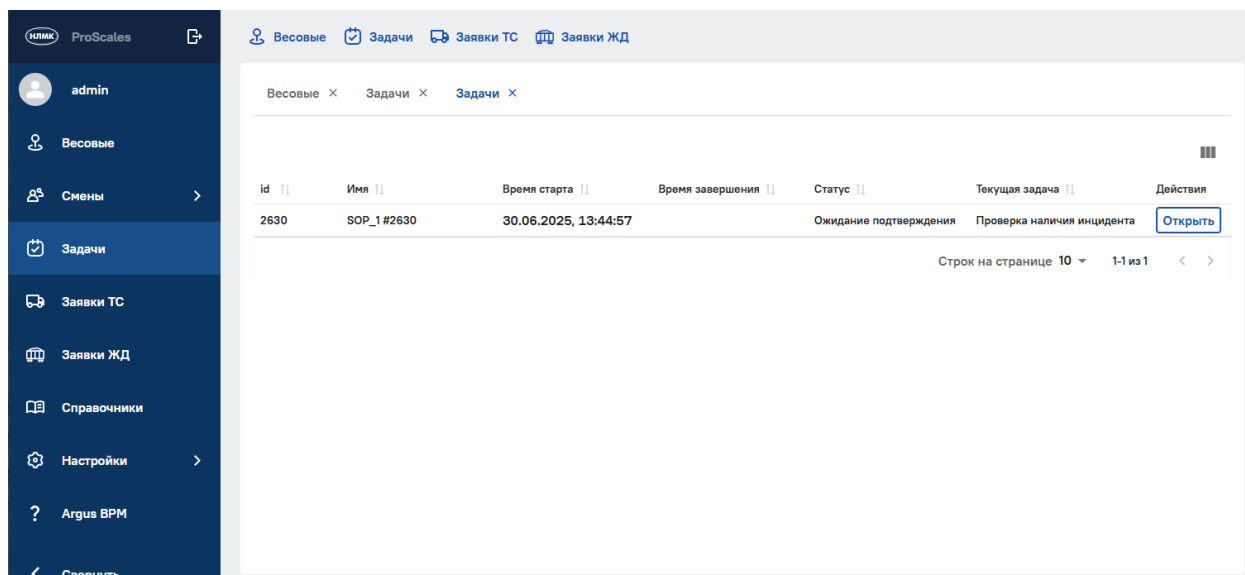
Нажимаем открыть, чтобы увидеть текущее состояние экземпляра



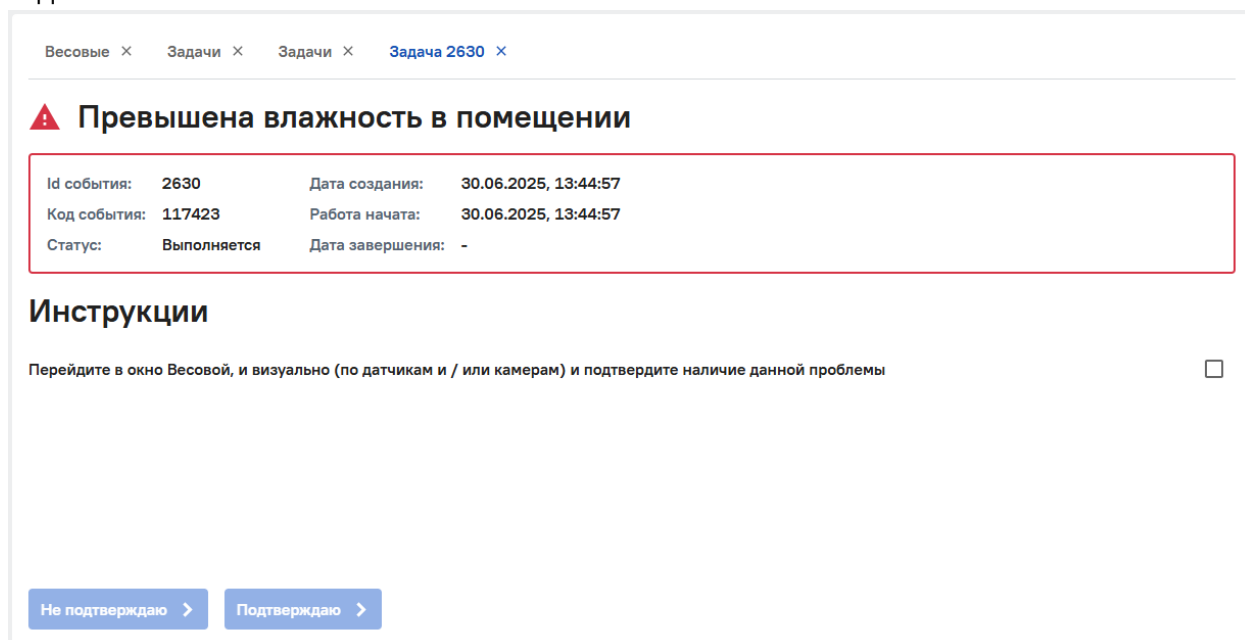
Сверху написан id (его необходимо запомнить, чтобы в интерфейсе пользователя найти нужную задачу). Так же мы видим, на какой задаче мы остановились в текущий момент: нам необходимо подтверждение со стороны пользователя, что инцидент активен для дальнейшего движения.

Переходим в интерфейс пользователя. Расположение интерфейса пользователя задается при установке системы, и точный адрес можно посмотреть именно там.

Переходим во вкладку «Задачи», видим задачу с нашим id. Нажимаем на кнопку «Открыть»



Открывается наша задача.



Теперь необходимо подтвердить наличие инцидента, для этого поставить отметку и после нажать на кнопку «Подтверждаю»

Весовые × Задачи × Задачи × Задача 2630 ×

⚠

Превышена влажность в помещении

Id события: 2630

Дата создания: 30.06.2025, 13:44:57

Код события: 117423

Работа начата: 30.06.2025, 13:44:57

Статус: Выполняется

Дата завершения: -

Инструкции

Перейдите в окно Весовой, и визуально (по датчикам и / или камерам) и подтвердите наличие данной проблемы

Не подтверждаю >

Подтверждаю >

И после этого появляется следующая инструкция, в которой мы видим номер телефона, который указали при редактировании инструкции ранее.

Весовые × Задачи × Задачи × Задача 2630 ×

⚠

Превышена влажность в помещении

Id события: 2630

Дата создания: 30.06.2025, 13:44:57

Код события: 117423

Работа начата: 30.06.2025, 13:44:57

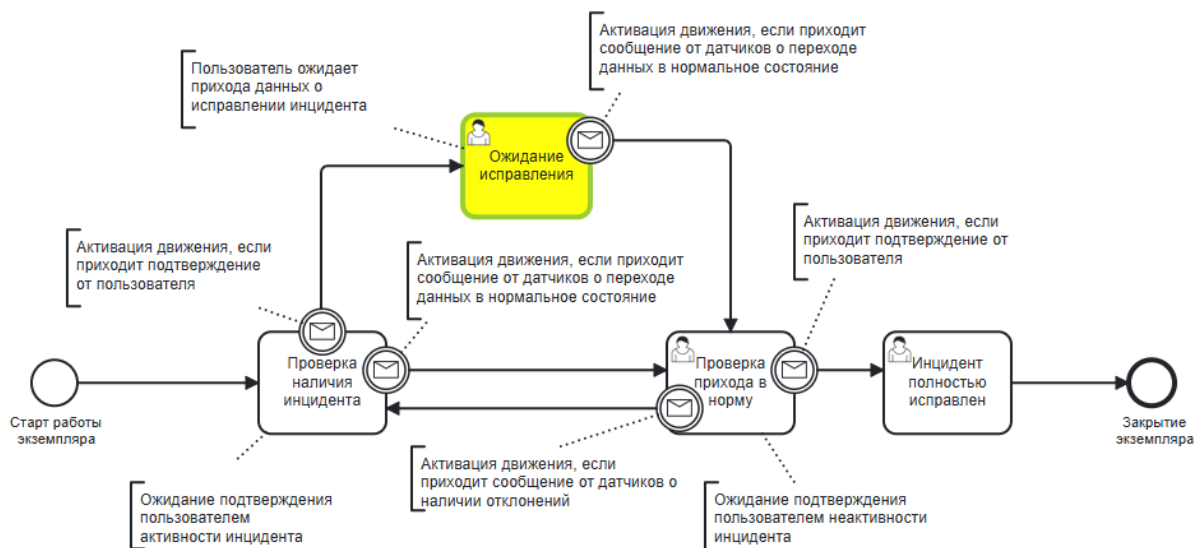
Статус: Выполняется

Дата завершения: -

Инструкции

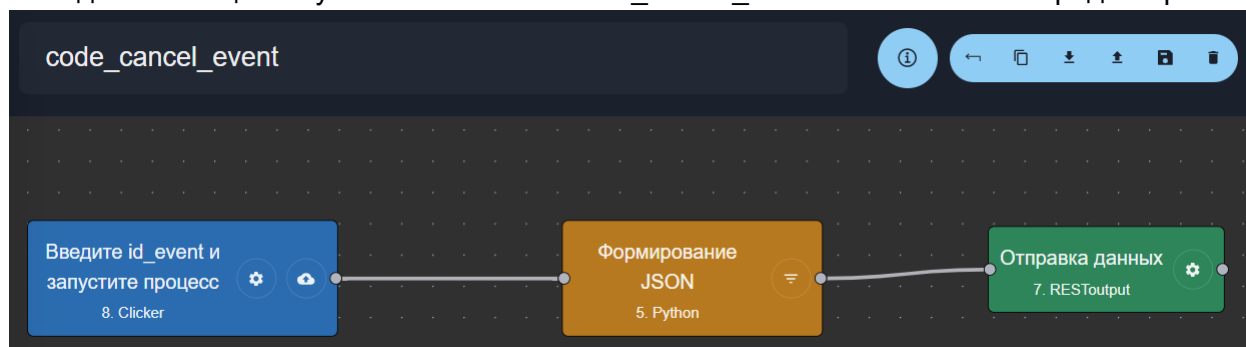
Заявка на исправление была отправлена в службу технической поддержки. Для уточнения статуса можно позвонить по номеру: 8-000-456-00-00

Переходим в PromUC Control —> Экземпляры процессов —> открываем экземпляр с нашим id. И видим текущее состояние



Чтобы перейти к следующему пункту, надо получить сигнал об отмене тревоги.

Теперь необходимо получить сигналы о том, что с датчиков приходит корректная информация. Для имитации отправки такого сообщения, у нас предусмотрена цепочка в RuleEngine. Возвращаемся в PromUC Control → RuleEngine. Находим там цепочку с названием «code_cancel_event» и нажимаем редактировать



В появившейся цепочке выбираем зеленый блок и на нем нажимаем на шестеренку.

Настройки блока "7. RESToutput"

Динамическая настройка

Headers

#	KEY	VALUE	
0	Content-Type	application/json	

Метод

POST

URI

http://dev.bpm.promuc.local:8000/bpmn/message/inactive

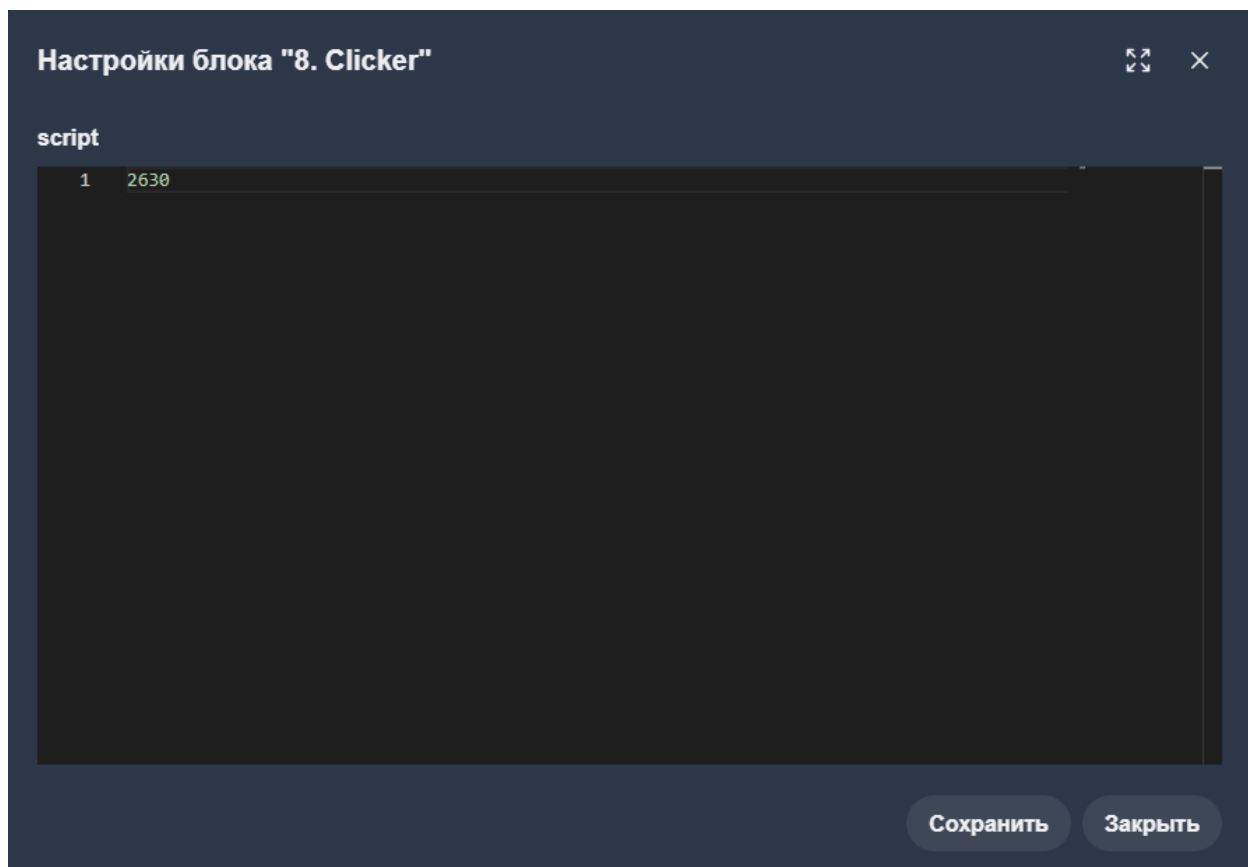
Сохранить

Заккрыть

В URI необходимо заменить часть адреса (выделена жирным и зачеркнута):
`http://dev.bpm.promuc.local:8000/bpmn/start-process/`

заменить необходимо на тот адрес и порт, который указан в инструкции по установке.

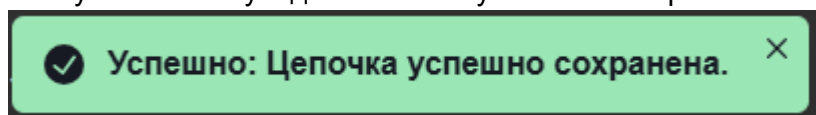
Перед нами появляется всплывающее окно, в котором необходимо в первой строке просто указать id нашего инцидента (он же id задачи или id экземпляра). Кроме этих цифр ничего вводить не надо, после введения нажать «Сохранить»



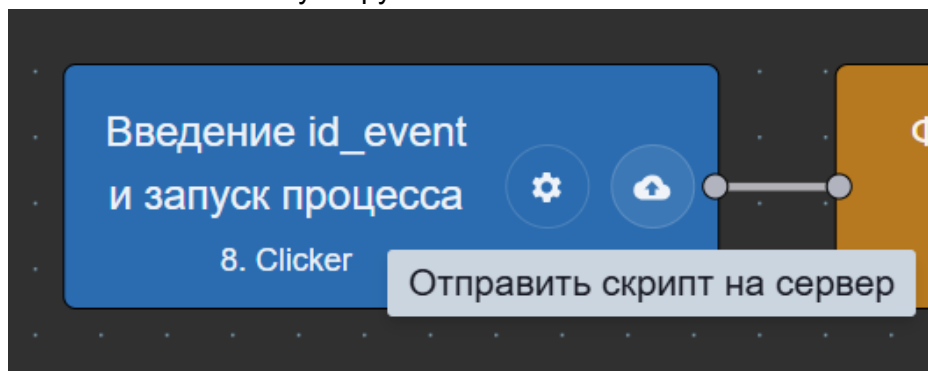
Теперь надо сохранить цепочку с новыми данными. Для этого в правой верхней

части  нажать на иконку сохранения .

Внизу появится уведомление об успешном сохранении изменений



Теперь надо передать данные на сервер. Для этого возвращаемся к первому блоку, и нажимаем на иконку загрузки в облако



После внизу появится уведомление об успешной отправке данных:

Успешно: данные успешно отправлены.

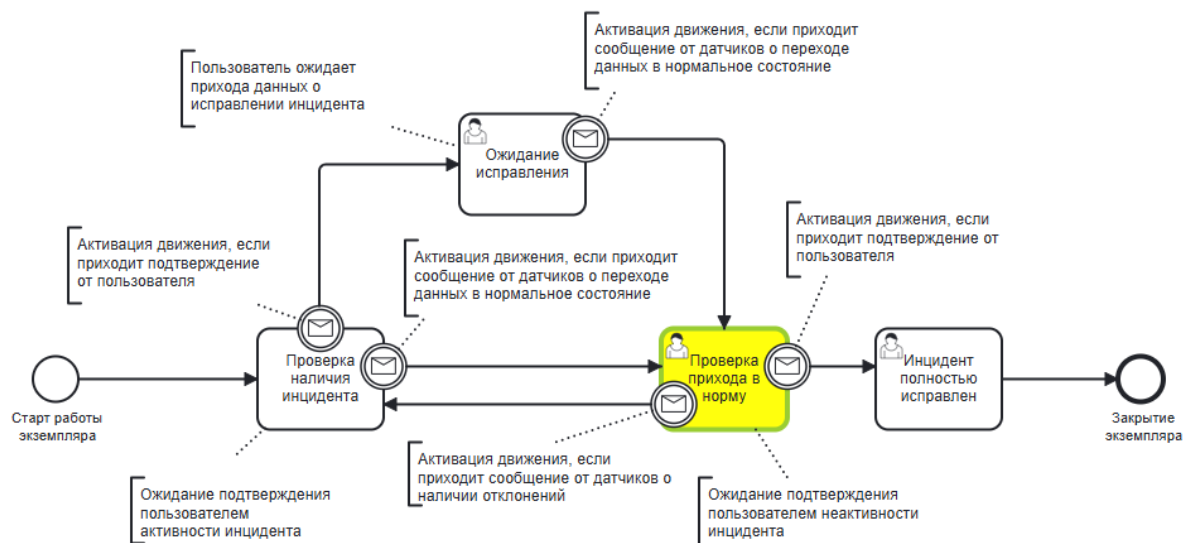
Переходим в PromUC Control —> Экземпляры процессов

ОБНОВИТЬ

id	Имя	Время начала	Время завершения	Статус	Текущая задача	
2630	SOP_1 #2630	30.06.2025, 13:44:57		Ожидание подтверждения	Проверка прихода в норму	ОТКРЫТЬ

Уже в списке можно увидеть изменения в столбце Текущая задача. Открываем экземпляр

И мы видим до какого момента по модели мы уже дошли. Для закрытия задачи осталось получить подтверждение пользователя



Переходим в интерфейс пользователя —> Задачи —> открываем задачу с нашим id

Задачи × Задача 2630 ×

⚠ Превышена влажность в помещении

Id события: 2630 Дата создания: 30.06.2025, 13:44:57
Код события: 117423 Работа начата: 30.06.2025, 13:44:57
Статус: Выполняется Дата завершения: -

Инструкции

Перейдите в окно Весовой, и визуально (по датчикам и / или камерам) и подтвердите отсутствие данной проблемы



Подтверждаю >

и теперь нас просят подтвердить, что инцидент исчерпан – для этого ставим галочку и нажимаем «Подтверждаю»

 Превышена влажность в помещении

Id события:	2630	Дата создания:	30.06.2025, 13:44:57
Код события:	117423	Работа начата:	30.06.2025, 13:44:57
Статус:	Выполняется	Дата завершения:	-

Инструкции

Перейдите в окно Весовой, и визуально (по датчикам и / или камерам) и подтвердите отсутствие данной проблемы

[Подтверждаю >](#)

Появляется сообщение о завершении работ, и в информационную часть добавлено время и дата завершения

 Превышена влажность в помещении

Id события:	2630	Дата создания:	30.06.2025, 13:44:57
Код события:	117423	Работа начата:	30.06.2025, 13:44:57
Статус:	Завершен	Дата завершения:	30.06.2025, 17:01:49

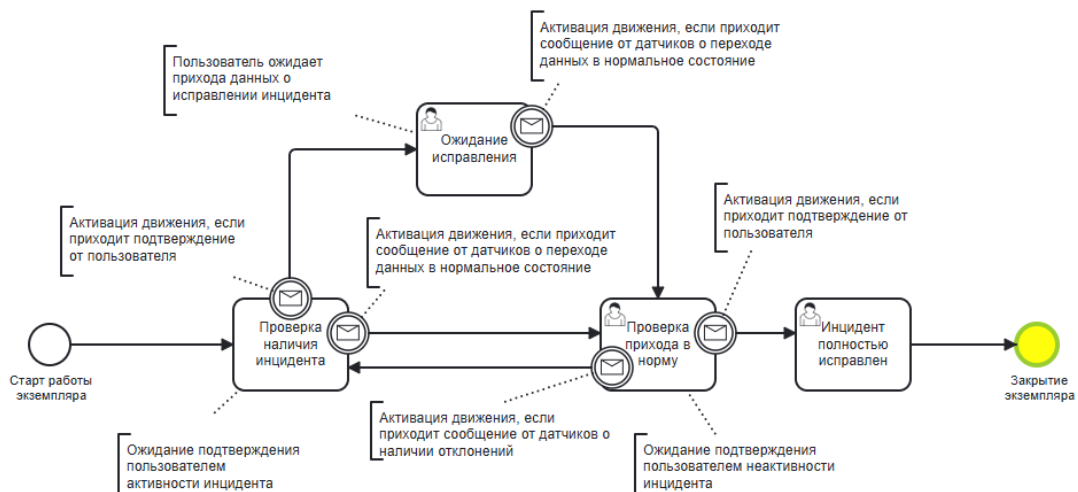
Работа над задачей завершена

Переходим в PromUC Control → Экземпляры процессов

В списке видно, что экземпляр отмечен как завершённый, и появилась дата завершения

2630	SOP_1 #2630	30.06.2025, 13:44:57	30.06.2025, 17:01:49	Завершен	ОТКРЫТЬ
------	-------------	----------------------	----------------------	----------	-------------------------

Открываем сам экземпляр – тут видно, что всё было отработано, и подсвечено, что был совершено закрытие.



Таким образом, мы полностью отработали весь инцидент, и проверили настройку процесса отработки инцидента системы.