

Руководство по настройке

Sherlock Platform

Боты

Книга содержит руководство по работе с ботами Sherlock Platform.

Содержание

1	Бот в системе Sherlock Platform	3
2	Создание/выбор/удаление бота	4
3	Настройка бота	5
3.1	<i>Общие свойства бота</i>	5
3.2	<i>Вкладка Сценарий</i>	6
3.2.1	Работа со списком сценариев	6
3.2.2	Групповые операции со сценариями и командами	7
3.2.3	Настройка общих свойств сценария	9
3.2.4	Добавление и настройка команд сценария	10
3.2.5	Сообщение клиенту	11
3.2.6	Кнопка	12
3.2.7	Внешний запрос	15
3.2.8	Переход к сценарию	16
3.2.9	Вызов пользователя	18
3.2.10	Уведомление	19
3.2.11	Таймер	20
3.2.12	Установка свойств запроса	21
3.2.13	Список	22
3.2.14	Элемент списка	24
3.2.15	Получение данных	25
3.2.16	Условия применения	26
3.3	<i>Настройка событий бота</i>	27
3.3.1	Вкладка События	27
3.3.2	Карточка события	28
3.3.3	Инициализация	29
3.3.4	Сообщение клиента	29
3.3.5	Сообщение оператора	29
3.3.6	Установка статуса	30
3.3.7	Постановка в очередь	30
3.3.8	Передача оператору / начало диалога	31
3.3.9	Внешнее событие	31
3.3.10	Файл от клиента	31
3.3.11	Участник группы добавлен/удален	31
3.3.12	Закрытие окна чата	32
3.3.13	Переход по реферальной ссылке	32
3.3.14	Просмотр информации о канале	32
4	Подключение/запуск бота	33
4.1	<i>Подключение бота в канал</i>	33
4.2	<i>Подключение бота в очереди</i>	34
5	Переменные бота	35
5.1	<i>Реквизиты клиента</i>	36
5.2	<i>Реквизиты диалога</i>	36
5.3	<i>Реквизиты оператора</i>	37
5.4	<i>Реквизиты сообщения</i>	38

1 Бот в системе Sherlock Platform

Боты в Sherlock Platform обеспечивают автоматическую обработку событий в системе. Функционал ботов в системе очень разнообразен – от справочных систем до маршрутизаторов обращений по очередям и реализации интеграции с внешними системами. Например, бот может поздороваться с клиентом, обратившись к нему по никнейму в мессенджере, или распределить ночное обращение в специальную очередь для ночных обращений, или авторизовать клиента в CRM, и т.д.

Механика бота реализуется посредством трёх составляющих – события, сценарии и команды:

- события – правила автоматического срабатывания сценариев бота;
- сценарии – это правила реакции Sherlock Platform на события, сценарии состоят из последовательностей команд, а также включают ограничения по условиям срабатывания.
- команды – конкретные единичные действия Sherlock Platform; команды объединяются в сценарии.



В ходе разработки бота рекомендуется сначала продумать архитектуру бота и только после этого переходить к настройке. Бот – это программа, и как любая программа требует проектирования.

В ходе настройки сначала удобнее настроить сценарии и команды, затем события, вызывающие сценарии бота.

Настройка ботов происходит в модуле боты.

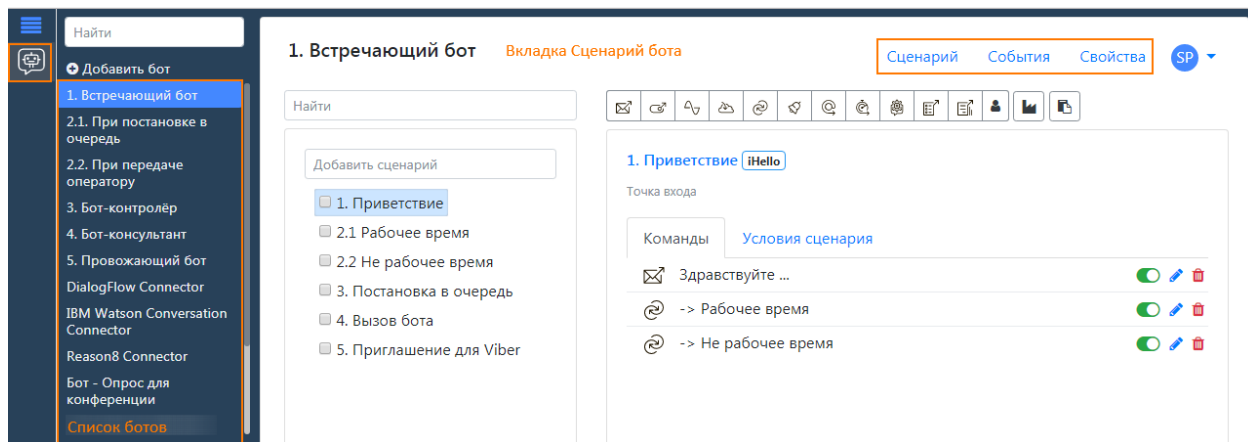



Рисунок 1.1 Модуль Боты

В левой области модуля можно добавлять и удалять ботов, а в правой – настраивать выбранного бота. Подробно по настройке бота см. главу 0.

2 Создание/выбор/удаление бота

Для создания бота, выбора или удаления существующего используется панель ботов в левой части экрана. В верхней части панели ботов доступен поиск по ботам ниже которого отражается список существующих ботов для выбора, а также кнопка **Добавить бот**.

Для отражения или скрытия панели ботов используется кнопка  в левом верхнем углу.

Для создания (добавления) бота достаточно выбрать **Добавить бот** в левой панели. При добавлении появится карточка нового бота с полностью пустыми свойствами. Также на карточке доступна кнопка **Вставить**, позволяющая создать бота на основе существующего, скопированного ранее, бота.

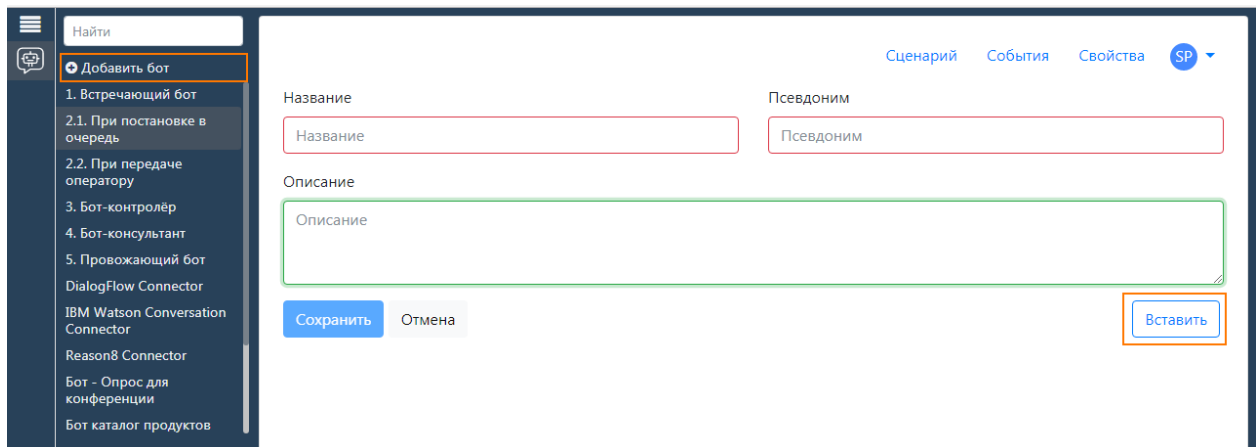


Рисунок 2.1 Добавление бота

Копирование и удаление существующего бота доступно на вкладке свойства по кнопкам **Копировать** и **Удалить** соответственно.

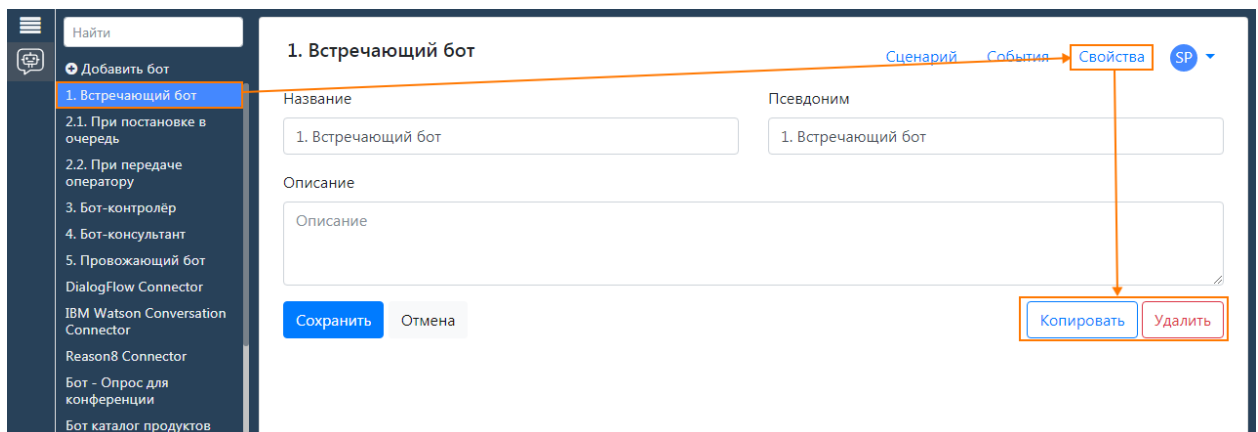


Рисунок 2.2 Удаление/копирование бота

3 Настройка бота

Настройка бота выполняется в правой части окна модуля Боты. Для настройки бота доступны три режима/вкладки:

- Сценарий – настройка сценариев бота
- События – настройка событий бота
- Свойства – просмотр и изменение общих свойств бота

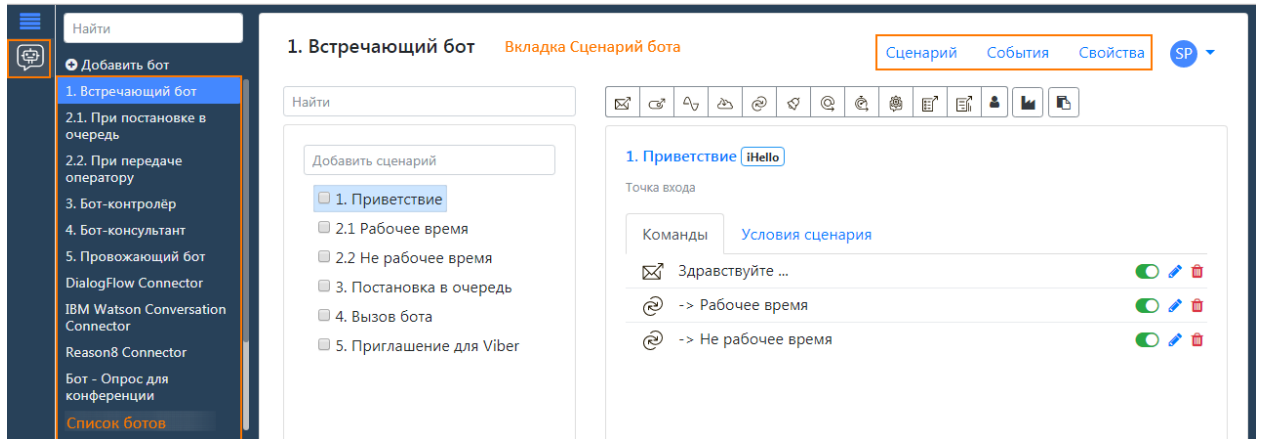



Рисунок 3.1 Сценарий бота

Примечание: Для отражения или скрытия панели ботов используется кнопка  в левом верхнем углу.

3.1 Общие свойства бота

Для изменения свойств бота необходимо перейти на вкладку Свойства

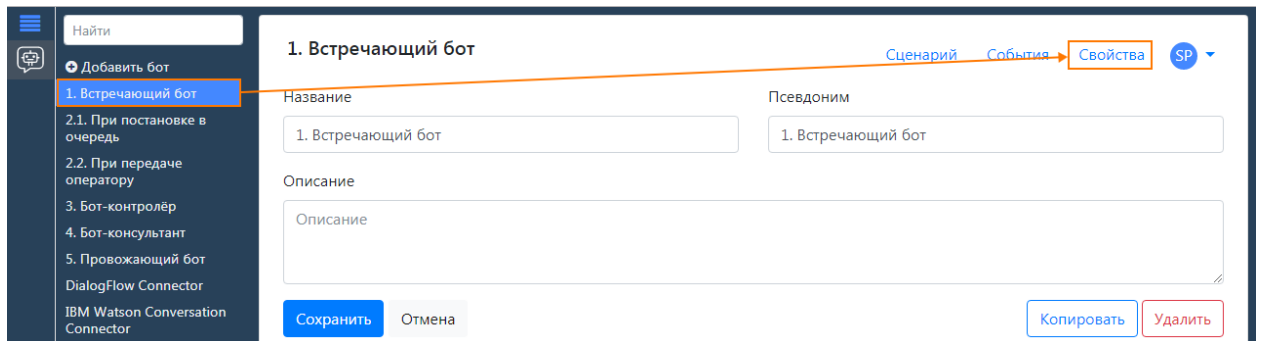


Рисунок 3.2 Общие свойства бота

Общие свойства бота:

Поле	Описание
Название	Название, под которым бот будет отображаться в списке ботов, а также фигурировать в любых полях выбора бота.
Псевдоним	Внутреннее название, используемое для ссылки на бота в командах бота.
Описание	Необязательное к заполнению поле, служащее для уточнения назначения бота, либо для текущих рабочих комментариев. Отображается в списке ботов под названием бота.

3.2 Вкладка Сценарий

Вкладка сценарий содержит:

1. Фильтр по сценариям бота. При вводе ключевых слов список сценариев (3) фильтруется по введенному тексту
2. Поле добавления сценария. Для добавления нового сценария достаточно внести его название и нажать Enter.
3. Список сценариев. Используется для выбора конкретного сценария для настройки
4. Панель операций. Включает кнопки для добавления команд, а также ряд дополнительных операций
5. Панель свойств сценария
6. Список команд сценария
7. Условия выполнения сценария

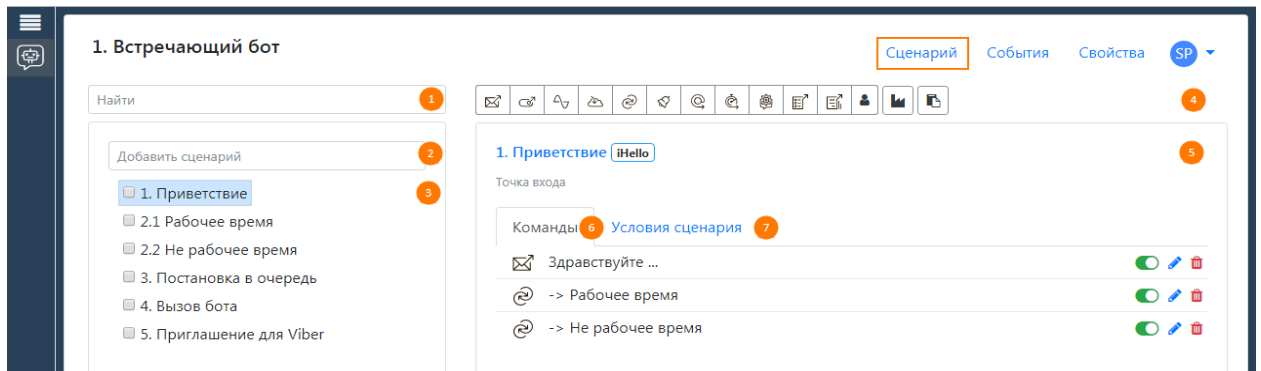


Рисунок 3.3 Вкладка Сценарий бота

3.2.1 Работа со списком сценариев

Основная функция списка сценариев – это поиск и выбор сценария для настройки. Для этого доступен поиск по ключевым словам. Кроме того, сценарии могут группироваться в дереве по папкам.

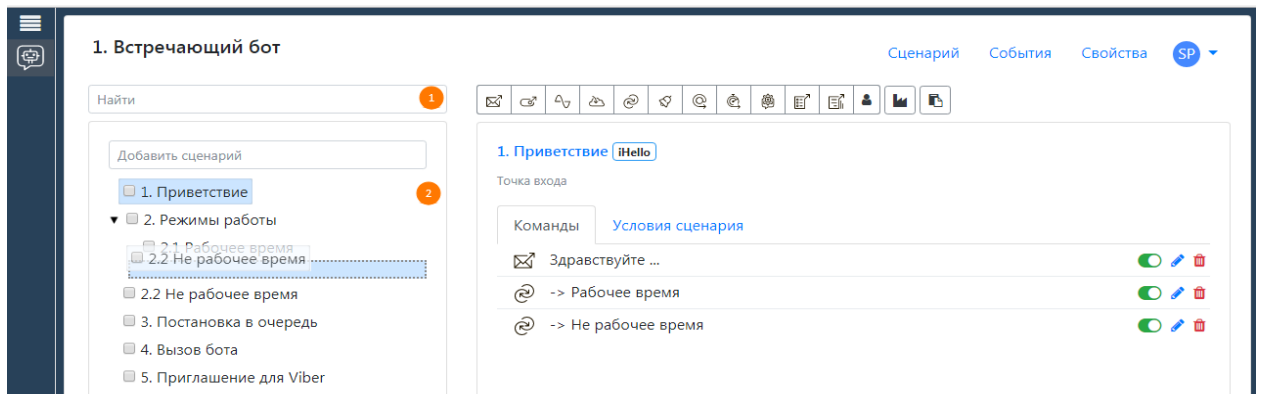


Рисунок 3.4 Основные операции со списком сценариев



1. Как правило настройка сценариев ведется по отдельным функциональным блокам, которые рекомендуется объединять в отдельную папку в дереве.
2. Для удобства настройки рекомендуется использовать последовательную нумерацию сценариев с уточнением уровня вложенности (например, 1, 1.1, 1.1.1, 1.1.1.1). Это позволит отфильтровать сценарии по такому номеру.

Дополнительно в списке сценариев доступны групповые операции и контекстное меню:

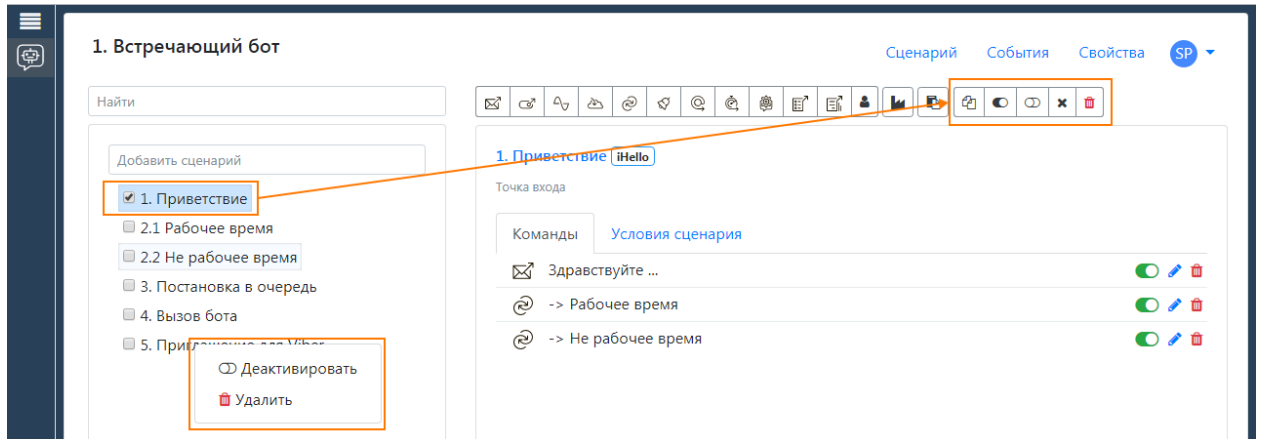


Рисунок 3.5 Дополнительные операции со списком сценариев

Описание групповых операций приведено в разделе 3.2.2 ниже. В контекстном меню доступны операции активации/деактивации сценария, а также операция удаления выбранного сценария.



Деактивация сценария используется для временного исключения сценария из логики обработки ботом. Деактивированный сценарий не включается в алгоритм обработки бот-платформой.

3.2.2 Групповые операции со сценариями и командами

При выборе нескольких сценариев или команд в панели действий отражаются кнопки групповых операций:

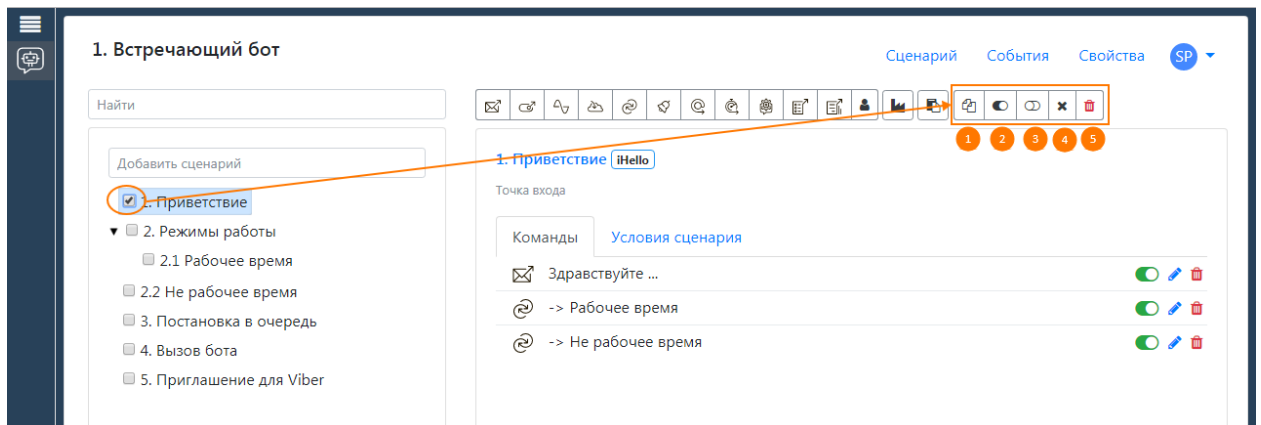


Рисунок 3.6 Групповые операции

Операция копирования (1) позволяет скопировать выбранные сценарии или команды для дублирования или переноса в другой бот. По кнопке **Копирование** отражается диалог с текстом сценариев и/или команд в формате json. Данные текст необходимо скопировать в буфер обмена стандартными операциями копирования операционной системы.

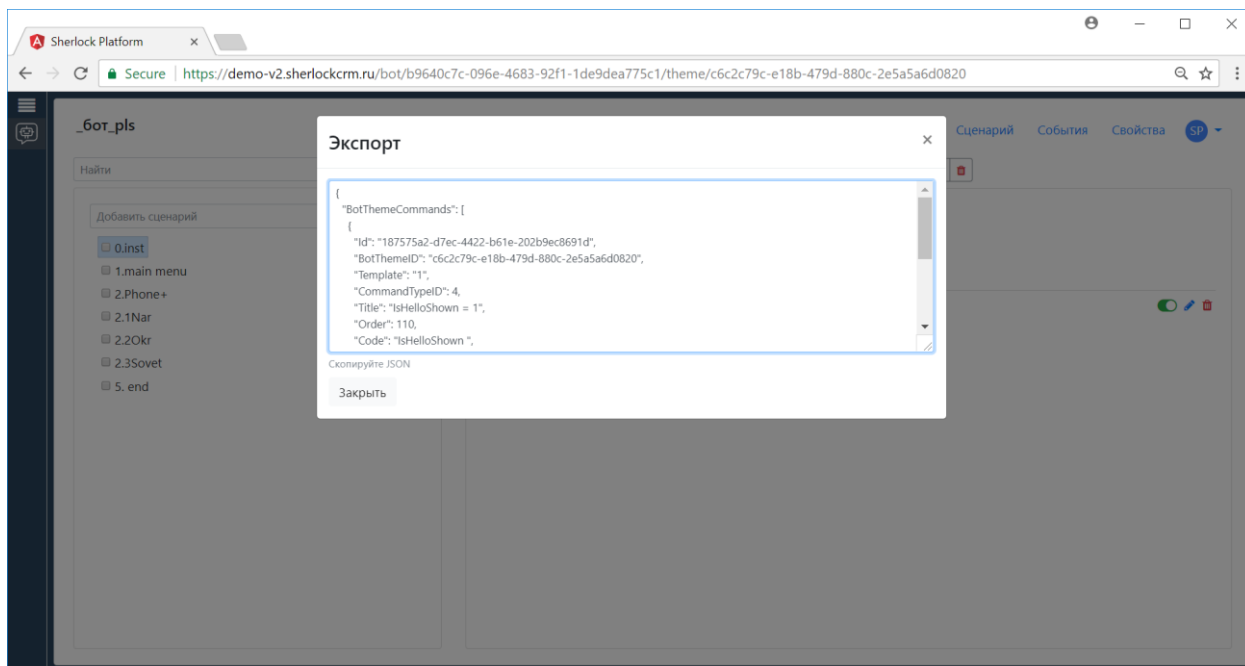


Рисунок 3.7 Диалог копирования сценариев/команд

Для последующего добавления скопированных сценариев/команд используется кнопка **Вставить** в панели действий. По кнопке отражается диалог импорта сценариев и команд, в который надо вставить скопированное ранее в буфер обмена описание сценариев/команд. При импорте сценарии добавляются в конец дерева сценариев, а команды – в выбранный сценарий.

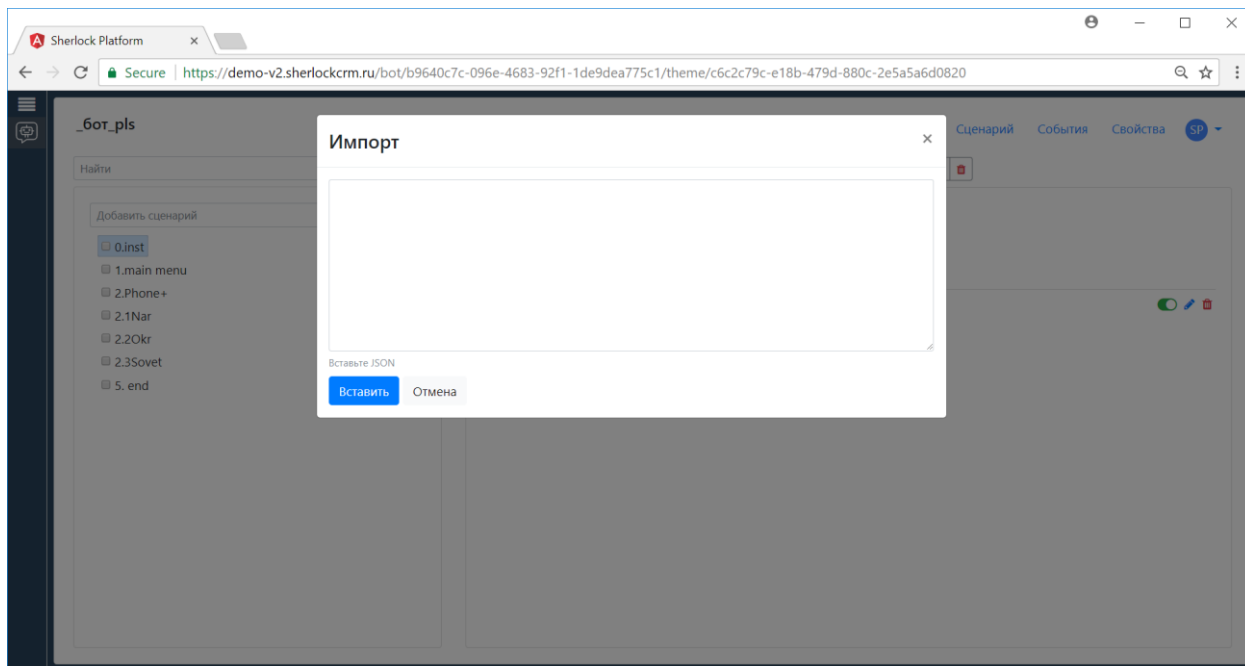


Рисунок 3.8 Диалог добавления сценариев/команд

Операции Активации (2) и Деактивации (3) позволяют соответственно активировать или деактивировать выбранные сценарии или команды.

Снять выделение (4) позволяет очистить текущее выделение сценариев или команд.

Удаление (5) приводит к удалению выбранных элементов. До выполнения операции Sherlock Platform запрашивает подтверждение удаления.

3.2.3 Настройка общих свойств сценария

Для настройки свойств сценария необходимо нажать на название сценария ниже панели операций. По действию Sherlock Platform отразит окно настройки свойств сценария:

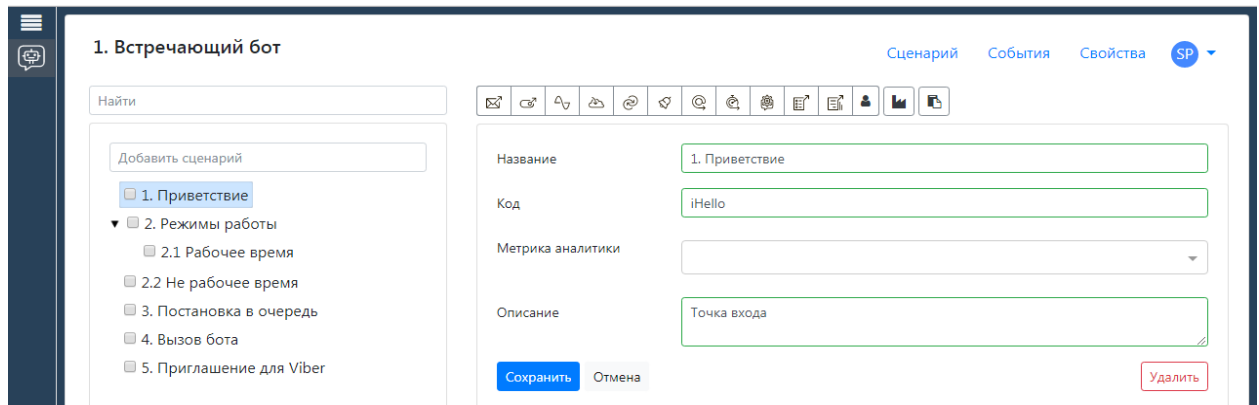


Рисунок 3.9 Окно настройки свойств сценария

Поле	Описание
Название	Название сценария. Выводится в списке сценариев бота. Также используется для любых ссылок на сценарий в командах (например, в команде перехода к указанному сценарию).
Код	Код сценария на английском языке. Служебное обозначение, системное имя сценария. При отправке кода сценария в мессенджер бот выполняет указанный сценарий.
Метрика аналитики	Метрика аналитики, определенная для сценария. Используется для анализа статистики вызова конкретных сценариев в отчетах.
Описание	Описание сценария. Отражается ниже названия сценария.

3.2.4 Добавление и настройка команд сценария

Для добавления новой команды используется одна из кнопок панели операций. После добавления команды (или выбора существующей команды в списке) отражается окно настройки команды, которое содержит **Общие свойства команды**, **Условия применения** и **Дополнительные свойства команды**, зависящие от типа команды:

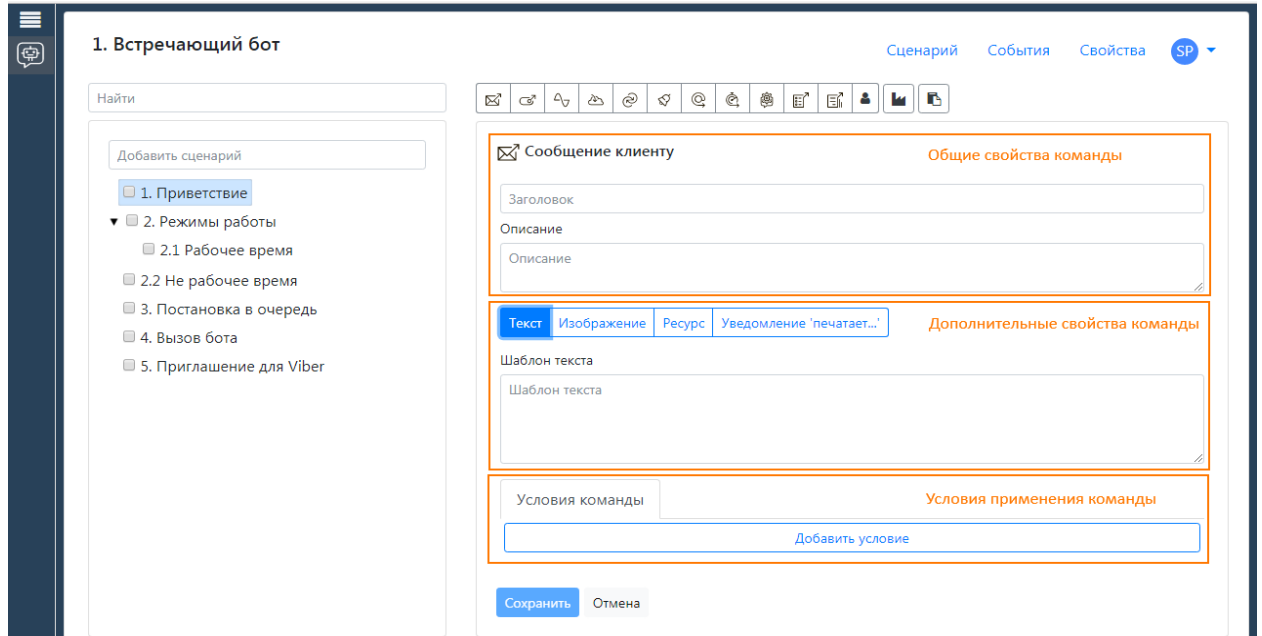


Рисунок 3.10 Окно настройки команды

Поле	Описание
Общие свойства команды	
Заголовок	Название команды.
Описание	Описание команды. Выводится в списке команд темы под названием команды.
Условия применения	
Тип проверки Список условий срабатывания команды. Если условия не заданы, команда срабатывает всегда. Если задано несколько условий, команда срабатывает при выполнении всех условий. Виды условий рассмотрены в разделе 3.2.13.	
Дополнительные свойства команды	
Зависят от Типа команды .	

3.2.5 Сообщение клиенту

Команда, отправляющая сообщение в чат.

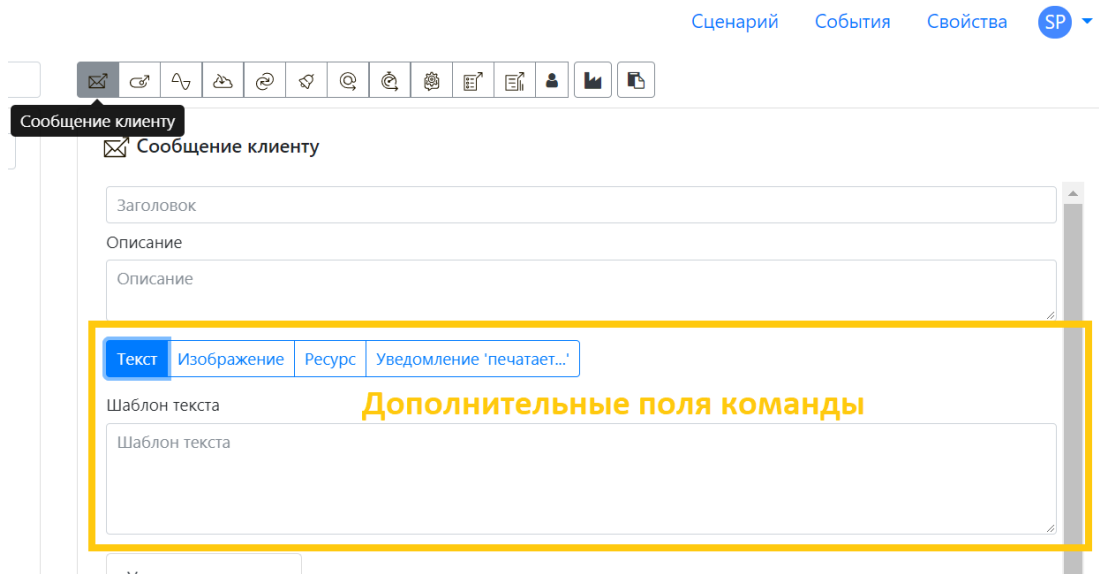


Рисунок 3.11 Типы команд. Сообщение клиенту

Поле	Описание
Текст	
Шаблон текста	Текст к выводу в чат. Возможно использование переменных контекста с применением синтаксиса: #имя_переменной_контекста#
Изображение	
Изображение	Стандартный диалог выбора изображения.
Ресурс	
Шаблон текста	Текст к выводу в чат. Возможно использование переменных контекста с применением синтаксиса: #имя_переменной_контекста# . В шаблоне ресурса доступны реквизиты ресурса Заголовок (#Title#) и Текст (#Text#)
Группа ресурсов	Поле выбора группы ресурсов из настроенных в модуле ресурсов Sherlock Platform ресурсов.
Проверить теги	Поле для фильтрации ресурсов по тегам.
Уведомление "Печатает"	
	Отправка уведомления клиенту (пользователю бота) о подготовке ответа ботом

3.2.6 Кнопка

Команда, отправляющая кнопку в мессенджер.

Рисунок 3.12 Типы команд. Кнопка

Дополнительные поля команды **Кнопка**:

Поле	Описание
Заголовок	Кроме имени команды используется в качестве имени кнопки в мессенджере.
Действие	Действие кнопки. Определяет механику работы команды (см. ниже).
Тип кнопки	Форма отображения кнопки. Работает в зависимости от канала (ВКонтакте, Telegram и т.д.) <ul style="list-style-type: none"> • В тексте – кнопка отражается в тексте сообщения. • Клавиатура – копка отражается как дополнительная клавиатура. • Команда – поддерживается Telegram и задает действие, вызываемое вводом текста /command.
С новой строки	Определяет расположение кнопок в мессенджере. Позволяет группировать кнопки для отражения в одной строке. Кнопка с признаком [С новой строки]=Да отражается на новой строке.
Метрика аналитики	Метрика аналитики, определенная для сценария. Используется для анализа статистики вызова конкретных сценариев в отчетах.
Значение кнопки	Позволяет различить нажатую кнопку в случае вызова одного сценария по нескольким кнопкам. Значение нажатой кнопке сохраняется при переходе к сценарию в переменную контекста бота с именем #ChatButtons_Value#

Дополнительные поля в зависимости от Действия:

Поле	Описание
Переход к теме	
Сценарий	Сценарий бота, к которой перейдёт бот по кнопке. Кнопка справа от списка позволяет скопировать название выбранного сценария
Переход к ссылке	
Шаблон ссылки	Полная отображаемая ссылка. В ссылке возможно использование переменных контекста.
Запрос телефона	
Нет специальных дополнительных полей. Нажатие на кнопку в мессенджере приводит к отправке в Sherlock Platform номера телефона клиента.	
Запрос геолокации	
Нет специальных дополнительных полей. Нажатие на кнопку в мессенджере приводит к отправке в Sherlock Platform геолокации с телефона клиента.	

3.2.6.1 Установка переменной контекста

Присвоение значения переменной контекста.

Рисунок 3.13 Типы команд. Установка переменной контекста

Дополнительные поля команды **Установка переменной контекста**:

Поле	Описание
Код переменной	Имя переменной, в которую сохраняется результат операции. Может использоваться имя переменной контекста или реквизиты клиента/обращения через префиксы Client.* /Request.* соответственно

Дополнительные поля в зависимости от **Вида установки переменной контекста**:

Поле	Описание
Формула	
Формула	<p>Формула на языке C#, результат вычисления которой будет присвоен переменной. В формуле могут использоваться классы C#. При указании формулы необходимо учитывать тип переменных. В частности, строку необходимо указывать в кавычках (например, "текстовое значение").</p> <p>Примеры:</p> <ol style="list-style-type: none"> 1. Извлечение из одной переменной значения в другую по маске <code>System.Text.RegularExpressions.Regex.Match("#ВходнаяПеременная#", "(?:error[\s \D]+)([\d]+)")</code> 2. Присвоение переменной контекста значения текста, присланного пользователем <code>"#System.Text#"</code> 3.
Поиск по маске	
Формула	<p>Формула в формате маски, результат вычисления которой будет присвоен переменной.</p> <p>Для поиска по маске используются следующие значения:</p> <ul style="list-style-type: none"> • 0 - Цифра, необходимый символ. Этот элемент будет принимать любой цифре от 0 до 9 • 9 - Цифра или пробел, необязательный символ. • L - Буква, необходимый символ. Ограничивающий входные данные ASCII буквы a-z и A-Z. Этот элемент маски аналогичен [a-zA-Z] в регулярных выражениях. • ? - Буква, необязательный символ. Ограничивающий входные данные ASCII буквы a-z и A-Z. Этот элемент маски аналогичен [a-zA-Z]? в регулярных выражениях.
Регулярное выражение	
Формула	<p>Формула в формате регулярного выражения, результат вычисления которой будет присвоен переменной.</p> <p>Примечание: для разработки и тестирования регулярных выражений рекомендуется использовать сайт https://regexr.com/</p>
XML	
Код переменной с XML	Имя переменной окружения в формате XML.
XML Path	<p>Путь к данным внутри переменной XML, значение которых будет присвоено переменной.</p> <p>Примечание: для разработки и тестирования запросов выборки данных из JSON рекомендуется использовать утилиту https://www.freeformatter.com/xpath-tester.html</p>
JSON	
Код переменной с JSON	Имя переменной окружения в формате JSON.

Поле	Описание
Объект JavaScript	Путь к данным внутри переменной JSON, значение которых будет присвоено переменной. Примечание: для разработки и тестирования запросов выборки данных из JSON рекомендуется использовать утилиту http://jsonparser.sherlockcrm.ru/ или сайт http://jsonpath.com/ . http://jsonparser.sherlockcrm.ru/ предпочтительнее, так как использует библиотеку Sherlock Platform. На сайте http://jsonpath.com/ поддерживается ряд конструкций, которые не примет http://jsonparser.sherlockcrm.ru/

3.2.7 Внешний запрос

Запрос к внешним системам.



Используется в сложных ботах и в интеграциях.

Рисунок 3.14 Типы команд. Внешний запрос

Дополнительные поля команды **Внешний запрос**:

Поле	Описание
Подключение к внешнему источнику	Выбор из списка реализованных на уровне сервера Sherlock Platform подключений.

Дополнительные поля в зависимости от **Вида внешнего запроса**:

Поле	Описание
SQL запрос	
Команда	SQL запрос к БД. При формировании запроса могут использоваться переменные контекста
SOAP запрос	
URL относительно источника	Путь относительно внешнего источника.
Данные	Данные, отправляемые на сервер. При формировании запроса могут использоваться переменные контекста
REST запрос	

Поле	Описание
Путь	Путь (URL) запроса относительно внешнего источника.
Данные	Данные, отправляемые на сервер. При формировании запроса могут использоваться переменные контекста. Данные могут быть в формате JSON или form data.
Метод	Метод HTTP запроса. Сейчас поддерживаются методы GET, POST, PUT, DELETE

Результаты выполнения запроса сохраняются в переменные контекста:

1. Для SQL запроса в переменные контекста **SQL_Result** и **SQL_RecordCount**. Обращаться к ним можно как к обычным переменным контекста, то есть **#SQL_Result#** и **#SQL_RecordCount#**.
2. Для SOAP запроса в переменную контекста **SOAP_Result**. Обращаться к переменной можно как к обычным переменным контекста, то есть **#SOAP_Result#**.
3. Для REST запроса в переменные контекста **REST_Result** и **REST_StatusCode**. Обращаться к переменной можно как к обычным переменным контекста, то есть **#REST_Result#** и **#REST_StatusCode#**.

3.2.8 Переход к сценарию

Используется для вызова другого сценария текущего или другого бота



Используется для ветвлений или вызова процедур в боте.

Рисунок 3.15 Типы команд. Переход к сценарию

Дополнительные поля команды **Переход к теме**:

Поле	Описание
Бот	Выбор бота, к сценарию которого должен быть переход. По умолчанию – текущий бот.
Сценарий	Поле выбора сценария бота для перехода из список доступных сценариев бота.

Поле	Описание
Продолжить сценарий после вызываемого	Если да, то сценарий вызывается с возвратом к текущему после вызываемого по аналогии с вызовом процедуры/функции в программировании.
Покинуть диалог	Если да, то текущий бот прекращает работу при вызове сценария

3.2.9 Вызов пользователя

Используется для подключения к диалогу пользователя.

Рисунок 3.16 Типы команд. Вызов пользователя/бота

Дополнительные поля команды **Вызов пользователя/бота**:

Поле	Описание
Вызвать пользователя	Тип вызова. Определяет механику работы команды (см. ниже).
Покинуть диалог	Флаг отключения вызывающего бота от диалога после выполнения команды.

Дополнительные поля в зависимости от поля **Вызвать пользователя**:

Поле	Описание
Через очередь	
Очередь	Поле выбора очереди из списка настроенных очередей.
Конкретный пользователь	
Пользователь/бот	Поле выбора пользователя/бота из списка сотрудников в системе. Все боты в системе автоматически привязываются к сгенерированным техническим пользователям.

3.2.10 Уведомление

Отправка сообщения на почту через SMTP сервер.

Сценарий События Свойства SP

Уведомление

Заголовок

Описание

Описание

SMTP Сервер

Шаблон заголовка

Шаблон заголовка

Шаблон текста

Шаблон текста

Уведомлять

Ответственного

Условия команды

Добавить условие

Сохранить Отмена

Дополнительные поля команды

Рисунок 3.17 Типы команд. Уведомление

Дополнительные поля команды **Уведомление**:

Поле	Описание
SMTP Сервер	Поле выбора SMTP сервера для отправки сообщения из настроенных на уровне сервера Sherlock Platform.
Шаблон заголовка	Шаблон заголовка сообщения. Можно использовать переменные контекста.
Шаблон текста	Шаблон текста сообщения. Можно использовать переменные контекста.
Уведомлять	<ul style="list-style-type: none">• Ответственного (ответственного за диалог)• По списку адресов (с дополнительным полем для списка)

3.2.11 Таймер

Паузы и отложенные действия в работе ботов. Команда управляет работой бота наравне с событиями.



Используется для ведения диалога в ходе ожидания ответа оператора, запроса оценки через какое-то время после общения и т.п.

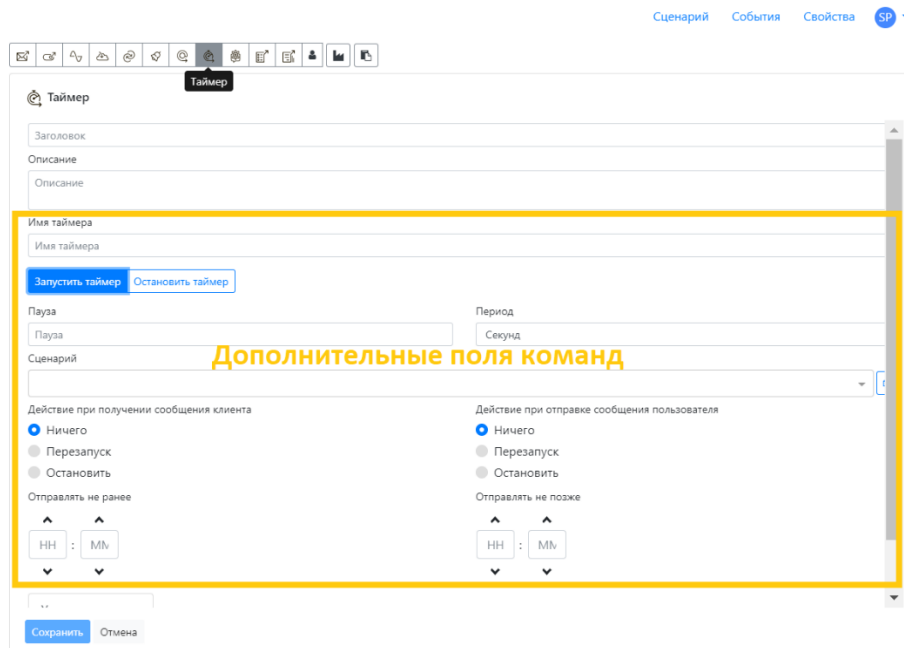


Рисунок 3.18 Типы команд. Таймер

Дополнительные поля команды **Таймер**:

Поле	Описание
Имя таймера	Наименование таймера. Используется для обращений к таймеру, в частности для его остановки.

Дополнительные поля в зависимости от **Вида команды**:

Поле	Описание
Запустить таймер	
Пауза	Длительность паузы. Число.
Период	Размерность длительности паузы. Секунд/Минут/Часов/Дней
Сценарий	Сценарий бота, на который будет осуществлён переход по срабатыванию таймера.
Действие при получении сообщения клиента	<ul style="list-style-type: none"> Ничего Перезапуск Остановить (остановка таймера, переход к сценарию не осуществляется)
Действие при получении сообщения клиента	<ul style="list-style-type: none"> Ничего Перезапуск Остановить (остановка таймера, переход к сценарию не осуществляется)

Поле	Описание
Отправлять не ранее	Ограничение запуска сценария временем суток «снизу».
Отправлять не позже	Ограничение запуска сценария временем суток «сверху».

3.2.12 Установка свойств запроса

Команда используется для изменения свойств запроса/диалога. В том числе для закрытия текущего запроса.

Рисунок 3.19 Типы команд. Установка свойств запроса

Дополнительные поля команды **Установка свойств запроса**:

Поле	Описание
Статус	Поле выбора статуса запроса из настроенных в Sherlock Platform.
Оценка для запроса	Запись значения в системную переменную контекста <code>Request.REQUEST_GRADE</code> . Обращаться к переменной можно как к обычным переменным контекста, то есть <code>#Request.REQUEST_GRADE#</code> .
Приоритет запроса	Установка приоритета запроса для обработки очередью при распределении диалога операторам. <ul style="list-style-type: none"> Высокий Низкий Средний
Исключить из очереди	Флаг исключения из очереди. Если да, то бот исключается из текущей очереди на распределение операторам
Остановка бота	Флаг остановки бота. Если да, то бот исключается из диалога после выполнения команды

3.2.13 Список

Команда Список предназначена для вывода наборов данных пользователям бота с поддержкой постраничного разбиения

Рисунок 3.20 Типы команд. Список

Дополнительные поля команды **Список**:

Поле	Описание
Код списка	Префикс для переменных, в которые будут сохраняться переменные, необходимые для работы списка
Тип источника	Источник данных для списка. Дополнительные реквизиты зависят от типа источника и описаны ниже
Количество на странице	Указывается количество записей, отражаемых на странице при постраничном выводе. Для вывода всех записей указывается 0
Режим отражения	Определяется что отражается в списке: <ul style="list-style-type: none"> Только кнопки Только текст Текст + кнопки
Шаблон элемента списка	Отражается для режимов Только текст и Текст + кнопки . Текст для вывода в чат по каждому элементу списка. Возможно использование переменных контекста с применением синтаксиса: #имя_переменной_контекста# Для доступа к реквизитам списка также используется синтаксис: #имя_столбца_списка#
Шаблон текста кнопки	Отражается для режимов Только кнопки и Текст + кнопки . Текст надписи на кнопке по каждому элементу списка. Возможно использование переменных контекста с применением синтаксиса: #имя_переменной_контекста# Для доступа к реквизитам списка также используется синтаксис: #имя_столбца_списка#

Поле	Описание
Шаблон значения кнопки	Отражается для режимов Только кнопки и Текст + кнопки . Значение, направляемое в чат при нажатии на кнопку, по каждому элементу списка. Возможно использование переменных контекста с применением синтаксиса: #имя_переменной_контекста# Для доступа к реквизитам списка также используется синтаксис: #имя_столбца_списка#
Тип кнопки	Форма отображения кнопки. Работает в зависимости от канала (ВКонтакте, Telegram и т.д.) <ul style="list-style-type: none"> • В тексте – кнопка отражается в тексте сообщения. • Клавиатура – копка отражается как дополнительная клавиатура. • Команда – поддерживается Telegram и задает действие, вызываемое вводом текста /command.
Сценарий при выборе элемента списка	Сценарий бота, вызываемый по нажатию на кнопку в списке.

Дополнительные поля в зависимости от поля **Тип источника**:

Поле	Описание
Ресурсы	
Группа ресурсов	Поле выбора группы ресурсов из настроенных в модуле ресурсов Sherlock Platform ресурсов.
Проверить теги	Поле для фильтрации ресурсов по тегам.
XML	
Переменная контекста	Имя переменной окружения в формате XML.
Запрос XPath	Путь к данным внутри переменной XML, значение которых будет присвоено переменной.
JSON	
Переменная контекста	Имя переменной окружения в формате JSON.
Запрос JSONPath	Путь к данным внутри переменной JSON, значение которых будет присвоено переменной. Примечание: для разработки и тестирования запросов выборки данных из JSON рекомендуется использовать утилиту http://jsonparser.sherlockcrm.ru/ или сайт http://jsonpath.com/ . http://jsonparser.sherlockcrm.ru/ предпочтительнее, так как использует библиотеку Sherlock Platform. На сайте http://jsonpath.com/ поддерживается ряд конструкций, которые не примет http://jsonparser.sherlockcrm.ru/

3.2.15 Получение данных

Команды Получение данных позволяет получить от пользователя строку, проверить ее значение и сохранить в переменную контекста.

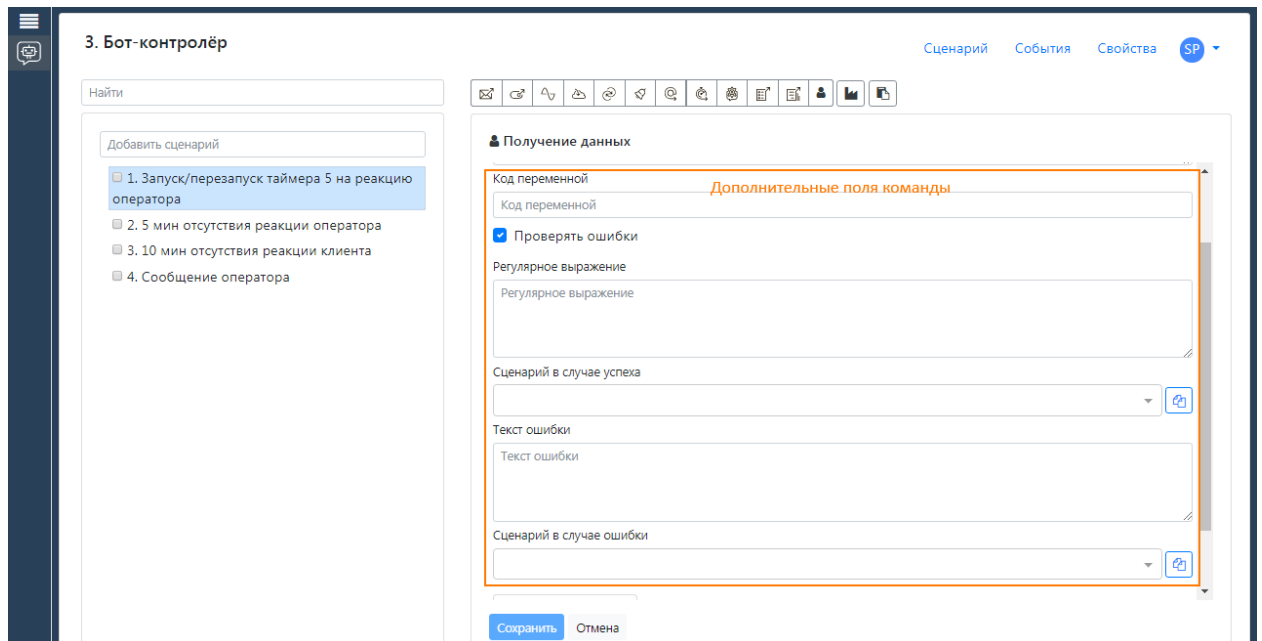


Рисунок 3.22 Типы команд. Получение данных

Дополнительные поля команды **Получение данных**:

Поле	Описание
Код переменной	Имя переменной, в которую сохраняется результат операции.
Сценарий в случае успеха	Сценарий, к которому выполняется переход после получения необходимых данных
Проверять ошибки	Если да, то полученный текст проверяется в соответствии с заданными правилами
Регулярное выражение	Регулярное выражение для проверки полученного текста Примечание: для разработки и тестирования регулярных выражений рекомендуется использовать сайт https://regexr.com/
Текст ошибки	Если указан, то выводится в чат пользователю после получения ошибочных данных
Сценарий в случае ошибки	Сценарий, к которому выполняется переход после получения ошибочных данных



Для простых проверок введенного значения рекомендуется использовать интегрированную проверку в соответствии с регулярными выражениями. Для сложных проверок (в том числе требующих обращений к внешним системам), проверка реализуется в сценарии, используемом при успешном получении данных

3.2.16 Условия применения

Условия применения – это список условий, при которых сценарий или команда срабатывает. Сценарий/команда срабатывает в случае выполнения всех условий применения (логический принцип И).

Для настройки условия необходимо выбрать **тип проверки**:

Тип проверки
 Переменная

Реквизит
 Реквизит

Условие
 Равно

Значение
 Значение

Сохранить Отмена

Рисунок 3.23 Условие применения

Параметры условий в зависимости от Типа проверки:

Поле	Описание
Переменная	
Реквизит	Переменная контекста. Может быть использована как переменная бота, так и системная переменная.
Условие	Равно/Не равно/Включает
Значение	Значение переменной для проверки условия
Последняя тема	
Условие	Равно/Не равно/Включает
Значения	Выбор из списка имеющихся в боте тем
История сценариев	
Условие	Равно (совпадает со списком) /Не равно/Включает
Значения	Список сценариев с выбором из списка сценариев текущего бота
Формула	
Значение	Формула на C# . Условие выполняется в случае, если формула возвращает true .
График работы	
Рабочий график	Для выбора доступны графики работы , настроенные в Sherlock Platform.
Не рабочее время	Флаг. Инвертирует выбранный график работы. Условие срабатывает всегда, кроме выбранного графика работы .

3.3 Настройка событий бота

3.3.1 Вкладка События

Вкладка События содержит **Фильтр событий** и **Список событий**, на которые реагирует бот:

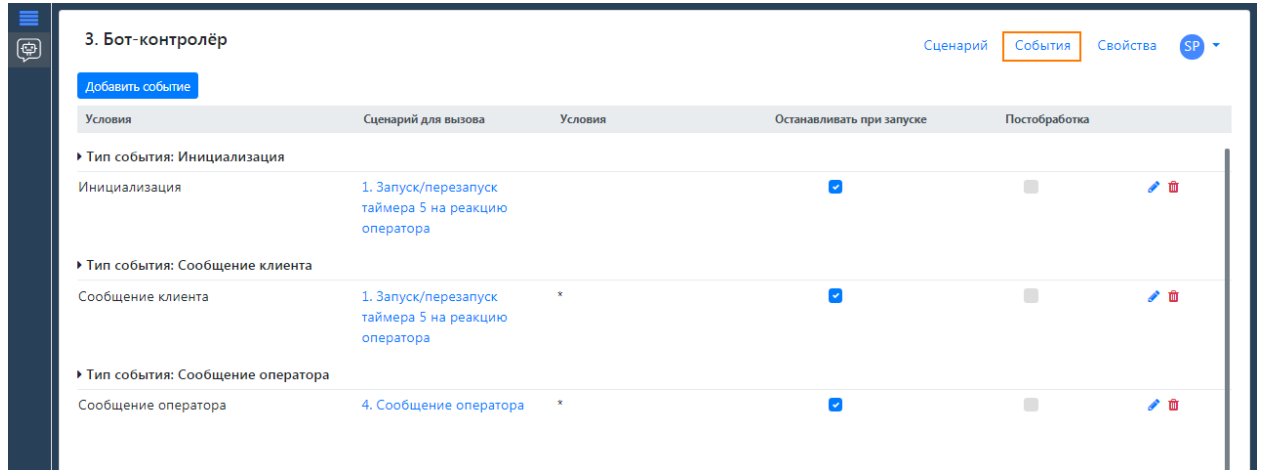


Рисунок 3.24 Окно настройки событий

В списке событий можно добавить или удалить событие. События в списке сгруппированы по типам событий. Порядок событий в списке можно менять с помощью Drag&Drop в пределах одного типа события.



Порядок используется, если в боте несколько событий одного типа. Обработка событий и вызов сценариев выполняется в порядке приоритета в списке. При этом, если для события в списке установлен признак **Останавливать при запуске**, то при срабатывании события последующие события не обрабатываются

3.3.2 Карточка события

Карточка события содержит **Общие свойства события** и **Дополнительные свойства события**, зависящие от типа события:

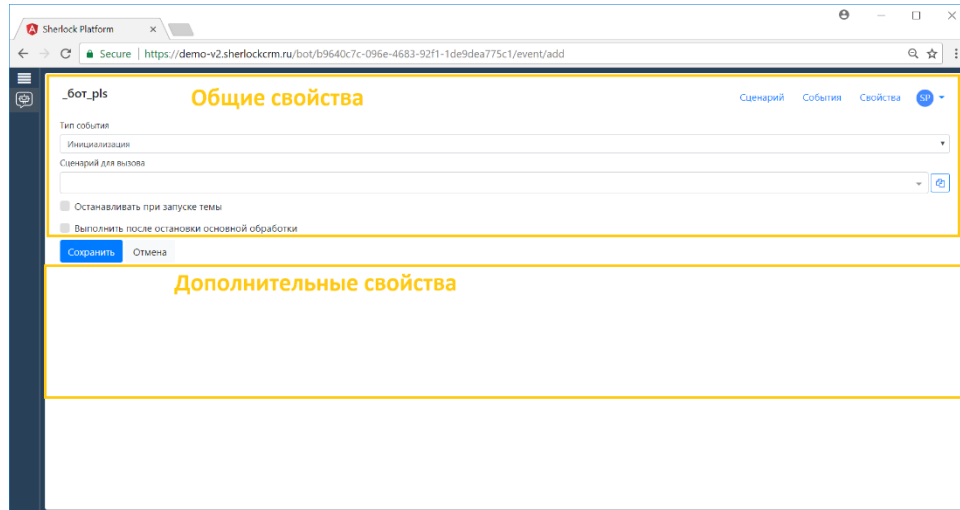



Рисунок 3.25 Окно настройки события

Общие свойства события:

Поле	Описание
Тип события	<p>Определяет источник события и, соответственно, механику работы системы.</p> <p>Подробнее по каждому типу см. ниже.</p>
Сценарий для вызова	<p>Сценарий бота, вызываемый по событию.</p>
Останавливать при запуске темы	<p>Останавливает дальнейшую проверку событий одного типа.</p> <div style="display: flex; align-items: flex-start;"> <div style="margin-right: 20px;">  </div> <div> <p><i>Используется, если в боте несколько событий одного типа. Когда нужно вызвать первую тему из нескольких, подходящую по условиям фильтров.</i></p> <p>Пример: <i>Клиент может спросить про регистрацию в личном кабинете или про личный кабинет в целом. В боте есть две темы – одна с пояснениями по регистрации в ЛК, другая переключает диалог в очередь техподдержки по работе в личном кабинете.</i> <i>Делаем два события. Первое, с фильтром *регистр*кабинет*, второе с фильтром *кабинет*. Первое событие с флагом остановки.</i></p> <p><i>Клиент пишет: «Как зарегистрироваться в личном кабинете». Срабатывает первое. И без флага остановки в первом событии сработало бы и второе. С флагом второе событие не срабатывает, то есть обработка будет выполнена корректно.</i></p> </div> </div>
Выполнить после остановки основной обработки	<p>Используется для выделения событий, которые должны быть обработаны и исполнены независимо от значения признака Останавливать при запуске темы</p>

3.3.3 Инициализация

Событие срабатывает один раз при старте бота. Дополнительных полей для события **Инициализация** нет.

3.3.4 Сообщение клиента

Иницируется при получении сообщения от клиента.

1. Встречающий бот

Тип события
Сообщение клиента

Сценарий для вызова

Останавливать при запуске темы

Выполнить после остановки основной обработки

Грамматика
Простой шаблон

Сохранить Отмена

Рисунок 3.26 Типы событий. Сообщение клиента

Дополнительные свойства события **Сообщение клиента**:

Поле	Описание
Грамматика	Вид шаблона для условия срабатывания события. <ul style="list-style-type: none">• Простой шаблон• Regular Expression
Шаблон	Условие срабатывания события в синтаксисе в соответствии с полем Грамматика . Может быть несколько. В этом случае событие срабатывает в случае выполнения любого условия, то есть по правилу ИЛИ.

3.3.5 Сообщение оператора

Иницируется при отправке сообщения оператором. Дополнительные свойства события **Сообщение оператора** идентичны событию **Сообщение клиента**.

3.3.6 Установка статуса

Иницируется при установке диалогу нового статуса.

The screenshot shows a configuration window titled '3. Бот-контролёр'. The 'Тип события' (Event type) is set to 'Установка статуса'. The 'Сценарий для вызова' (Call scenario) is '1. Запуск/перезапуск таймера 5 на реакцию оператора'. There are two checkboxes: 'Останавливать при запуске темы' (checked) and 'Выполнить после остановки основной обработки' (unchecked). The 'Статусы' (Statuses) field is highlighted with an orange border. At the bottom are 'Сохранить' (Save) and 'Отмена' (Cancel) buttons.

Рисунок 3.27 Типы событий. Установка статуса

Дополнительные свойства события **Установка статуса**:

Поле	Описание
Статусы	Поле выбора статусов из списка статусов настроенных в системе. Событие срабатывает при установке одного из выбранных статусов.

3.3.7 Постановка в очередь

Иницируется при постановке диалога в очередь.

The screenshot shows a configuration window titled '3. Бот-контролёр'. The 'Тип события' (Event type) is set to 'Постановка в очередь'. The 'Сценарий для вызова' (Call scenario) is '1. Запуск/перезапуск таймера 5 на реакцию оператора'. There are two checkboxes: 'Останавливать при запуске темы' (checked) and 'Выполнить после остановки основной обработки' (unchecked). The 'Очереди' (Queues) field is highlighted with an orange border. At the bottom are 'Сохранить' (Save) and 'Отмена' (Cancel) buttons.

Рисунок 3.28 Типы событий. Постановка в очередь

Дополнительные свойства события **Постановка в очередь**:

Поле	Описание
------	----------

Поле	Описание
Очереди	Поле выбора очередей из списка очередей, настроенных в системе. Событие срабатывает при постановке запроса в одну из выбранных очередей.

3.3.8 Передача оператору / начало диалога

Иницируется при передаче диалога оператору.

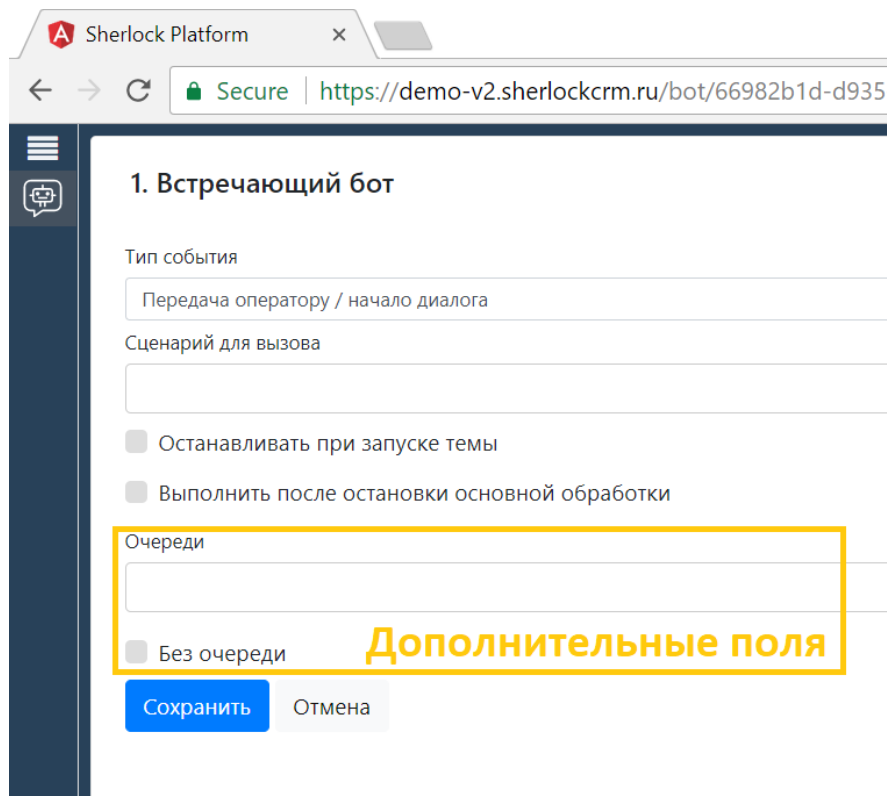


Рисунок 3.29 Типы событий. Передача оператору / начало диалога

Дополнительные свойства события **Передача оператору / начало диалога**:

Поле	Описание
Очереди	Поле выбора очередей из списка очередей настроенных в системе. Событие срабатывает при передаче диалога оператору в одной из выбранных очередей.
Без очереди	Событие срабатывает при передаче диалога оператору без очереди.

3.3.9 Внешнее событие

Иницируется при уведомлении Sherlock Platform внешней системой через web сервис. Дополнительные свойства события **Внешнее событие** идентичны событию **Сообщение клиента**.

3.3.10 Файл от клиента

Иницируется при получении файла от клиента. Дополнительные свойства события **Файл от клиента** идентичны событию **Сообщение клиента**.

3.3.11 Участник группы добавлен/удален

События добавления/исключения участника группы срабатывают для бота, включенного в группу Telegram (в других каналах поддержки события в текущей версии нет). Дополнительных полей для событий **Участник группы добавлен** и **Участник группы удален** нет.

Руководство по настройке: Боты	Редакция: 2.0
Документация по Sherlock Platform	

3.3.12 Закрытие окна чата

Событие срабатывает для канала **Чат на сайт** при закрытии пользователем окна чата. Дополнительных полей для события **Закрытие окна чата** нет.

3.3.13 Переход по реферальной ссылке

Событие срабатывает при переходе к боту с указанием реферальной ссылки в стартовых параметрах бота. Дополнительных полей для события **Переход по реферальной ссылке** нет.

3.3.14 Просмотр информации о канале

Событие срабатывает при открытии пользователем экрана с информацией о канале/боте в Viber (в других каналах поддержки события нет). Дополнительных полей для события **Просмотр информации о канале** нет.

Важно: При обработке данного события Sherlock Platform не запускает бота и, соответственно, не вызывает события инициализации бота. По требованиям Viber информация должна быть отправлена единым сообщением без задержки. Поэтому для события **Просмотр информации о канале** недопустимо использовать команду **Таймер** и не рекомендуется использовать длительные команды (например, **Внешний запрос**).

4 Подключение/запуск бота

Боты подключаются к запросу либо через очередь, либо через канал. Дополнительно бот может быть вызван командой другого бота (см. выше).

4.1 Подключение бота в канал

Бота можно подключить к каналу на карточке настроек канала (модуль Настройки => Каналы).

Профиль SherlockTest Viber аккаунт [Настроить](#)

Имя

Бот для обработки 1. Встречающий бот ▼

Тип запроса по умолчанию Выберите значение ▼

Создавать новое обращение в канале при открытых обращениях клиента в других каналах

Права по умолчанию [Добавить](#)

Роль доступа/сотрудник	Уровень доступа	
Администратор	Полный доступ	Редактировать Удалить

[Сохранить](#) [Отменить](#)

Рисунок 4.1 Карточка канала. Бот для обработки

4.2 Подключение бота в очереди

Вкладка **Боты** в окне настройке очереди (модуль Настройки => Справочники => Очереди) позволяет добавить к обработке очереди ботов.

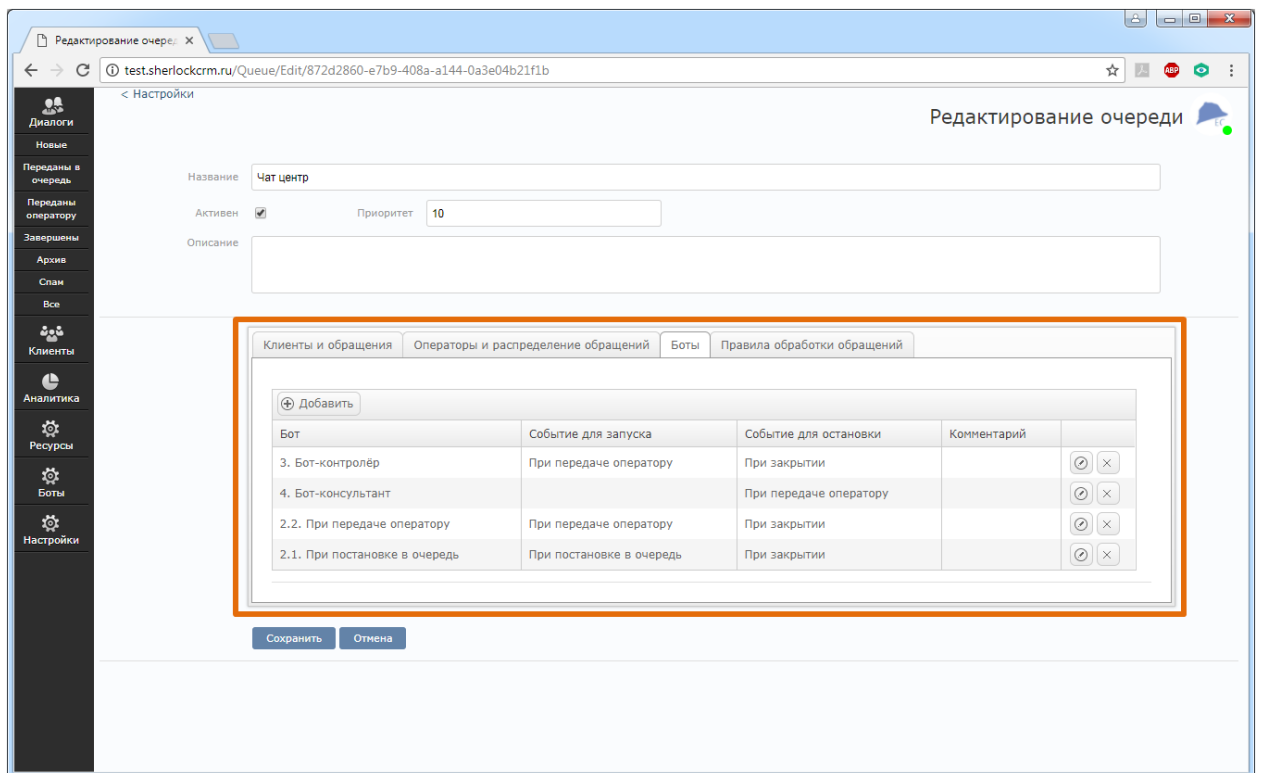


Рисунок 4.2 Карточка очереди. Боты

При этом для каждого добавленного бота задаётся событие, по которому бот подключается к диалогу и событие, по которому бот отключается от диалога.

Событие	Для запуска	Для остановки
При постановке в очередь	+	
При передаче оператору	+	+
При передаче в другую очередь	+	+
При закрытии	+	+

5 Переменные бота

Каждая команда бота может обращаться к переменным для реализации логики. Это могут быть переменные контекста текущего диалога, значения карточек клиента, обращения и системные переменные.

Переменные контекста сохраняются для каждого обращения в процессе работы. Переменные контекста предназначены для сохранения текущего состояния диалога и могут использоваться ботами в алгоритмах подготовки ответов клиентами. Переменные контекста формируются как автоматически (например, по полученному тексту), так и как результат выполнения определенных команд бота.

Для обращения к системным переменным используются predetermined имена переменных. Доступ к реквизитам клиента, обращения и других связанных данных идет через predetermined префикс. Перечень стандартных (системных) переменных контекста приведен в таблице 5.1

Таблица 5.1. Предопределенные системные переменные и префиксы

Имя переменной	Описание
#System.text#	Содержит текст последнего сообщения клиента
#SQL_Result#	Результаты последнего SQL запроса в XML формате
#SQL_RecordCount#	Количество записей, возвращенных последним SQL запросом
#SOAP_Result#	Результаты последнего SOAP запроса в XML формате
#REST_Result#	Результаты последнего REST запроса
#REST_StatusCode#	Код ответа сервера последнего REST запроса
#System.OperatorLive#	Количество операторов в сети (для очереди запроса)
#System.OperatorFree#	Количество операторов свободно (для очереди запроса)
#System.ResponsibleStatus#	Статус ответственного по запросу со значениями 0 – нет ответственного, 1 – не активен, 2 - активен
#Client.*#	для доступа ко всем стандартным реквизитам клиента. Смотрите пункт 5.1 для полного списка
#ClientExtra.*#	для доступа ко всем дополнительным реквизитам клиента. Точный список зависит от конфигурации приложения и задается в справочнике дополнительных реквизитов.
#Request.*#	для доступа ко всем стандартным реквизитам диалога. Смотрите пункт 5.2 для полного списка
#Request.Responsible.*#	для доступа к реквизитам ответственного за диалог. Смотрите пункт 5.3 для полного списка
#RequestExtra.*#	для доступа ко всем дополнительным реквизитам диалога. Точный список зависит от конфигурации приложения и задается в справочнике дополнительных реквизитов.
#Tags.*#	проверяет или устанавливает конкретный тег для клиента. Точный список зависит от конфигурации приложения и задается в справочнике тегов.
#LastMessage.*#	возвращает все реквизиты последнего сообщения. Смотрите пункт 5.4 для полного списка
#LastMessageIn.*#	возвращает все реквизиты последнего входящего сообщения. Смотрите пункт 5.4 для полного списка
#LastMessageOut.*#	возвращает все реквизиты последнего исходящего сообщения. Смотрите пункт 5.4 для полного списка
#ChatButtons_Value#	Значение, заданное для кнопки перехода к сценарию. Заполняется, если сценарий вызван пользователем бота по кнопке с указанным значением кнопки

При подстановке переменных контекста Sherlock автоматически обеспечивает маскирование символов в зависимости от точки подстановки. В случае, если маскирование символов использовать не надо, то переменную надо указывать как # $\$$ имя_переменной#

5.1 Реквизиты клиента

Название	Тип поля	Обязательное	Описание
ID	uniqueidentifier	Да	Первичный ключ
CreatedAt	datetime	Да	Дата и время создания
CreatedByID	uniqueidentifier	Нет	Автор записи
ModifiedAt	datetime	Да	Дата и время последнего изменения
ModifiedByID	uniqueidentifier	Нет	Автор изменения
Name	nvarchar(256)	Да	ФИО клиента
FirstName	nvarchar(256)	Нет	Имя клиента
LastName	nvarchar(256)	Нет	Фамилия клиента
PhotoUrl	nvarchar(1024)	Да	Ссылка на фото профиль
ClientTypeID	Uniqueidentifier	Да	Тип клиента по таблице tbl_ClientType
CellPhone	nvarchar(50)	Нет	Сотовый телефон
AdditionalPhone	nvarchar(50)	Нет	Дополнительный телефон
Age	int	Нет	Возраст
BirthDay	date	Нет	Дата рождения
CountryID	Guid	Нет	Ссылка на страну
CityTitle	nvarchar(128)	Нет	Город клиента
DeliveryAddress	nvarchar(1024)	Нет	Адрес доставки
Note	nvarchar(max)	Нет	Примечание к карточке клиента
GenderID	Int	Нет	Пол. 1 - женский, 2 - мужской
Email	nvarchar(100)	Нет	Email клиента
LanguageCode	nvarchar(5)	Да	Код языка

5.2 Реквизиты диалога

Название	Тип поля	Обязательное	Описание
ID	uniqueidentifier	Да	Первичный ключ
CreatedAt	datetime	Да	Дата и время создания
CreatedByID	uniqueidentifier	Нет	Автор записи
ModifiedAt	datetime	Да	Дата и время последнего изменения
ModifiedByID	uniqueidentifier	Нет	Автор изменения
ClientID	uniqueidentifier	Да	Ссылка на клиента
Number	Int	Да	Номер обращения
Date	datetime	Да	Дата обращения
RequestStatusID	uniqueidentifier	Нет	ID статуса обращения
Note	nvarchar(max)	Нет	Примечание к карточке обращения
LastMessageDate	datetime	Нет	Дата последнего сообщения
LastMessageID	uniqueidentifier	Нет	ID последнего сообщения

Название	Тип поля	Обязательное	Описание
LastMessageText	nvarchar(max)	Нет	Текст последнего сообщения
LastMessageStatusID	int	Нет	Смотри описание tbl_Message
LastMessageDirectionID	int	Нет	
LastMessageCategoryID	int	Нет	
AuthorUserID	uniqueidentifier	Нет	Автор диалога
RequestCategoryID	int	Да	Категория обращения. Предусмотрено для будущих расширений Всегда PersonalDialog = 1
LastMessageAuthorPhotoUrl	nvarchar(1024)	Нет	Ссылка на фото профиля автора последнего сообщения
NewMessageID	uniqueidentifier	Нет	
RequestTypeID	uniqueidentifier	Нет	Ссылка на справочник типов запросов
ResponsibleID	uniqueidentifier	Нет	Ссылка на ответственного
QualityGrade	int	Нет	Оценка диалога
ClosedAt	datetime	Нет	Дата закрытия
IsFileAttach	bit	Да	1, если для последнего сообщения есть вложение
QueueID	uniqueidentifier	Нет	Текущая (последняя) очередь диалога
RequestPriorityID	uniqueidentifier	Нет	Приоритет обращения (ссылка на справочник приоритетов)
LastMessageInID	uniqueidentifier	Нет	ID последнего входящего сообщения
LastMessageOutID	uniqueidentifier	Нет	ID последнего исходящего сообщения
LastMessageUserProfileID	uniqueidentifier	Нет	Ссылка на профиль пользователя последнего сообщения
SocialNetworkMessengerId	uniqueidentifier	Да	Ссылка на тип мессенджера, с которым связано обращение
ChannelType	nvarchar(250)	Нет	Тип канала, с которым связано обращение
LastMessageAttachmentName	nvarchar(250)	Нет	Имя вложения к последнему сообщению

5.3 Реквизиты оператора

Название	Тип поля	Обязательное	Описание
ID	uniqueidentifier	Да	Первичный ключ
CreatedAt	datetime	Да	Дата и время создания
CreatedByID	uniqueidentifier	Нет	Автор записи
ModifiedAt	datetime	Да	Дата и время последнего изменения
ModifiedByID	uniqueidentifier	Нет	Автор изменения
Email	nvarchar(256)	Да	Email (логин) пользователя
Password	nvarchar(256)	Да	Пароль

Название	Тип поля	Обязательное	Описание
IsActive	bit	Да	1, если пользователь активен
FullName	nvarchar(256)	Да	Полное имя
CellPhone	nvarchar(50)	Нет	Сотовый телефон
OfficePhone	nvarchar(50)	Нет	Офисный телефон
PhotoUrl	nvarchar(1024)	Да	Ссылка на фото профиля
AccessRoleID	uniqueidentifier	Да	Ссылка на роль доступа пользователя
OperatorStatusID	uniqueidentifier	Да	Ссылка на текущий статус оператора
IsBot	bit	Да	1, если это запись бота
Prefix	nvarchar(250)	Нет	
Description	ntext	Нет	
IsSendFileAfterAttach	bit	Да	
IPAddress	nvarchar(50)	Нет	
SessionID	nvarchar(250)	Нет	
LastActivityDate	datetime	Нет	Дата и время последнего действия в системе
Position	nvarchar(max)	Нет	Должность пользователя
UserHash	nvarchar(100)	Нет	

5.4 Реквизиты сообщения

Название	Тип поля	Обязательное	Описание
ID	uniqueidentifier	Да	Первичный ключ
CreatedAt	datetime	Да	Дата и время создания
CreatedByID	uniqueidentifier	Нет	Автор записи
ModifiedAt	datetime	Да	Дата и время последнего изменения
ModifiedByID	uniqueidentifier	Нет	Автор изменения
RequestID	uniqueidentifier	Нет	Ссылка на диалог
UserID	uniqueidentifier	Нет	Ссылка на пользователя
ClientID	uniqueidentifier	Нет	Ссылка на клиента

Название	Тип поля	Обязательное	Описание
MessageStatusID	Int	Да	Статус сообщения [Description("Запланирован")] Planned = 1, [Description("Не прочитано")] NotRead = 2, [Description("Прочитано")] Read = 3, [Description("В очереди")] Queued = 4, [Description("Ошибка")] Error = 5, [Description("Удален")] Deleted = 6, [Description("Send")] Send = 7,
Text	ntext	Да	Текст сообщения
MessageDirectionID	int	Да	Направление сообщения [Description("Входящее")] In = 1, [Description("Исходящее")] Out = 2 Примечание: Системные сообщения считаются исходящими
MessageCategoryID	int	Да	Категория сообщения [Description("Личное сообщение")] PersonalMessage = 1, [Description("Комментарий")] Comment = 2, [Description("Кнопки")] Button = 3, [Description("Статус диалога")] DialogStatus = 4
PlannedDate	datetime	Нет	Плановая дата
SendDate	datetime	Нет	Дата фактической отправки
SocialNetworkMessengerID	uniqueidentifier	Да	Ссылка на тип мессенджера, с которым связано сообщение
ExternalMessageID	nvarchar(250)	Да	Идентификатор сообщения во внешней сети
ExternalAuthorID	nvarchar(100)	Да	Профиль автора во внешней сети
ExternalRecipientID	nvarchar(100)	Да	Профиль получателя во внешней сети
PreviousMessageDelay	int	Нет	Задержка в миллисекундах от предыдущего сообщения
Draft	bit	Да	1 для черновика, 0 для обычного сообщения
IsImportant	bit	Да	1 для важного сообщения, 0 для обычного сообщения

Название	Тип поля	Обязательное	Описание
MessageUserCategoryID	uniqueidentifier	Нет	Ссылка на категорию сообщений
ActualSendDate	DateTime	Нет	Время фактического получения сообщения для ВКонтакте по данным Sherlock Platform. При этом в SendDate указана дата отправки сообщения клиентом.
PreviousMessageDelayType	int	Нет	Определяет случай для которого измеряется задержка 1 - первое сообщение 2 - ответ клиенту 3 - ответ клиента 4 - пауза между сообщениями клиента или пользователя