

# 1С:Предприятие 8

## Платформа интеграции 1С с чат-ботами

Версия 9.02.00

Руководство оператора, разработчика

Г. Иннополис

Разработчики «Bots 1С»

2024 г.

# ОГЛАВЛЕНИЕ

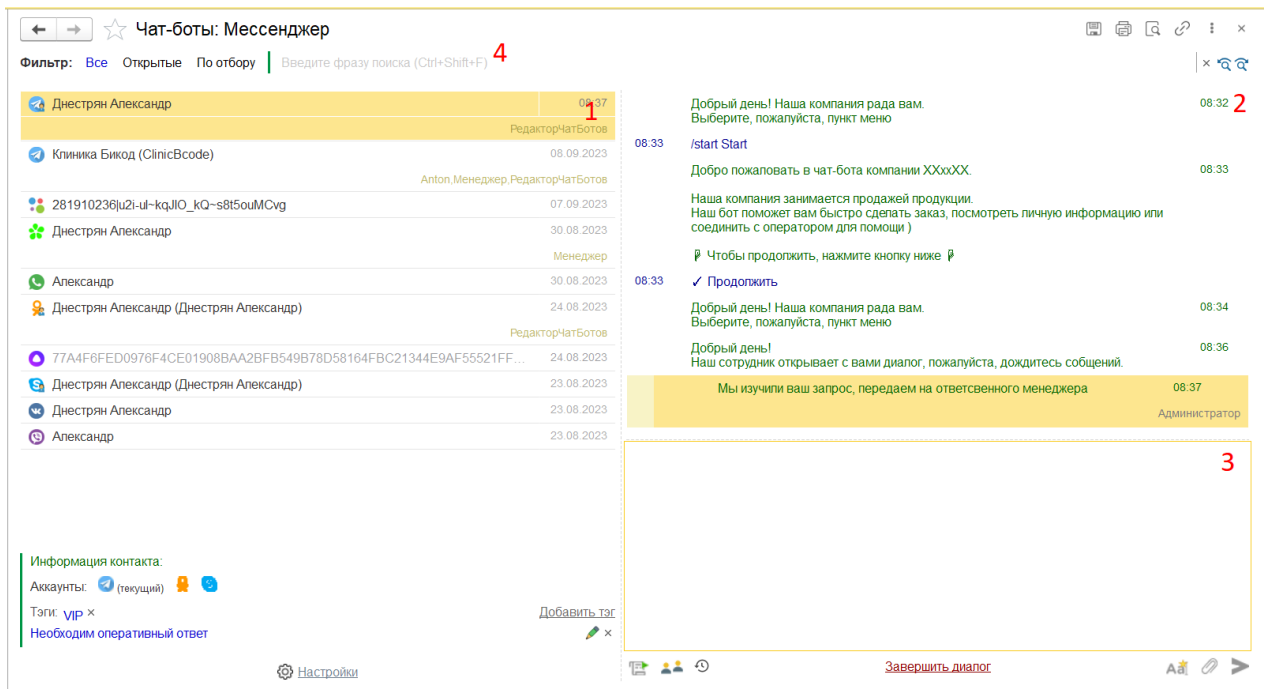
<b>Руководство оператора.....</b>	<b>3</b>
Мессенджер.....	3
Индивидуальные настройки .....	7
<b>Руководство настройки бизнес логики (сценариев поведения) .....</b>	<b>9</b>
6.1. Общие настройки.....	9
6.2. Создание и настройка чат-бота .....	11
6.3. Графический редактор сценариев поведения чат-бота .....	13
<b>Руководство по написанию методов (кодинг).....</b>	<b>22</b>
4.1. Подключаемые методы.....	22
4.2. API .....	26

# РУКОВОДСТВО ОПЕРАТОРА

## Мессенджер

Мессенджер – это форма, которая позволяет сотруднику просматривать историю диалогов клиента с чат-ботом, вести диалог с клиентом напрямую, производить рассылку сообщений. Интерфейс представлен в стандартном, удобном виде, где с левой стороны расположены контакты, а с правой – переписка с клиентом.

Рассмотрим по интерфейсу подробнее:



1. **Список контактов** – в данном списке отображаются все клиенты, которые когда-либо взаимодействовали с чат-ботом, даже если не входили в диалог с сотрудником.

В каждой строке контакта отображается имя контакта, которое может принимать несколько значений, первоначально, имя получается как оно представлено в аккаунте, в случае необходимости данное имя можно откорректировать (эта возможность описана ниже), и в случае, если контакт соединен с контрагентом/партнером или другой ссылкой владельцем, то будет отображаться наименование владельца по ссылке.

Если контакт соединен с владельцем (контрагентом/партнером и т.д.) то на иконке чат-бота появляется маленькая иконка с контрагентом:

Правее имени контакта отображается дата или время последнего взаимодействия клиента с чат-ботом.

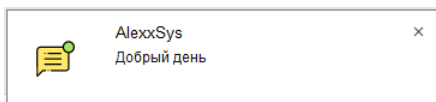
Немного ниже отображаются сотрудники к которым прикреплен клиент, это может быть как один сотрудник или несколько, а так же может быть проставлено \*, что означает что клиент направлен всем сотрудникам без ограничения. Для ответственного, у своих клиентов поле остается пустое.

Если клиент отписался от чат-бота, он будет отображен как неактивный:  Александр 10.11.2020

Если от клиента есть непрочитанные сообщения – тогда строка будет жирным шрифтом:

 AlexxSys 10:57

На каждое сообщение от клиента, будет отображаться оповещение, вне зависимости от того активен ли мессенджер, и активно ли сама 1С. Оповещение, если не прочитано, повторяется каждую минуту.

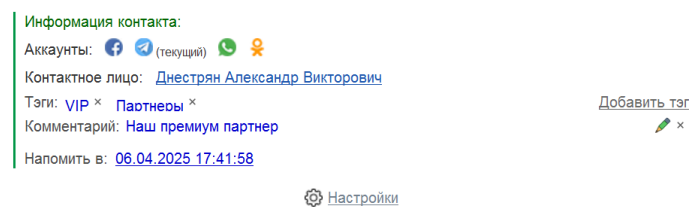


Для прочтения необходимо перейти на контакт или нажать на оповещение – тогда автоматически активизируется мессенджер и нужный контакт с диалогом. Если контакт активен – тогда оповещение не будет отображено.

При нажатии на контакте правой кнопки мышки, доступно контекстное меню, из которого стоит остановиться на возможностях:

1. **Добавить** - Данная кнопка позволяет добавить новый контакт, который ранее не писал. Но надо понимать, что контакт чат-бота идентифицируется не номером телефона, а специальным внутренним идентификатором. Т.е. что б добавить контакт надо знать его идентификатор, и в большинстве системах необходимо что б он был подписан на вашего бота. Исключения составляют некоторые мессенджеры, допустим WhatsApp – так как он работает как обычный аккаунт, контакт можно добавить, зная его номер телефона. В этом случае необходимо обязательно с начала выбрать системы чат-ботов, а после ввести представление и номер телефона.
2. **Изменить** - Данная кнопка предоставляет возможность изменить некоторые данные контакта – имя, номер телефона и привязку к владельцу (контрагент, партнер и т.д.). Форма открывается такая же как и в предыдущем меню при добавлении контакта.

Ниже контактов отображается информация по контакту:




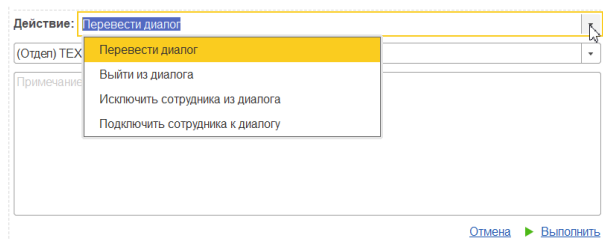
1. **Аккаунты** – здесь отображаются все аккаунты контакта чат бота (если он авторизовался и прикрепился к контактному лицу). При нажатии будет осуществлен переход на данный контакт чат-бота
  2. **Контактное лицо** – контактное лицо которое привязано к контакту чат-бота. В зависимости от начальных настроек здесь может быть контрагент, лид, контакт лида, сотрудник или любой другой справочник.
  3. **Тэги** – Дает возможность установить несколько тэгов
  4. **Комментарий** – Дает возможность установить небольшой комментарий на контакте
  5. **Напомнить** – Здесь отображается информация об установленном напоминании на контакт, так же можно изменить или удалить его, для этого надо нажать на время напоминания.
2. **История сообщений** – все сообщения входящие и исходящие сообщения взаимодействия с клиентом. На каждое сообщение указано время, и при смене даты – ее представление.
- В случае если был передан мультимедиа файл – то будет отображено имя файла и, при нажатии, файл будет открыт в внешней программе (просмотр картинок, музыки, видео, документов и т.д.). Все файлы хранятся в 1С без ограничения по времени хранения, их можно получить за любой период.
- Если сообщения написаны в рамках диалога клиента с сотрудником – то они подсвечиваются вертикальной желтой линией справа. Так же для каждого сообщения указывается сотрудник который его написал.
3. **Область нового сообщения** – В данной области содержатся элементы, необходимые для взаимодействия с клиентом, т.е. отправки сообщений, файлов, открытия диалога и т.д.
- Сотрудник может отправить как сообщение, так и любые мультимедиа данные клиенту. **Для отправки текстового сообщения** этого необходимо его написать в поле и нажать кнопку отправки или сочетание клавиш Ctrl+Enter. **Для отправки файлов**, необходимо нажать на кнопку со скрепкой, выбрать нужный тип файлов (картинка, аудио, видео, файл) и сам файл, после выбора файла он сразу отправится. Отправка файла и текста одновременно не поддерживается.
- Отправляться сообщения могут в 2х разных вариантах:

1. Без открытия диалога – Если клиент не находится в режиме диалога с сотрудником, при отправке сообщения клиенту, оно отправится как рассылка, т.е. без открытия диалога. После отправки единичного сообщения управление перейдет боту, т.е. будет активирован сценарий главного меню, и бот будет ожидать ответ клиента.
2. В режиме диалога с сотрудником – Если клиент находится в режиме диалога с сотрудником, т.е. автоматические ответы бота отключены и все сообщения направляются сотруднику, то сообщение отправится в этом режиме, после отправки управление боту передано не будет, т.е. доступно будет дальнейшая переписка.

Для корректной работы открытия диалога с сотрудником в настройках, администратору необходимо проставить сценарий диалога, в нем должен быть достижимый шаг «диалог с сотрудником». Достижимый, это означает что без какого либо ожидания ответа клиента этот шаг выполнится

Так же, есть некоторые команды, для управления клиентом:

1.  - Перевод диалога другому сотруднику. При нажатии на данную кнопку, появится данные для выбора другого сотрудника (пользователя базы, с правом открытие мессенджера чат-ботов) и отделов (определенные в настройках, в разделе Общие -> Отделы операторов).



Для каждого пользователя добавляется статус: Онлайн / Не в сети, для отдела такой статус не добавляется. Та же можно дополнительно ввести текст сообщения в поле «Примечание», который будет отображен сотруднику при переводе диалога.




Возможные действия:

**Перевести диалог** – данный диалог полностью переводится указанному сотруднику, при этом пропадает у текущего сотрудника

**Выйти из диалога** – Диалог пропадает у данного сотрудника, сообщения больше не будут ему направляться. Имеет смысл при групповом общении

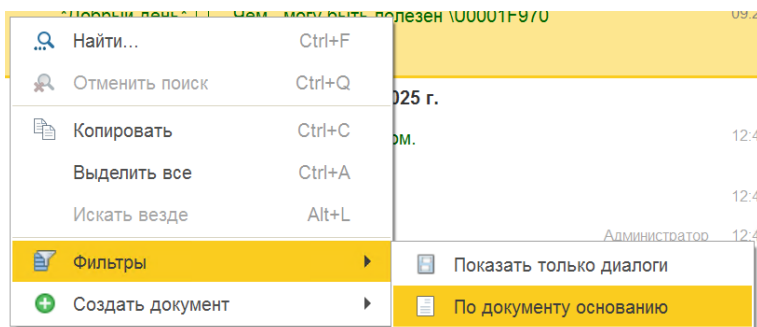
**Исключить сотрудника из диалога** – Исключает указанного сотрудника из диалога

**Подключить сотрудника к диалогу** – подключает сотрудника или группу к диалогу, делая диалог групповым

2. **Начать диалог** - Нажав на данную кнопку, сотрудник открывает диалог с клиентом для дальнейшего общения.
- Принять диалог** - Нажав на данную кнопку, сотрудник примет диалог на себя, для дальнейшего общения. При этом данный диалог скроется у всех остальных сотрудников.
- Завершить диалог** - Завершение режима диалога клиента с сотрудником. При нажатии диалог будет завершен, и будет передано управления автоматическим ответам чат-бота (чат бот завершит элемент диалога, и дальше продолжит выполнять следующие шаги сценария).
- Aa** - При нажатии данной кнопки – появляется выпадающий список с быстрыми фразами. Быстрые фразы добавляются индивидуально для каждого сотрудника в настройках мессенджера – раздел «Индивидуальные» - Быстрые сообщения.
-  - Запуск сценария у клиента, по инициативе сотрудника. Допустим можно запустить опрос, и в режиме массовой отправки какую-то акцию и ведения дальнейших шагов по этой акции, которые прописаны в сценарии. Для того что бы сценарий появился в списке, в сценарии необходимо включить галочку «Разрешить запуск из мессенджера»
-  – Прикрепить файл для отправки. Можно прикрепить сразу несколько файлов, их название отобразится чуть выше, после написать сообщение (опционально) и отправить.
-  - Используется для отправки сообщений

Для массовой отправки сообщений, можно выделить нужные контакты через Ctrl+КликМышкой, и написать сообщение или выбрать файл. При такой отправке, режим работы с клиентом (в режиме бота / в режиме диалога с сотрудником) не меняется, т.е. если клиент был в режиме диалога, ему отправляется сообщение и остается в этом же режиме, если клиент был в автоматическом режиме работы с ботом – после отправки сообщения переходит в главное меню.

Так же есть контекстное меню, для этого необходимо нажать правой кнопкой мыши на область сообщений.



Через данное меню возможно:

1. **Показать только диалоги** - Установить / снять фильтр на отображения только тех сообщений которые писались в режиме диалога с менеджером. Для этого просто нажмите на нужный пункт меню
2. **По документу основанию** – В данном пункте можно выбрать документ основания и все сообщения отфильтруются только теми, которые привязаны к документу. Так же, если в этом режиме писать сообщения – они все будут прилинкованы к документу.
3. **Создать документ** – Если выделить несколько сообщений и выбрать данный пункт, выбрав нужный тип документа – то будет создан документ и сообщения будут прилинкованы к нему. К примеру на основании переписки создаем документ заказ товара.

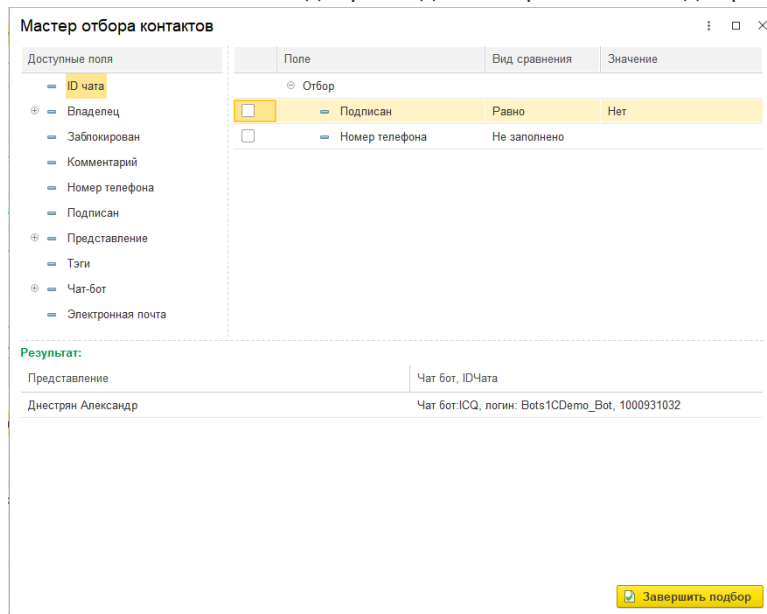
#### 4. **Фильтр списка контактов:** Фильтр: Все [Открытые](#) [По отбору](#)

Данный фильтр позволяет ограничить отображение контактов, убрав лишние:

**Все** – Показывает все контакты, доступные сотруднику

**Открытые** – Показывает только контакты, которые в данный момент находятся в режиме диалога с сотрудником, т.е. ожидают ответ от сотрудника.

**По отбору** – По данной команде, можно узко настроить отображение контактов по любым условиям, полям. Допустим кто связан с контрагентом, подписан, по полу, группе владельца, и любым другим различным полям. Может использоваться допустим для выборки клиентов для рассылки.




При выборе полей отбора или изменении – обновления результирующей таблицы с контактами происходит автоматически.

После нажатия кнопки завершить отбор – происходит сохранение настроек отбора с автоматической загрузкой при следующем открытии, и в мессенджере список отфильтруется с учетом выбранных отборов.

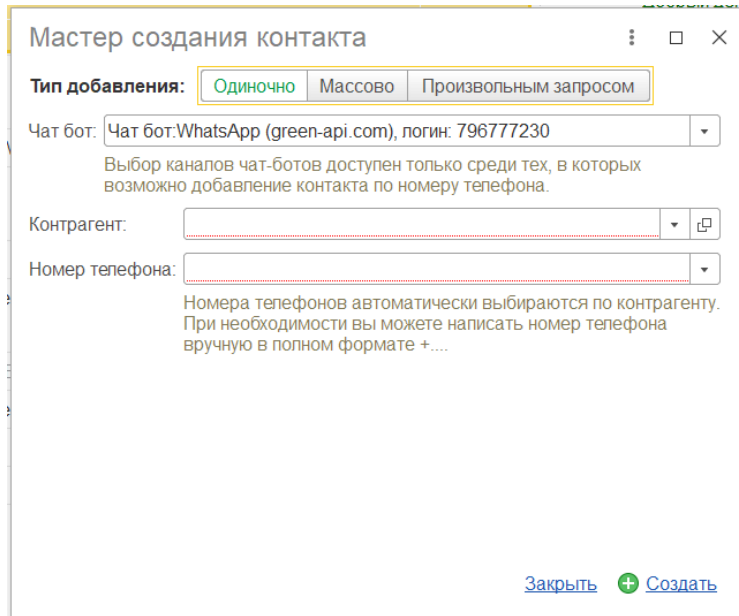
#### 5. **Поиск** – данное поле позволяет сделать глобальный поиск как по списку контактов, так и по сообщениям.

Фильтр: Все [Открытые](#) [По отбору](#)

После ввода фразы необходимо нажать Enter, после чего список контактов отфильтруется, и выделится

иское сообщение. Для перехода по найденным позициям используйте кнопки . Если необходимо очистить поиск – нажмите крестик.

**6. Массовое добавление контактов** – Есть возможность массового добавления контактов чат ботов для каналов, которые это поддерживают (WhatsApp). Для этого на списке контактов вызовите контекстное меню и нажмите кнопку «Добавить». Откроется следующее окно:



Возможны несколько типов добавления контактов:

**Одиночно:** Для добавления одного контакта. Здесь необходимо указать контрагента (опционально), и выбрать или написать номер телефона и нажать кнопку «Создать»

**Массово:** При необходимости добавить сразу множество контактов можно выбрать данный пункт. Для добавления необходимо нажать кнопку «Подобрать контакты», после чего выбрать справочник, и можно через shift выделить множество контактов для добавления. Они все перенесутся в таблицу и автоматически подхватится номер телефона, при необходимости его можно изменить. После нажать кнопку «Создать»

**Произвольным запросом:** Это такое же массовое добавление, только контакты выбираются не визуально, а написав SQL запрос, здесь вы получаете гибкость и можете использовать различные условия или логику.

## Индивидуальные настройки

В мессенджере, снизу есть кнопка настроек. Которая позволяет провести индивидуальные настройки оператора. Рассмотрим их подробнее.

## Индивидуальные

Общие

Лицензия

Администрирование

## Индивидуальные настройки

 Звуковое оповещение при получении сообщений

При включенной галочке, на полученное сообщение от клиента - будет звуковое оповещение. Функция экспериментальная, работа не гарантируется (для работы необходимо проверить разрешение у браузера на проигрывание звука). Необходим перезапуск.

 Мессенджер на рабочем столе

Для применения изменений, после сохранения требуется перезапустить 1С

Имя сотрудника: 

В чатах использовать следующее представление (имя) сотрудника

## Быстрые сообщения

Приветствую!

Конечно, пожалуйста, пришлите данные вашей организации

Мы анализируем ситуацию, позже с вами свяжемся

 Удалить  Добавить

**Звуковое оповещение при получении сообщений** – При включении галочки, на получение сообщения от клиента – будет звуковое оповещение..

**Мессенджер на рабочем столе** – Если необходимо отображать мессенджер на начальной странице с автоматическим открытием при запуске конфигурации, то необходимо поставить данную галочку. Эта настройка индивидуальная для пользователя.

**Имя сотрудника** – в чатах будет использовать следующее представленное имя сотрудника

**Быстрые сообщения** – Здесь указываются быстрые сообщения, которые в сотрудник может выбрать нажатием одной кнопки при общении с клиентом через Мессенджер в 1С.

# РУКОВОДСТВО НАСТРОЙКИ БИЗНЕС ЛОГИКИ (СЦЕНАРИЕВ ПОВЕДЕНИЯ)

В данном разделе рассматривается работа с платформой пользователем или разработчиком. Будет рассмотрено настройка системы, добавление и настройка чат-бота, разработка сценариев работы.

## 6.1. ОБЩИЕ НАСТРОЙКИ

Первоначально, рекомендуется настроить параметры системы. Открыть форму настроек можно пользователю с правом «Чат-боты: Добавление и изменение настроек» из мессенджера, нажав кнопку «Настройки». В форме есть несколько разделов:

**Лицензия** – в данном разделе можно активировать или переактивировать лицензию.

Чат-боты: Настройка

Индивидуальные Лицензионные данные

Общие Тип лицензии: Тестовая (для разработчика)

**Лицензия**

Администрирование

ВНИМАНИЕ:  
- Рабочую лицензию можно получить только 1 раз (или переполучить на те же данные)  
- Тестовая лицензия для разработчиков, без ограничения в количестве получений, ограничена 1-м активным чат-ботом и 2-мя контактами для каждого чат-бота

Лицензия ИНН:

№ Договора (счета):

Статус лицензии: Активна [Изменить лицензию](#)

Тип лицензии: work, ограничения: чат-ботов: 8

Версия: 7.0.0 [Активировать лицензию](#)

**Тип лицензии** – Тестовая или рабочая.

Тестовая – это лицензия для разработчиков, можно получить неограниченное кол-во раз, но она ограничена в функциональности: только 1-н активный чат бот и не более 3х контактов написавших боту (если эти условия превышены, подсистема перестанет функционировать, необходимо пометить на удаления или удалить лишние объекты, после бот начнет функционировать снова).

Рабочая – лицензия для ПРОД среды, возможно получить только один раз или переполучить на тот же путь к вэбхуку (путь к 1С).

Для новой базы необходимо выбрать тестовую лицензию, когда все запустится на тестовой, уже переактивировать на рабочую, что бы избежать ошибки неправильного указания пути к вэбхуку.

**ИНН** – Код ИНН лицензиара из счета на оплату или, при наличии, договора.

**№ Договора (счета)** – Указывается номер с счета на оплату или, при наличии, договора.

**Путь к вэбхуку 1С** – Путь, вида `https://{ДоменноеИмяСервера1С}/{ИмяПубликации}`, по которому доступна данная система 1С. Для исключения вероятности ошибки, в этом поле можно только выбрать из списка значения, которые указаны в карточке чат-ботов.

**ВНИМАНИЕ:** Лицензия привязывается к пути вэбхука и изменить ее будет невозможно! Именно поэтому необходимо указать корректно с использованием рекомендаций.

**Индивидуальные** – Рассматривается в руководстве оператора

**Общие** – В данном разделе содержатся настройки работы платформы.

Чат-боты: Настройки: Чат-боты: Настройка

Индивидуальные Диалог с сотрудником (демо)

**Общие** Сценарий, который будет запускаться при диалоге, инициатором которого является сотрудник. Если сценарий не указан - автоматически будет создан служебный сценарий диалога

Лицензия

Администрирование VIP Конкурент | Тэги для проставления на контактах... | x

Список доступных тэгов для контактов

Отделы операторов

Имя отдела	Операторы	Руководители
СПА	Менеджер,Партнер,Админи...	Администратор
ОПГ	Менеджер,Менеджер2	Администратор

Удалить + Добавить

Настройки пользователей

Пользователь	Видимость всех клиентов	Режим "Открытые диалоги"
Администратор	<input type="checkbox"/>	<input type="checkbox"/>
Менеджер	<input type="checkbox"/>	<input type="checkbox"/>
Партнер	<input type="checkbox"/>	<input type="checkbox"/>

Удалить + Добавить

Версия: 9.01.00

Сохранить и закрыть

**Сценарий диалога:** указывается сценарий, который будет запускаться, при открытии диалога, инициализированного сотрудником. К данному сценарию предъявляются некоторые требования: при его выполнении должен выполняться шаг с элементом «диалог с сотрудником». Допускается, чтобы до этого были шаги с элементами «Ветвление» и «Действие» с галочкой «Не ожидать ответа». Т.е. запуск шага, должен привести к открытию диалога с клиентом (пример можно посмотреть в сценариях поставки). **Если его не указать – оператор не сможет открывать диалоге с клиентом со своей стороны.**

**Тэги:** это список с возможными тэгами, которые можно будет присвоить контакту чат-бота.

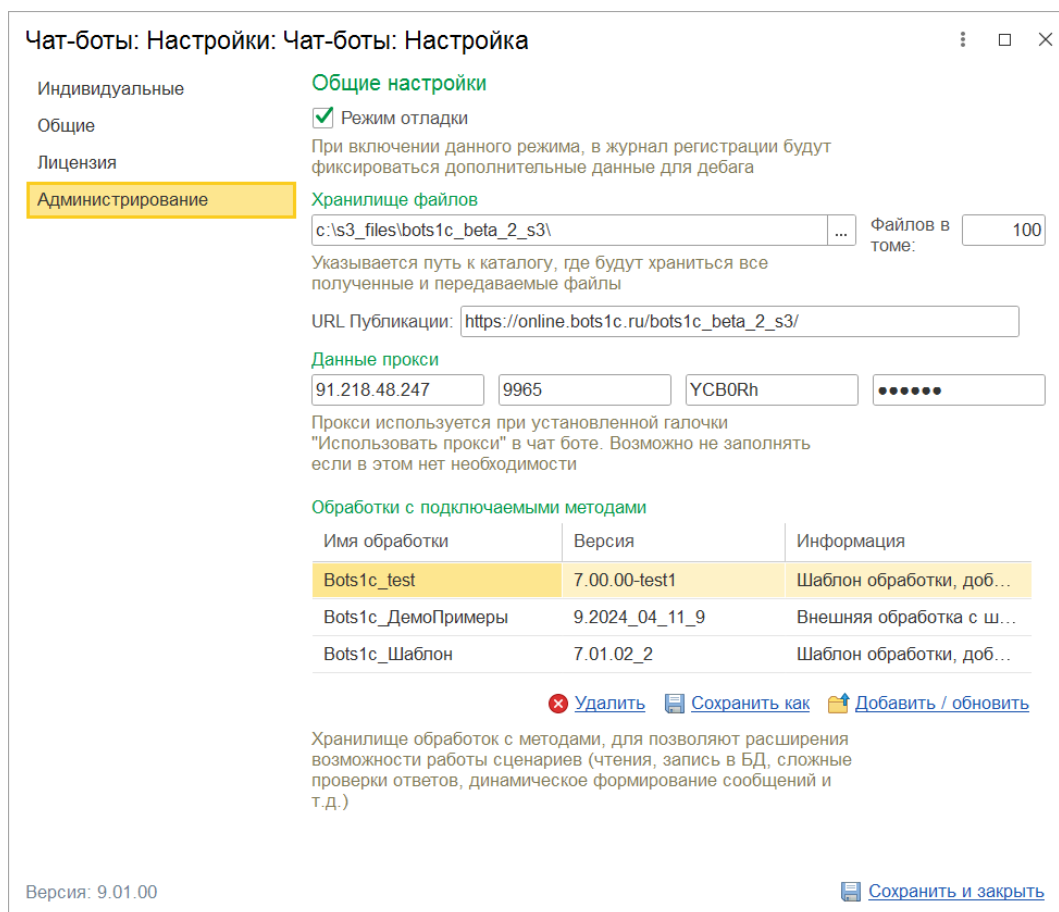
**Отделы операторов:** это список отделов сотрудников, для переключения диалога целиком на отдел, а так же для указания руководителей, которые будут видеть все диалоги своих сотрудников. В поле вводится имя отдела, и логины Информационной Базы пользователей сотрудников. В полях Операторы и Руководители могут быть несколько сотрудников, разделяются запятыми без пробелов.

**Отделы операторов:** В таблице можно указать список отделов и операторов/руководителей отдела, для внутренней маршрутизации оповещений от клиентов и/или передача диалога. Для выбора доступны пользователи с ролью ИспользованиеМессенджера или же ИзменениеЧатБотов

**Настройки пользователей:** В таблице указываются настройки / разрешения для пользователей: Видимость всех клиентов – при установленной галочке, данный пользователь будет видеть всех клиентов чат-ботов, вне зависимости к кому клиент прикреплен.

Режим «Открытые диалоги» - данный параметр ограничивает видимость сотрудником клиента только когда клиент открыл диалог с сотрудником, в других случаях клиент будет скрыт из списка

**Администрирование** – Этот раздел содержит настройки, которые не влияют на поведение платформы, но необходимы для администрирования.



**Режим отладки:** при выставлении флажка, в журнал регистрации будет фиксироваться дополнительная информация о принимаемых и отправляемых данных. Рекомендуется включать только для отладки каких-либо проблем.

**Хранилище файлов:** В данной группе указываются настройки каталоге где будут храниться полученные и отправляемые файлы, а так же имя публикации для отправки файлов. Заполнение данных пунктов описано в руководстве Администратора.

**ВНИМАНИЕ:** Данные настройки завязаны с публикацией, в некоторых случаях их так же следует изменять (к примеру при изменении имя публикации, пути к каталогу), иначе клиенты не смогут получить файлы






**Обработки с подключаемыми методами:** В данный список загружаются обработки, в которых содержатся методы расширения сценариев, т.е. которые будут вызваться из сценариев. Рекомендуется загружать обработки, при наличии, в штатную подсистему «Дополнительные обработки и отчеты». В случае, если этой подсистемы нет – тогда использовать текущий список.

## 6.2. СОЗДАНИЕ И НАСТРОЙКА ЧАТ-БОТА

Чат-боты добавляются в соответствующем справочнике «Чат-боты». В форме списка отображается представление (тип + логин) и комментарий. Предварительно, необходимо создать сам аккаунт чат-бота во внешней системе, описание в [Раздел 2: Руководство по созданию чат-ботов.](#)

Для добавления нового чат-бота, в форме списка необходимо нажать кнопку «подключить» и выбрать необходимый мессенджер/соц.сеть. Далее в форме создания элемента заполняем поля:

## Чат-боты


Ссылка	Ссылка на чат-бота	Комментарий
 Чат бот: Facebook, логин:		
 Чат бот: Skype, логин: Bots1C_Tes...	<a href="https://join.skype.com/bot/7e...">https://join.skype.com/bot/7e...</a>	
 Чат бот: VKontakte, логин:		
 Чат бот: Одноклассники, логин:	<a href="ok.ru/messages/group/70000...">ok.ru/messages/group/70000...</a>	
 Чат бот: Яндекс 'Алиса' логин:		

Записать и закрыть    Записать    Еще ▾

▼ Параметры подключения к серверу ботов


Сервер:  Отключен    Активен

Токен...:  ?



Логин:     Имя:     Связь установлена 

▼ Параметры подключения к 1С


Путь:

Путь к 1С:     Связь с 1С доступна 

▼ Разрешенные сценарии

 Подобрать из возможных     Исключить

Главное меню

Главное меню:  ... 

Активируется после завершения сценария (если не происходит возврат в предыдущий сценарий). Если не указан - то после завершения главное меню не будет показано.

[Дополнительно](#)

### Параметры подключения к серверу ботов:

В данной группе содержатся поля для подключения к внешней системе аккаунта чат-бота.

- **Сервер** – данное поле недоступно и проставляется автоматически. Содержит URL путь к серверу соединения с системой мессенджера. Для WhatsApp оно вводится вручную по данным из внешней системы.  
[Если у вас на сервере ограничение по открываемым URL, то вам необходимо данный путь добавить в исключения.](#)
- Токен – это ключ, который вы получаете во внешней системе мессенджера после создания аккаунта чат-бота. Если он указан верно – то после перехода на следующий элемент появятся данные об аккаунте (Или, если система не предполагает чтение логина – то текст «xxx») и отобразится статус «Соединение установлено». Можно нажать ручную кнопку «Тест соединения» - если удалось подключиться к системе – то статус поменяется на «успешно», в случае ошибки – установится статус с красным текстом и отобразится сообщения расшифровки ошибки.
- ID – Данное поле является дополнительным к токену, и появляется на некоторых типах систем (Skype, Яндекс.Алиса). При создании аккаунтов чат-ботов ([Раздел 2: Руководство по созданию чат-ботов](#)), описано из каких полей его брать.

**ВНИМАНИЕ:**

- Если не подключается к внешней системе, то проблема может быть в: Интернет-соединении с приложения 1С, недоступности/заблокированном соединении до URL, неверно указанного токена и/или ID приложения.

### Параметры подключения к 1С

В этом разделе описываются поля, для подключения внешней системы мессенджера к 1С и получения новых сообщений от клиента.

- Путь – путь к 1С, по которому внешняя система (канал чат-бота) будет отправлять вам сообщение. Данное поле обязательно указывается в следующем формате – [https://{{доменное\\_имя}}/{Имя\\_публикации}](https://{{доменное_имя}}/{Имя_публикации}). Так же доступен список с путями указанными в других чат-ботах
- Код – Для некоторых систем (ВКонтакте) необходимо добавить код подтверждения, он получается из внешней системы при создании бота.

#### ВНИМАНИЕ:

- Некоторые системы не поддерживают автоматическую установку вэб хуков, в этом случае на форме отобразится строка с пояснением. Для установки, необходимо сохранить чат-бота, и нажать на гиперссылку «Путь к 1С», далее в открывшемся браузере, с адресной строки, скопировать URL, и вставить его в соответствующее поле на странице аккаунта чат-бота (описано в [разделе 2: Руководство по созданию чат-ботов](#)).

- В пути к 1С обязательно указывать только доменное имя, если указать IP адрес, то работать не будет.

### Сценарии чат-бота

Основой взаимодействия Чат-бота является сценарий. Сценарий определяет все шаги, ветки и сообщения чат-бота. Для любого чат-бота можно установить произвольный набор сценариев. При этом один и тот же сценарий может работать на различных системах чат-бота.

В списке возможно установить галочку «Разрешены все сценарии» - тогда чат-боту будут доступны все сценарии, или **ограничить сценарии**, которые будут разрешены для текущего чат-бота.

Одни и те же сценарии можно устанавливать для всех чат-ботов, исключение составляет Яндекс.Алиса, из-за особенностей взаимодействия с ней (как правило, из-за голосового управления, требуется оформлять текст по-другому).

## 6.3. ГРАФИЧЕСКИЙ РЕДАКТОР СЦЕНАРИЕВ ПОВЕДЕНИЯ ЧАТ-БОТА

Сценарии – это описание правил поведения какой-то одной функциональности чат-бота, допустим – создание заказа, отображение баланса, опрос, диалог с сотрудником и т.д.

Сценарий не привязывается к какой-либо системе чат-ботов и может использоваться в чат-ботах различных систем, в том числе и одновременно.

### Список сценариев

Данную форму можно открыть как из подсистемы Чат-боты -> Сценарии чат-бота.

В списке отображается наименование сценария и комментарий. Справочник поддерживает иерархическое представление, с отображением групп. Рекомендуем пользоваться группами, для более удобного расположения сценариев по функциональности. Как правило, минимально разделяют по следующим группам: базовые, функциональность.

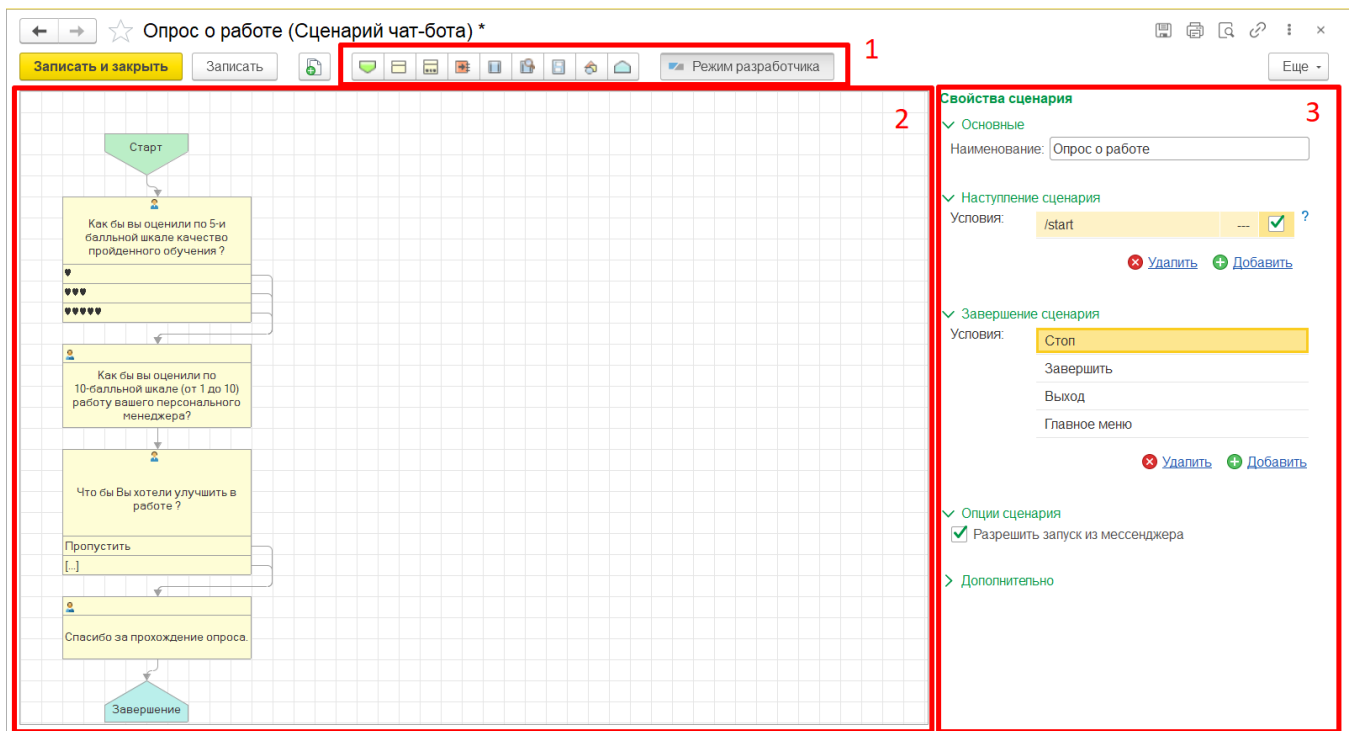
В командной панели доступна кнопка «Импортировать сценарии», с помощью нее можно импортировать множество выбранных сценарий.

### Графический редактор сценариев

Сценарии описываются в визуальном редакторе, с помощью блок схем и установки свойств для каждого элемента (в дальнейшем - шага). Это очень мощный механизм, который позволяет описать практически любое поведение чат-бота, при необходимости, подключая методы (к примеру работы с БД, записи документа и т.д.) для расширения возможностей сценария. Эта целевая схема создания сценариев, она позволяет значительно сократить время разработки сценариев.

Редактор сценария открывается при открытии элемента справочника сценарии или создания нового элемента сценария.

Рассмотрим подробнее каждую область формы:



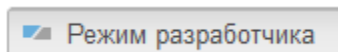
1. Область **Командная панель** – здесь располагаются основные элементы управления сценарием:



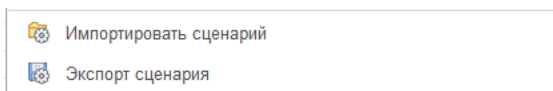
**Записать, Записать и закрыть** – после нажатия все данные записываются в базу и, если чат-бот запущен, изменения сразу применяются на поведении чат-бота.



**Элементы блок схемы** – В данной области расположены все элементы, с помощью которых строятся сценарии. Каждая из кнопок добавляет элемент на схему: Начало сценария, Отправка данных, Отправка данных с клавиатурой, Программное ветвление, запуск сценария, поиск сценария, диалог с сотрудником, Завершение сценария. Подробнее о них описано ниже.



**Режим разработчика** – при включении этого режима, отобразятся элементы и настройки для разработчиков (элементы программного ветвления и поля с подключаемыми методами). [Пример – Получить номенклатуру из БД, создать документ заказа, проверить введенный текст на соответствие договору и т.д.](#)



**Импорт/Экспорт** – через данные функции можно сохранить сценарий в внешний файл, для передачи его или архивирования. Или загрузить сценарий, при этом все настройки, блок схемы и т.д. заменяться, включая название сценария. Данные кнопки находятся в меню «Еще»

2. Область **Редактор блок-схемы** – Эта область представляет собой визуальный редактор, посредством стандартных блок-схем. Все элементы – соединяются между собой. Каждый элемент – это шаг.

3. Область **Свойства Сценария / Шага** - в данной панели находятся основные свойства, которыми настраивается сценарий и элемент шага. У сценария и элементов (шагов) разные свойства, которые отображаются автоматически при изменении фокуса выделения (если выделен элемент, то отображаются свойства шага, если нет выделенного элемента – то свойства сценария).

## Создание/редактирование сценария

Создание сценария заключается в добавление необходимых шагов и установки связей между ними, где шаг – это элемент на форме. Каждый элемент производит определенные действия, которые будут выполнены платформой. Элементы Начало и Завершение – не производят действий, они используются только как управляющие элементы. Основные правила при создании схемы:

- Сценарий должен начинаться с элемента «Старт»
- Элементы обязательно должны быть соединены линиями связи.
- В элемент могут приходить неограниченное число связей.
- В Элемент «Старт» начала сценария, не должно входить ни одной линии-связи.
- Завершаться сценарии могут множеством элементов «Завершение»

Необходимо учесть, что:

- При копировании элемента **ctrl+c**, его свойства не копируются.
- При откате действия **ctrl+z**, свойства не возвращаются.

## Свойства сценария

Создания сценария начинается с указания основных свойств, которые отображаются справа, в панели свойств (панель с именем «свойства сценария» отображается, если в редакторе блок схем не выбран ни один элемент).

Доступные настройки:

**Наименование** – Представление сценария.

**Тип подбора** – Выбирает один из вариантов при котором сценарий будет вызван:

- **По условию или переходу** – Если сценарий выбирается в зависимости от фразы или непосредственным вызовом из другого сценария

- **Если нет подходящих сценариев** – В случае если нет подходящего сценария по условиям фразы.

- **Если произошла ошибка** – Сценарий который будет вызван, если произошла ошибка, и в рамках сценария где была ошибка, она не была обработана.

**Условия наступления сценария** – Описывает слова/фразы активизации сценария. Между строчками действует логическое «ИЛИ». С этими условиями сравнивается сообщение от клиента и наименование нажатой кнопки.

Для каждого условия доступно **ранжирование** (колонка, правее ключевого слова), т.е. указание веса данного правила/слова. В итоге, если будет подобрано несколько сценариев, подбирается тот, у которого в сумме вес слов больше.

Так же можно установить галочку, пометив условие как глобальное. Это означает что вне зависимости на каком сценарии / этапе будет пользователь, у него запустится текущий сценарий при выполнении условия. К примеру, рекомендуется устанавливать глобальными команды `/start3`

Здесь допустимы некоторые регулярные выражения:

% - произвольное кол-во символов. Т.е. слово обрaмленное этим знаком будет означать что при любом его вхождении правило будет считаться подобранным. Часто используется указание символа в конце слова.

\_ - один произвольный символ.

[...] – символы из списка.

**Условия завершения сценария** – В данном списке указываются ключевые слова, по которым будет происходить безусловное завершение сценария и переход в сценарий главного меню. Как правило он используется если клиент на каком-то из шагов решил отказаться от дальнейшего взаимодействия.

**С большей долей вероятности клиент использует слова:** выход, завершить, стоп, меню...

Проверка работает на равенство слов, без учета регистра.

**Шаг ошибки** – Имя шага в рамках текущего сценария, в который будет осуществлен переход если возникла ошибка. К примеру, если ваш сценарий может вываливаться в исключения, и это запись на услугу, то стоит добавить шаг для перехода сразу к оператору, что б не потерять клиента, если была какая либо ошибка.

**Разрешить запуск из мессенджера** – Признак, который добавляет возможность запускать сценарий сотрудником для клиента прямо из мессенджера. К примеру может быть сценарий опроса, или заказа (в чате договариваются о заказе и далее сотрудник запускает для создания клиентом, возможно какая то авторизация и т.д.)

### Свойства сценария

#### Основные

Наименование:

#### Наступление сценария

Условия:    ?

Удалить  Добавить

#### Завершение сценария

Условия:

Удалить  Добавить

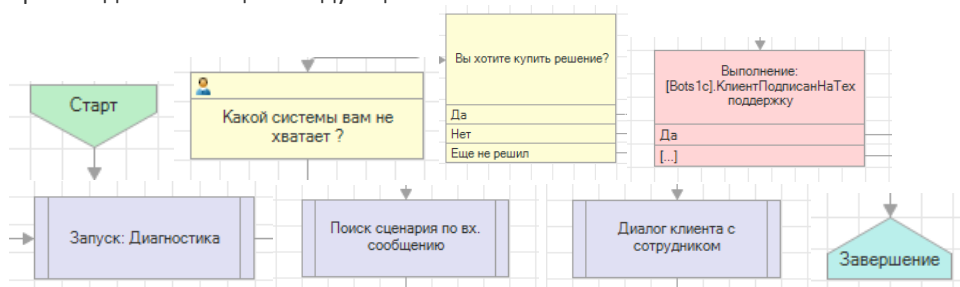
#### Опции сценария

Разрешить запуск из мессенджера

#### > Дополнительно

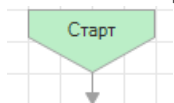
## Элементы блок схемы, назначение и их свойства

После установки основных настроек сценария, можно приступить к добавлению шагов и связей между ними, это происходит с помощью следующих элементов:



Разберем их подробнее:

### Начало сценария

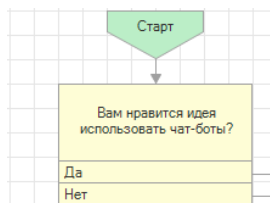


Этот элемент, обозначает начало сценария или используется для зацикливания шага.

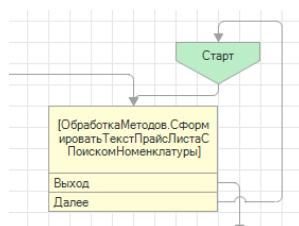
Использоваться в 2-х вариантах:

1. Для указания шага начала сценария, при этом к нему не допускается ни одного соединения.
2. Для связи элемента с самим собой, т.е. выход шага зациклить на вход в данный шаг. Данная возможность может понадобиться допустим при постраничном выводе каких-либо данных (кнопки следующая страница, предыдущая страница будут зациклены на текущий элемент)

Пример:

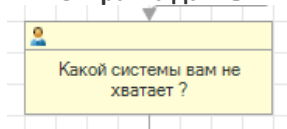


Вариант использования 1.



Вариант использования 2.

### Отправка данных



Этот элемент описывает отправляемые данные, которые платформа выполнит (отправит клиенту). Данные отправки могут быть следующих типов: сообщение, картинка, аудио, видео, файл, контакт, местоположение, запрос номера телефона, запрос местоположения. Данный элемент линейный, т.е. от него выходит одна линия связи. Его можно использовать для вывода информации или для запроса от клиента информации, введенной вручную, т.е. запрос ФИО, возраста, адреса доставки, артикула, комментариев и т.д. Возможно настройка, при которой шаг только отправляет данные, не ожидая ответ клиента.

**Свойства шага (элемента)** – Содержит настройки поведения шага:

**Шаг** – Содержит имя шага, по умолчанию имя генерируется, однако, если к нему на каком-то из шагов необходимо обращаться (допустим для получения ответа на этот шаг) – то лучше указать осмысленное имя или номер шага.

При переименовании шага обязательно следить за тем, что имя не повторяется, иначе сценарий будет вести себя непредсказуемо!

**Отправляемые данные** – Описываются сообщения и данные для отправки клиенту. Возможные следующие типы сообщений:

- Сообщение – отправляет текстовое сообщение
- Картинка, Аудио, Видео, Файл – отправляет файл клиенту в необходимом формате, который, в зависимости от мессенджера будет отображен в превью с возможностью открыть/просмотреть. Можно не беспокоиться, если мессенджер не поддерживает отправку файлов, ядро об этом знает, и отправит клиенту url ссылку с файлом.
- Контакт – отправка любого номера телефона клиенту, если мессенджер поддерживает, то у клиента будет отображен как карточка, с возможностью позвонить/сохранить.
- Местоположение – отправка местоположения клиенту с отображением на карте. Удобно для уточнения расположения вашего офиса и т.д.
- Запрос номера телефона – Для авторизации клиента можно запросить у него номер телефона, привязанный к аккаунту.

Если мессенджер не поддерживает, то запрос номера телефона произойдет отправкой текстового сообщения. Телеграмм, Вайбер – поддерживает запрос номера телефона, WhatsApp – номер телефона сразу известен, для остальных происходит запрос обычным текстовым сообщением.

- Запрос местоположения – возможность запросить у клиента местоположения.

Для Телеграмм, Вайбер, Facebook – поддерживает данный тип, для остальных происходит запрос обычным текстовым сообщением.

*При включении «режим разработчика»:*

Если в поле «Формирование» установить «По методу» то появится возможность динамического формирования отправляемых данных, т.е. текстовое сообщение или мультимедиа файл, контакт и т.д будет формироваться кодом, подключаемой функцией. Для указания функции в отобразившемся поле необходимо указать метод, который будет формировать данные для отправки. Синтаксис: «[ИмяОбработки].ИмяЭкспортногоМетода» или «ПутьКМодулю.ИмяЭкспортногоМетода». Подробный синтаксис и правила описание функции смотрите в [Руководство разработчика: Подключаемые методы](#).

**Не ожидать ответ от клиента** – Данную галочку стоит поставить, если после выполнения шага (отправки клиенту сообщений) не ожидать от него ответ, и выполнить следующий шаг.

К примеру, может использоваться перед шагом с диалогом, что бы отправить клиенту сообщение с информацией что чат начат, и перейти в диалог, и т.д.

**Аккумулировать ответы шага в массиве** – Этот параметр, позволяет все ответы клиент на текущем шаге (при построении шага с зацикливанием, т.е. множественным вызовом), накапливать и далее они будут доступны в контексте ответов. Если галочка не установлена – то будет доступен единственный ответ, если их было несколько – то последний.

Использовать этот механизм можно, к примеру, в сценариях покупки, когда клиент вводит артикул интересующего товара, и далее может еще раз добавить другой товар, в таком случае мы увидим множество ответов.

*При включении «режим разработчика»:*

Если галочка установлена – то в параметре с контекстом ответом, на данном шаге будет значение с типом массив, в котором содержится структура с ответом клиента. В случае если галочка не стоит – то на шаге будет значение с типом Структура.

*При включении «режим разработчика»:*

**Свойства шага: Действия с клавиатурой**

✓ Основные

Шаг: ПожеланиеКомментарий

✓ Отправляемые данные

Сообщение Фиксированное

Что бы Вы хотели улучшить в работе ?

↑ ↓ × Удалить + Добавить сообщение

✓ Клавиатура

Меню (вместо клавиатуры)

✓ Не проверять корректность ответа по кнопкам

№ Наименование

1 Пропустить

↑ ↓ × Удалить + Добавить кнопку

✓ Опции

Изменить данные отправки динамически

Путь к методу динамического изменения действий

Аккумулировать ответы шага в массиве

✓ События

Проверка ответа: Путь к методу для проверки ответа

**Изменить данные отправки динамически** - Для более гибкого создания данных отправки и кнопок, можно использовать возможности поля «Изменить данные отправки динамически». Основная цель его – дать возможность динамически, кодом, изменять состав данных отправки сообщений и кнопок. Т.е. в подключаемой функции есть возможности добавить, изменить и удалить как сообщения, так и кнопки.

Функция может указываться несколькими способами: «[ИмяОбработки].ИмяЭкспортногоМетода» или «ИмяКМодулю.ИмяЭкспортногоМетода», подробней можно посмотреть в [Руководство разработчика: Подключаемые методы](#)

Данный механизм может применяться, к примеру, для отправки товаров, их по поиску может быть разное кол-во и статически невозможно определить их кол-во. Поэтому при получении списка товаров добавляется кол-во данных отправки. Изменение кнопок применяется в различных целях, допустим какому-то клиенту необходимо предоставить дополнительные возможности если он авторизован, или, к примеру, для добавления и удаления кнопок с переходом на предыдущую или следующую порцию данных (стр. 7, стр. 9 – далее стр. 8 и отсутствует кнопка перехода на следующую страницу).

**События** – Это триггеры, при наступлении которых, выполняется указанная процедура/функция. Данный функционал позволяет: проводить дополнительные проверки ответа (допустим сравнивая с данными из БД); вызывать метод создания заказа, сделки и т.д.;

**Проверка ответа** – Событие срабатывает при получении ответа от клиента и вызывает функцию, указанную в данном поле. Может применяться для проверки корректности ответа (пример: корректность номер договора, номера телефона, почты и т.д.). В случае некорректного значения, можно запросить повторить ввод или прекратить выполнение сценария (Пример: договор существует, но не прошел проверку, следовательно сценарий нет смысла продолжать)

Функция может указываться несколькими способами: «[ИмяОбработки].ИмяЭкспортногоМетода» или «ИмяКМодулю.ИмяЭкспортногоМетода», подробней можно посмотреть в [Руководство разработчика: Подключаемые методы](#)

#### Отправка данных с клавиатурой

Вы хотите купить решение?
Да
Нет
Еще не решил

Этот элемент позволяет выполнять такие же действие как «Отправка данных», только дополнительно можно прикреплять кнопки меню, которое отобразится пользователю. От этого элемента будет выходить столько связей, сколько кнопок, т.е. данный элемент реализует разветвление сценария в зависимости от ответа клиента. Добавление кнопок происходит в панели свойств.

**Свойства шага (элемента)** – Полностью повторяет возможности элемент «Отправка данных» (назначение свойств смотрите выше в предыдущем пункте). Дополнительно, здесь содержатся следующие настройки:

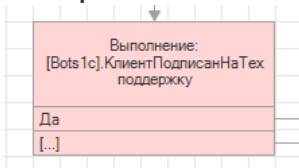
**Клавиатура** – В данной таблице добавляются кнопки, которые будут выведены клиенту после отправки всех данных сообщений. Для добавления кнопки, можно нажать команду «Добавить кнопку» и указать номер строки «№» вывода клавиатуры (**Не путать с номером кнопки**), он должен начинаться с 1 и может повторяться на нескольких кнопках (т.е. на одной строке могут располагаться несколько (до 4х) кнопок). Увеличивать обязательно последовательно (т.е. не допускается 1, 3, пропустив 2). На каждую кнопку появится вариант в элементе шага на схеме, его необходимо будет связать с следующим по логике шагом.

Кнопки бывают 2х видов – **меню** (показываются вместо клавиатуры) и **инлайн** (показываются в ленте с текстом). Инлайн кнопки поддерживаются не всеми системами, если не поддерживается –вывод эмулируется текстом с проставлением порядкового номера, для более легкого выбора пункта.

**Не проверять корректность ответа по кнопкам** - При отображении кнопок, ответ клиента проверяется на корректность соответствие кнопкам, т.е. ответ должен быть дан либо по нажатию кнопки, либо введено ее название, или номер кнопки если она выведена текстом. При установлении галочки в данном пункте, данная проверка отключается и у клиента появится возможность ввода любого произвольного текста. При этом на элементе добавиться еще один вариант «...» по нему пойдет сценарий при альтернативном ответе.

При включении «режим разработчика»:

### Обработка с ветвлением



Элемент, позволяющий программно указывать ветку, по которой будет вестись дальнейший сценарий. Так же можно его сделать линейным и вызвать любую функцию, для выполнения произвольного кода.

**Свойства шага: Обработка с ветвлением**

✓ Основные  
Шаг: ed425bff

✓ Опции  
 Ветка "по умолчанию" (любой ответ метода)  
Ветвления:  
- ВРЕМЯ\_РАБОЧЕЕ  
- [...]

Удалить Добавить ветвление

✓ События  
Выполнить: [Bots 1c\_ДемоПримеры].Выполнен  
Записать результат в: \_РезультатБаллы

### Свойства шага (элемента):

**Ветвления** – Список, в котором указываются возможные значения результата выполнения подключаемой функции, на каждое значение создается вариант ветвления в элементе (пример Да, [...]), от которого можно направить связь в необходимую ветку сценария. По умолчанию создается вариант ветвления [...], что означает любой результат функции или иной результат (если не подошел ни один из перечисленных).

Если варианты ветвление создавать нет необходимости, и элемент используется как линейный, т.е. только для выполнения функции, то можно оставить одно значение «[...]»

Если значения «[...]» не будет, но функция вернет значение которого нет в списке – будет зафиксирована ошибка, и сценарий завершится.

**Выполнить** – Поле, с указанием функции, которая будет выполнена. Результат выполнения будет сравнен с значениями в списке «Ветвления».

Синтаксис: «[ИмяОбработки].ИмяЭкспортногоМетода» или «ПутьКМодулю.ИмяЭкспортногоМетода». Подробный синтаксис и правила описание функции смотрите в [Руководство разработчика: Подключаемые методы](#).

**Записать результат** – при указании имени шага, результат выполнения метода так же будет записан в контекст. Можно возвращать любые типы данных.

**Свойства шага: Обработка с ветвлением**

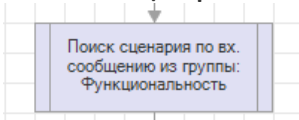
✓ Основные  
Шаг: 3

✓ Опции  
 Ветка "по умолчанию" (любой ответ метода)  
Ветвления:  
[...]  
Да

Удалить Добавить ветвление

✓ События  
Выполнить: [Bots 1c].КлиентПодписанНаТехп  
Записать результат в: 1

### Поиск сценария



Данный элемент позволяет автоматически подобрать сценарий по введенной ранее клиентом фразе. Поиск может проводиться как по всем сценариям, доступным чат-боту, так и ограниченном группой. При поиске, фраза клиента анализируется с условиями наступления в сценарии, включая вес фразы, если заполнен вес ранжирования.

После данного шага обязательно должен располагаться элемент «Завершение сценария».

### Свойства шага (элемента):

**Ограничить группой** – При указании группы из справочника «Сценарии чат-бота», поиск будет проводиться только среди сценариев, находящихся в данной группе. Естественно, так же применяются условия что сценарий должен быть разрешен для чат-бота и не должен быть помечен на удаление.

**Если не найден** – Указывается имя шага текущего сценария, который будет выполнен если сценарий не удалось найти. Как правило в этом шаге можно написать клиенту что бот не понял, такой функциональности еще нет, или предложить связь с менеджером.

**Не возвращаться** – Галочка проставляется если по бизнес логике, после выполнения шага, в текущий сценарий не надо возвращаться. В этом случае в вызываемом сценарии следует предусмотреть вызов какого то сценария (к примеру главное меню), после его завершения.

**Свойства шага: Поиск сценария**

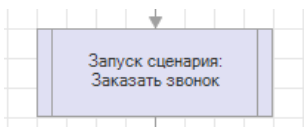
✓ Основные  
Шаг: c671c1dd

✓ Поведение  
Ограничить группой: Демонстрационные примеры ...  
НеудачаПоиска

Если не найден: Имя шага, к которому будет осуществлен переход если не удалось найти сценарий.

Не возвращаться:   
Если установить - после завершения вызываемого сценария переход в родительский, с которого он был вызван, не будет осуществлен.

### Вызов сценария



Данный элемент позволяет запустить указанный сценарий.

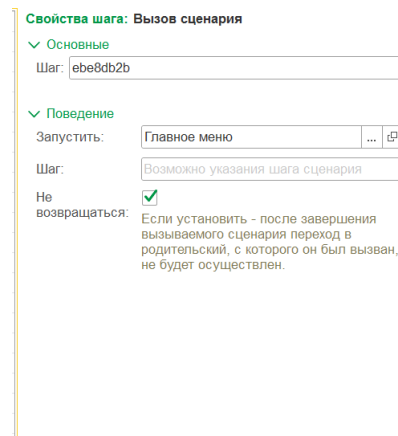
После данного шага обязательно должен располагаться элемент «Завершение сценария».

#### Свойства шага (элемента):

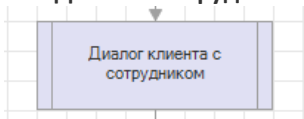
**Запустить** – В данном поле указывается элемент справочника «Сценарии чат-бота», который будет запущен. Пустое значение в данном поле не допускается.

**Шаг** – Указывается имя шага, если необходимо перейти не на начало, а сразу на определенный шаг сценария

**Не возвращаться** – Булево, при проставлении галочки, после завершения сценария не произойдет возврат по стеку в текущий. В этом случае сценарий сам должен предусмотреть запуск главного меню или какие либо другие действия.



#### Диалог с сотрудником



Элемент, с помощью которого можно переводить бота из автоматического режима в режим диалога с сотрудником. При получении нового сообщения или мультимедиа данных от клиента создается оповещение сотруднику, и отображается на в нижнем правом углу. Данное оповещение отображается даже если мессенджер чат-ботов в 1С не активен (но должен быть открыт в вкладках), и даже если 1С свернута. При нажатии на оповещение сотрудник переходит в чат с клиентом и может вести диалог. Платформа, позволяет очень гибко настраивать сотрудника, отдел или всех сотрудников, на которого будете переведен диалог.

#### Свойства шага (элемента):

**Правила подбора менеджера диалога** – Здесь можно указать список правил, по которым будет подбираться сотрудник или сотрудники для диалога. Все правила, указанные в этом списке будут применятся последовательно, пока не вернется значение с именем/именами пользователей.

В списке можно выбрать predefined правила:

**Из последней переписки** – Будет выбран сотрудник, который ранее взаимодействовал с пользователем. В случае, если было несколько сотрудников – будет выбран последний. Если клиент не общался ранее не с одним сотрудников, будет выполнен поиск сотрудника в следующем правиле.

**Основной менеджер контрагента** – Если клиент связан с контрагентом (партнером, лидом и т.д.), то будет осуществлен поиск основного менеджера из его карточки. Если контрагент не указан или основной менеджер пустой, как и в предыдущем правиле, будет выполнен поиск сотрудника в следующем правиле.

**В случайном порядке** – Из всех сотрудников, у которых есть право открывать мессенджер чат-ботов, будет выбран случайный.

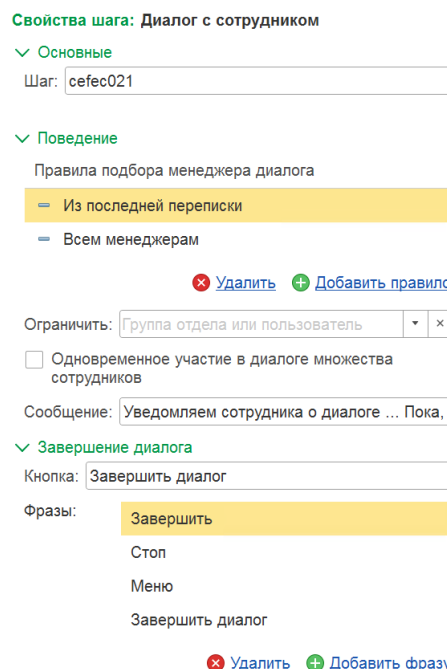
**Всем менеджерам** – Оповещение направится всем сотрудникам. Даже если не указать это правило, и никакие из указанных не вернули имя сотрудника – так же оповещение будет отправлено всем сотрудникам.

*При включении «режим разработчика»:*

**{Свой вариант определение сотрудника}** – Платформа позволяет указать свой вариант определения сотрудника, в данном поле можно задать путь к функции, которая выполнится, и в возврате выполнения, ожидается строка с именем (или именами, разделенные запятой) пользователя как оно указано в конфигураторе.

**Синтаксис:** «[ИмяОбработки].ИмяЭкспортногоМетода» или «ПутьКМодулю.ИмяЭкспортногоМетода». [Подробный синтаксис и правила описание функции смотрите в Руководство разработчика: Подключаемые методы.](#)

**Ограничить** – Можно указать список сотрудников и отделов, на которые переводится данный диалог. В случае если указания правила подбора менеджера – оба условия обрабатывают вместе.



**Одновременное участие в диалоге множества сотрудников** – при установленной галочке – все назначенные сотрудники диалога смогут одновременно писать сообщение. При отключенной – диалог забирает сотрудник, и у остальных он пропадает.

**Сообщение** – Данный текст будет выведен клиенту.

**Кнопка завершения диалога** – Текст кнопки, который будет отображен клиенту для завершения / выхода из диалога. Кнопка доступна только если указан текст сообщения.

**Фразы завершения диалога** – Здесь можно указать фразы, при вводе клиентом которых, диалог завершится и будет осуществлен переход к следующему шагу.

Сотрудник так же может завершить диалог, при нажатии соответствующей кнопки  в мессенджере.

### Завершение сценария



Этот элемент используется для указания завершения сценария, к нему может идти связь с любого элемента, данных элементов может быть сколько угодно. После перехода на данный элемент – сценарий будет завершен.

## Шаблонные сообщения для WhatsApp 1msg

В данной системе чат-ботов, возможно открытие диалога, однако есть ограничение, текст первого сообщения должен быть согласован. При этом он определенным образом оформляется у нас в сценарии:

1. Регистрируется шаблонное сообщение в ЛК 1msg, к примеру сообщение:

*{{1}}, приветствуем вас в НашаКомпания!*

*Благодарю за заказ и отправляю подробности.*

*Номер заказа: {{2}}*

*Сумма к оплате: {{3}}*

*Способ оплаты: {{4}}*

2. В сценарии создается сообщение следующего вида, вместо параметров 1msg проставляя имена параметров из контекста:

*{ИмяКл}, приветствуем вас в Reuwood!*

*Благодарю за заказ и отправляю подробности.*

*Номер заказа: {НомерЗаказа}*

*Сумма к оплате: {СуммаКОплате}*

*Способ оплаты: {СпособОплаты}*

*Дата и время доставки: {ДатаВремяДост}*

*Адрес доставки: {АдресДост}*

3. В чат-боте 1msg в разделе Опции -> Данные шаблонных сообщений - добавляете строку, указывая

- Сценарий - текущий сценарий, один из шагов которых работает с шаблонным сообщением

- Шаг - имя шага, в котором содержится шаблонное сообщение

- Имя шаблона - Зарегистрированное имя шаблона в 1msg

- Имя параметров - Имя из контекста, в котором будут находиться все значения параметров, в нашем случае:

ИмяКл, ИмяМен, НомерЗаказа, СуммаКОплате, СпособОплаты, ДатаВремяДост, АдресДост

Пример: ОповещениеОЗаказе Открытие24ЧасовогоДиалога order\_conf\_msk\_hand

СтруктураПараметровШаблонна

4. При запуске сценария:

- либо запускаете его передавая в поле параметры структуру, с элементом, содержащим все значения

- или первым шагом используете ПрограмноеВетвление и в нем записываете в контекст / возвращаете нужную структуру.

Пример

Структура:

СтруктураПараметровШаблонна: Структура

ИмяКл: Александр

ИмяМен: Дмитрий

НомерЗаказа: 000385

СуммаКОплате: 100000

СпособОплаты: Наличная

ДатаВремяДост: 12.12.2025

АдресДост: xxxx

Примечание: Параметры подставлять можно, как в заголовке, так и в теле. Картинки передавать нельзя. Имена кнопок должны быть статические и соответствовать шаблонным

# РУКОВОДСТВО ПО НАПИСАНИЮ МЕТОДОВ (КОДИНГ)

## 4.1. ПОДКЛЮЧАЕМЫЕ МЕТОДЫ

Графический редактор сценариев – это очень мощный механизм, с помощью которого можно построить даже сложный сценарий за небольшое время. Однако, как правило, при создании сложного сценария, необходим дополнительный механизм, расширяющий возможности управления шагами - динамическое формирование текста, сложных проверок сообщений, итоговое создание документов и т.д. Для этого реализована возможность подключения произвольных методов к сценарию, и их выполнения в определенных случаях.

### Правила указания метода:

Метод может быть указан 2-я вариантами:

#### 1. [ИмяОбработки].ИмяЭкспортногоМетода

Данное указание функции позволяет расположить ее в внешней обработке, и подключать/обновлять динамически прямо в 1С:Предприятии, т.е. после изменения кода, нет необходимости применять изменения в конфигураторе, можно просто обновить обработку.

Обработка с методами может быть загружена несколькими способами:

**Через механизм «Дополнительные обработки и отчеты»** - данная подсистема присутствует в любых конфигурациях 1С или 1С:Совместимо. Преимущества – все подключаемые обработки находятся в одном месте.

**Через внутренний механизм «Мессенджер чат-ботов - Настройки – Администрирование – Подключаемые обработки»** - используется если у вас самописная конфигурация или по каким либо причинам стандартная подсистема не работает.

**ВНИМАНИЕ:** В данном механизме есть небольшой минус – для разработчика будет тяжело «дебажить» код в такой обработке, так как в них невозможно установить точки останова и проследить за выполнением кода.

#### 2. ПутьКМодулю.ИмяЭкспортногоМетода

Любой путь к экспортному методу конфигурации (модуля, справочника, документа и т.д.). Данное указание не рекомендуется, так как сценарий привязывается к конфигурации, но так-же возможен.

**ВНИМАНИЕ:** Плюс этого механизма – отладка с дебагом кода работает. Минус – неудобно переносить и экспортировать такие методы.

### Правила к подключаемой обработке:

Обработка должна соответствовать стандартным требованиям внешних подключаемых обработок в 1С. Т.е. в модуле обработке должен быть определен метод с возвращаемой структурой – информацией об обработке:

Функция СведенияОВнешнейОбработке() Экспорт

Результат = Новый Структура();

Результат.Вставить("Вид", "ДополнительнаяОбработка");

Результат.Вставить("Версия", "7.0.0");

Результат.Вставить("Наименование", "Bots1С: Шаблон обработки");

Результат.Вставить("БезопасныйРежим", Ложь);

Результат.Вставить("Назначение", Новый ТаблицаЗначений);

Результат.Вставить("Команды", Новый ТаблицаЗначений);

Результат.Вставить("Информация", "Шаблон обработки, добавляющая методы, для расширения возможностей сценарев работы с БД.");

КонецФункции

**ВНИМАНИЕ:** Для повышения производительности обработка кэшируется, чтобы, после обновления, подтянулась новая версия обработки, необходимо предварительно изменить значение в поле «версия» структуры СведенияОВнешнейОбработке.

Все методы, которые будут доступны для вызова – необходимо объявлять с ключевым словом «Экспорт».

Методы, которые подключены к сценарию, вызываются ядром и у них **сигнатура параметров должна точно соответствовать описанным ниже**, в зависимости от назначения метода. Актуальную обработку-шаблон должна

быть в поставке модуля, а так же можно скачать по адресу: [https://bots1c.ru/download/extensionTemplate\\_7.xx.zip](https://bots1c.ru/download/extensionTemplate_7.xx.zip)  
Теперь остановимся на методах и их назначении, которые можно подключить к сценарию:

### 1. Подключаемая функция - Динамически формируемые данные сообщения.

Применяется для динамического формирования сообщения (текстового, картинки, аудио, видео, файла, контакта, местоположение, запроса контакта и местоположения) в данных отправки.

Подключение: В элементах сценария Действие или Действие с клавиатурой, в группе «Отправляемые данные», в добавленных данных для отправки необходимо выбрать Формирование – «по методу», и в поле ниже указать путь к методу.

Функция ДинамическоеСообщение(ЧатБот, IDЧата, СценарийСсылка, ШагиВзаимодействия) Экспорт  
Возврат ..... ;  
КонецФункции

Входящие параметры:

**ЧатБот** – <Ссылка> на справочник чат ботов.

**IDЧата** – <Число, Строка> ID пользователя во внешней системе.

**СценарийСсылка** – <Ссылка> на выполняемый сценарий.

**ШагиВзаимодействия** – Соответствие, где ключ – имя шага, а значение структура ответа, с некоторыми полями из регистра. Если установлена галочка аккумулировать ответ – то массив с элементами такой структуры. Содержимые поля: **IDЗаписи, Период, ТипЗаписи, ТипСообщения, Сообщение, СообщениеИмяФайла, ПользовательИБДиалогаССотрудником, КлючЗаписиВнутр**. Для получения двоичных данных (они не добавляются так как критически влияют на производительность), можно использовать следующий код:

```
ШагСДвоичнымиДанными = ШагиВзаимодействия.Получить("1");
```

```
ДвоичныеДанные = Bots1C_Служебный.ПолучитьЗаписьПоКлючуЗаписи(
```

```
ЗначениеИзСтрокиВнутр(ШагСДвоичнымиДанными.КлючЗаписиВнутр)).СообщениеХранимыйФайл;
```

Возвращаемое значение: в зависимости от типа сообщения:

Сообщение: <Строка>

Картинка, Аудио, Видео, Файл: <Структура(«МультимедиаДвоичныеДанные, МультимедиаИмяФайла», <ДвоичныеДанные>, <Строка>)>

Местоположение: <Структура(«Широта, Долгота», <Число>, <Число>)>

Контакт: <Структура(«Телефон, Имя», <Строка>, <Строка>)>

ЗапросМестоположения: <Структура(«Сообщение, ТекстКнопки», <Строка>, <Строка>)>

ЗапросНомераТелефона: <Структура(«Сообщение, ТекстКнопки», <Строка>, <Строка>)>

Пример:

Функция ДинамическоеСообщение(ЧатБот, IDЧата, СценарийСсылка, ШагиВзаимодействия) Экспорт  
Возврат "Этот шаг тестирует отправку динамических данных";

Возврат Новый Структура("МультимедиаДвоичныеДанные, МультимедиаИмяФайла",

ЭтотОбъект.ПолучитьМакет("Демо\_Картинка"), "Демо\_Картинка.jpg")

Возврат Новый Структура("Широта, Долгота", 55.751596, 48.750269);

Возврат Новый Структура("Телефон, Имя", "+79677727515", "Интеграция ботов с 1С");

Возврат Новый Структура("Сообщение, ТекстКнопки", "Укажите местоположение", "Отправить");

Возврат Новый Структура("Сообщение, ТекстКнопки", "Укажите номер телефона", "Отправить");

Возврат Новый Структура("Сообщение, ТекстКнопки", "Укажите ваше местоположение", "Отправить");

Возврат Новый Структура("Сообщение, ТекстКнопки", "Укажите ваш номер", "Отправить");

КонецФункции

### 2. Подключаемая функция - Динамическое изменение данных отправки (сообщений и клавиатуры):

Данный механизм позволяет динамически изменить сформированные сообщения и клавиатуру. Т.е. возможно добавить, изменить и удалить какое-либо отдельное сообщение или кнопку клавиатуры.

Функция ДинамическоеИзменениеСообщенийКлавиатуры (ЧатБот, IDЧата, СценарийСсылка, ШагиВзаимодействия, ДанныеСообщений, ДанныеКлавиатуры)

Возврат Структура("ДанныеСообщений, ДанныеКлавиатуры", <МассивСтруктур>, <МассивСтруктур>);

## КонецФункции

Входящие параметры:

**ЧатБот** – <Ссылка> на справочник чат ботов.

**IDЧата** – <Число, Строка> ID пользователя во внешней системе.

**СценарийСсылка** – <Ссылка> на выполняемый сценарий.

**ШагиВзаимодействия** – Стандартный параметр, описан выше.

**ДанныеСообщений** – <Элемент массива>: Структура("Тип<Перечисление.Bots1C\_ТипыСообщений>, ФормироватьПоМетодуПризнак<Булево>, ...СвойстваПоТипу..."), где **ФормироватьПоМетодуПризнак** - Ложь, если сообщение будет статическое, если далее вычисляться методом - Истина, и добавляется свойство **ФормироватьПоМетодуПуть**<Строка> **СвойстваПоТипу** - Уникальные свойства для текущего типа данных, можно посмотреть выше в ДинамическоеФормированиеСообщений(Для сообщение доп.свойство "Сообщение")

**ДанныеКлавиатуры** – [Элемент массива]: Структура("Тип<Перечисление.Bots1C\_ТипыСообщений>, НомерСтроки<Число>, КлючКнопки<Строка>, Представление<Строка>"), где **НомерСтроки** - Номер строки вывода кнопок, должен быть неразрывным, может повторяться у нескольких кнопок, не более 5-и.

**КлючКнопки** - Должен соответствовать существующему ключу из редактора, для корректности связей.

**Представление** – Наименование кнопки.

Возвращаемое значение:

Структура("ДанныеСообщений, ДанныеКлавиатуры", <МассивСтруктур>, <МассивСтруктур>);

Пример:

Функция Диагностика\_ДинамическоеИзменениеКнопок(ЧатБот, IDЧата, СценарийСсылка, ШагиВзаимодействия, ДанныеСообщений, ДанныеКлавиатуры) Экспорт

ДанныеСообщений.Добавить(Новый Структура("Тип, ФормироватьПоМетодуПризнак, Сообщение", Перечисления.ТипыСообщенийЧатБота.Сообщение, Ложь, "Товар №1: .....");

ДанныеСообщений[ИндексСообщенияРанееВычисленный].Сообщение = "Страница 5 из 10";

ДанныеСообщений.Удалить(ИндексСообщенияРанееВычисленный);

ДанныеКлавиатуры.Добавить(Новый Структура("Тип, НомерСтроки, КлючКнопки, Представление", Перечисления.ТипыСообщенийЧатБота.КнопкаКлавиатуры, 1, "Дальше", "Последняя страница"));

ДанныеКлавиатуры[ИндексКнопкиДальшеРанееВычисленный].Представление = "Страница 5 из 10";

ДанныеКлавиатуры.Удалить(ИндексКнопкиДальшеРанееВычисленный);

Возврат Новый Структура(«ДанныеСообщений, ДанныеКлавиатуры», ДанныеСообщений,

ДанныеКлавиатуры);

КонецФункции

### 3. Подключаемая функция – Выполняемый метод ветвления

Применяется для реализации программного ветвления или для выполнения произвольного метода. Возвращаемый результат метода сравнивается с вариантами ветвления шага и при совпадении будет выполнен следующий шаг ветки, или ветка Иначе [...].

Функция Метод\_ПриОтвете(ЧатБот, IDЧата, СценарийСсылка, ШагиВзаимодействия,) Экспорт

Возврат ..... ;

КонецФункции

Входящие параметры:

**ЧатБот** – <Ссылка> на справочник чат ботов.

**IDЧата** – <Число, Строка> ID пользователя во внешней системе.

**СценарийСсылка** – <Ссылка> на выполняемый сценарий.

**ШагиВзаимодействия** – Стандартный параметр, описан выше.

Контракт на возвращаемое значение: <Строка>

Пример:

Функция ВыдатьПроизвольноЧислоДо2х(ЧатБот, IDЧата, СценарийСсылка, ШагиВзаимодействия) Экспорт

```
РезультатПроверки = ОпределитьДоговорКорректный(.....);
Возврат ?( РезультатПроверки = Истина, "YES", "NO");
КонецФункции
```

#### 4. Обработчик события: ПроверкаОтвета

Применяется для проверки корректности ответа (пример: корректность номер договора, номера телефона, почты и т.д.). Возможное поведение: повторный запрос повтор ввода ответа, или принудительное прекращение текущего сценария (Пример: договор не прошел проверку, клиент в черном списке и т.д.)

```
Функция Метод_ПриОтвете(ЧатБот, IDЧата, СценарийСсылка, ШагиВзаимодействия, Запись_ОтветНаШаг)
Экспорт
    Возврат ..... ;
КонецФункции
```

Входящие параметры:

**ЧатБот** – <Ссылка> на справочник чат ботов.  
**IDЧата** – <Число, Строка> ID пользователя во внешней системе.  
**СценарийСсылка** – <Ссылка> на выполняемый сценарий.  
**ШагиВзаимодействия** – Стандартный параметр, описан выше.  
**Запись\_ОтветНаШаг** – <Структура> все поля записи регистра Vots1C\_ИсторияСообщений.

Контракт на возвращаемое значение:

```
<Структура("ПовторитьШаг, СообщениеКлиенту")>
Где,
ПовторитьШаг – <Булево> Если ответ не прошел проверку и надо запросить повтор ввода.
СообщениеКлиенту – <Строка> Сообщение, которое будет отправлено клиенту.
```

*Примечание: Не выполняется на последнем шаге сценария, т.к. последний шаг завершающий, без ожидания ответа.*

Пример:

```
Функция ДинамическоеСообщение(ЧатБот, IDЧата, СценарийСсылка, ШагиВзаимодействия,
Запись_ОтветНаШаг) Экспорт
    Результат = Новый Структура("ПовторитьШаг, СообщениеКлиенту", Ложь, "");
    Если ПроверитьНаличиеДоговораПоКоду(Запись_ОтветНаШаг.Сообщение) = Ложь Тогда
        Результат. ПовторитьШаг = Истина;
        Результат. СообщениеКлиенту = "Договор не существует, доступ запрещен.";
    КонецЕсли;
    Возврат Результат;
КонецФункции
```

#### 5. Обработчик события: ПослеЗавершенияШага

Применяется для выполнения метода при завершающих действий шага (пример: создание итогового документа, справочника и т.д.)

```
Процедура Метод_ПослеШага(ЧатБот, IDЧата, СценарийСсылка, ШагиВзаимодействия) Экспорт
.....
КонецПроцедуры
```

Входящие параметры:

**ЧатБот** – <Ссылка> на справочник чат ботов.  
**IDЧата** – <Число, Строка> ID пользователя во внешней системе.  
**СценарийСсылка** – <Ссылка> на выполняемый сценарий.  
**ШагиВзаимодействия** – Стандартный параметр, описан выше.

Контракт на возвращаемое значение: **НЕТ**

Пример:

```
Процедура Диагностика_Сообщение(ЧатБот, IDЧата, СценарийСсылка, ШагиВзаимодействия) Экспорт
    ЗаявкаОбъект = Документы.Заявка.СоздатьДокумент();
    ЗаявкаОбъект.Контрагент = Bots1C_API.ПолучитьКонтактЧата(ЧатБот, IDЧата).Контрагент;
    ЗаявкаОбъект.ТекстЗаявки = ШагиВзаимодействия.Получить("1").Сообщение;
    ЗаявкаОбъект.ФотоЗаявки = ШагиВзаимодействия.Получить("2").СообщениеХранимыйФайл;
    ЗаявкаОбъект.Записать();
КонецПроцедуры
```

## 6. Подключаемая функция: ОпределениеСотрудникаДиалога

Применяется для определения сотрудника (сотрудников), на которого будет переведен диалог с клиентом, при его запуске.

```
Функция ВыборСотрудникаДиалога_ПереводНаАдминистратора(ЧатБот, IDЧата, СценарийСсылка) Экспорт
.....
КонецКонецФункции
```

Входящие параметры:

- ЧатБот** – <Ссылка> на справочник чат ботов.
- IDЧата** – <Число, Строка> ID пользователя во внешней системе.
- СценарийСсылка** – <Ссылка> на выполняемый сценарий.

Контракт на возвращаемое значение: <Строка> - имя (или имена) пользователя как оно задано в конфигураторе. В случае нескольких имен – они должны быть разделены запятой без пробела. Если перевести необходимо на всех сотрудников – возвращается символ «\*».

Пример:

```
Процедура Диагностика_Сообщение(ЧатБот, IDЧата, СценарийСсылка, ШагиВзаимодействия) Экспорт
    Возврат «Администратор,Admin»;
КонецПроцедуры
```

## 4.2. API

Все методы взаимодействия с внешними системами чат ботов и ядра – унифицированы и приведены к одному общему виду и выделены в отдельный API, скрывающей всю внутреннюю реализацию транспорта и ядра, для более легкого взаимодействия с чат-ботами и самим ядром.

Все методы разделяются по группам.

Взаимодействие с чат ботом:

В данной группе содержатся методы для «ручной» отправки сообщений:

- При создании оповещений клиента при определенных действиях, в обработке подписки на события
- При создании оповещений клиента о задолженностях, в регламентных заданиях
- При создании розыгрышей, когда розыгрыш стартует в определенное время, и необходимо инициировать первое сообщение с запуском сценария.

### 1. Функция ЧатБотAPI.ОтправитьСообщение(КонтактЧатБота, Сообщение, ДанныеКлавиатуры, Настройки)

Данная функция необходима для отправки сообщений, последним параметром можно передать клавиатуру для вывода.

Где параметры:

- КонтактЧатБота** – <Ссылка> на справочник чат ботов.
- Сообщение** – <Строка>, отправляемое сообщение.
- ДанныеКлавиатуры** – <Массив<Структур>> Клавиатура, отображаемая пользователю, описан ниже.
- Настройки** – <Структура> Этот параметр содержит дополнительные настройки отправки сообщений, он не является обязательным. Возможные свойства:
  - УстановитьСценарий**– <Ссылка.СценарииЧатБотов> При передаче данного значения, при записи в историю сообщений запишется данный сценарий, цель его указания – направить взаимодействие

по определенному сценарию.

**УстановитьОжиданиеОтвета** – <Булево> Признак, при указании которого будет ожидаться ответ пользователя, после ответа управление передается ядру с целью подбора дальнейшего шага внутри активного сценария.

Возвращаемое значение <структура>:

**Статус** – <Булево> Признак, показывающий результат отправки (Удачно или ошибка отправки)

**IDЗаписи** – <УникальныйИдентификатор> ID записи в регистре сведений ИсторияСообщений.

**ОписаниеОшибки** – <Строка> Описание ошибки

*Пример:*

*Результат = Vots1C\_Транспорт.ОтправитьСообщение(КонтактЧатБота, Сообщение, ДанныеКлавиатуры, Настройки);*

## 2. Функция ЧатБотAPI. ОтправитьДвоичныеДанные(КонтактЧатБота, ТипДанных, ДвоичныеДанные, ИмяФайла, Сообщение, ДанныеКлавиатуры, Настройки)

С помощью данной функции можно отправить мультимедиа-данные: картинки, аудио, видео или произвольные файлы.

Где параметры:

**КонтактЧатБота, Сообщение, ДанныеКлавиатуры, Настройки** – см. в пункте 1.

**ТипДанных** – <Перечисления.ТипыСообщенийЧатБота> Тип отправляемых данных:

Перечисления.ТипыСообщенийЧатБота.Картинка

Перечисления.ТипыСообщенийЧатБота.Аудио

Перечисления.ТипыСообщенийЧатБота.Видео

Перечисления.ТипыСообщенийЧатБота.Файл

**ДвоичныеДанные** – <ДвоичныеДанные> Данные мультимедиа файла

**ИмяФайла** – <Строка>, тестовое имя файла с расширением. Пользователь увидит его при скачивание, или при отображении файла – гиперссылкой.

Возвращаемое значение: <структура>:

**Статус** – <Булево> Признак, показывающий результат отправки (Удачно или ошибка отправки)

**IDЗаписи** – <УникальныйИдентификатор> ID записи в регистре сведений ИсторияСообщений.

**ОписаниеОшибки** – <Строка> Описание ошибки

*Пример:*

*Результат = Vots1C\_Транспорт.ОтправитьДвоичныеДанные(КонтактЧатБота, ТипДанных, ДвоичныеДанные, ИмяФайла, ДанныеКлавиатуры, Сообщение, Настройки);*

## 3. Функция ЧатБотAPI. ОтправитьКонтакт(КонтактЧатБота, НомерТелефона, Представление, ДанныеКлавиатуры, Настройки);

Функция предназначена для отправки сообщения в виде контакта, т.е. с номером телефона и представлением.

Где параметры:

**КонтактЧатБота, ДанныеКлавиатуры, Настройки** – см. в пункте 1.

**НомерТелефона** – <Строка> Номер телефона в международном формате.

**Представление** – <Строка> Представление контакта.

Возвращаемое значение: см. в пункте 1.

*Пример:*

*Vots1C\_Транспорт.ОтправитьКонтакт(КонтактЧатБота, НомерТелефона, Представление, ДанныеКлавиатуры, Настройки);*

## 4. Функция ЧатБотAPI. ОтправитьМестоположение(КонтактЧатБота, Широта, Долгота, ДанныеКлавиатуры, Настройки);

Данная функция позволяет отправить координаты с отображением их на карте.

Где параметры:

**КонтактЧатБота, ДанныеКлавиатуры, Настройки** – см. в пункте 1.

**Широта** – <Число> Широта, в виде xx.xxxxxx

**Долгота** – <Число> Долгота, в виде xx.xxxxxx

Возвращаемое значение: см. в пункте 1.

*Пример:*

*Vots1C\_Транспорт.ОтправитьМестоположение(КонтактЧатБота, Широта, Долгота, ДанныеКлавиатуры, Настройки);*

## 5. Функция ЧатБотAPI. ЗапроситьНомерТелефона(КонтактЧатБота, Сообщение, ТекстКнопки, Настройки);

Данная функция используется для запроса номера телефона клиента, прикрепленного к аккаунту, ответ

сохраняется в истории. Может быть с разным типом данных: ПодтвержденныйКонтакт – если была нажата кнопка и отправлен контакт на аккаунте или Контакт - если номер телефона был введен вручную как сообщение. Таким образом производится идентификация клиента и связка его с контрагентами из справочника Контрагенты.

Где параметры:

**КонтактЧатБота , Настройки** – см. в пункте 1.

**Сообщение** – <Строка> Сообщение с запросом номера телефона, отправляемое пользователю.

**ТекстКнопки** – <Строка> Текст кнопки отправки номера телефона (если бот поддерживает).

Возвращаемое значение: см. в пункте 1.

*Пример:*

*Vots1C\_Транспорт.ЗапроситьНомерТелефона(КонтактЧатБота, Сообщение, ТекстКнопки, Настройки);*

## **6. Функция ЧатБотAPI. апроситьМестоположение(КонтактЧатБота, Сообщение, ТекстКнопки, Настройки)**

Данная функция используется для запроса местоположения клиента. Специфическая функция, обычно не используется, ответ сохраняется в истории.

Где параметры:

**КонтактЧатБота , Настройки** – см. в пункте 1.

**Сообщение** – <Строка> Сообщение с запросом местоположения, отправляемое пользователю.

**ТекстКнопки** – <Строка> Текст кнопки отправки местоположения (если бот поддерживает).

Возвращаемое значение: см. в пункте 1.

*Пример:*

*ЧатБотAPI. ЗапроситьМестоположение(КонтактЧатБота, Сообщение, ТекстКнопки)*

Работа с сценариями:

## **7. Процедура ЧатБотAPI. ЗапуститьСценарий(КонтактЧатБота, СценарийСсылка, УстановитьШаг, ПараметрыЗапуска);**

Метод необходим для запуска сценария по какому-то событию внутри базы данных. Может применяться допустим в регламентном задании, рассылках и т.д. При запуске сценария выполняется первый шаг сценария и дальше ожидается ответ от клиента.

Где параметры:

**КонтактЧатБота** – см. в пункте 1.

**УстановитьШаг** – <Строка>, необязательный параметр. Указывает шаг сценария с которого запустить сценария. По умолчанию сценарий запускается сначала.

**ПараметрыЗапуска** – <Структура>, в данном параметре можно передать структуру. Все значения структуры поместятся в контекст с соответствующим ключем и значением и будет храниться в контексте. Один из вариантов использования – для заполнения параметров шаблона текста.

**СценарийСсылка** - Обычно этот идентификатор представляет собой URL-адрес сценария, расположенного на сервере, либо имя файла, содержащего сценарий, хранящегося локально.

*Пример:*

*Vots1C\_Служебный.ЗапуститьСценарий(КонтактЧатБота, СценарийСсылка, УстановитьШаг, ПараметрыЗапуска);*

## **8. Процедура ЧатБотAPI. ЗаписатьПользовательскиеДанныеВКонтекст(КонтактЧатБота, СценарийСсылка, КлючШаг, Данные);**

Данный метод позволяет записать любые временные данные в контекст сценария и доступны пока сценарий не завершен. К примеру, можно туда записать список номенклатур, которые подходят под отбор, или сумму корзины, или список корзины и т.д. Это одна из главных функций, которая позволяет записывать, хранить и получать результат вычислений в рамках контекста.

Где параметры:

**КонтактЧатБота** – см. в пункте 1.

**КлючШаг** – <Строка> Ключ данных

**Данные** – <Любой тип> Любые данные, строка, массив, таблица значений и т.д.

**СценарийСсылка** - Обычно этот идентификатор представляет собой URL-адрес сценария, расположенного на сервере, либо имя файла, содержащего сценарий, хранящегося локально.

*Пример:*

*Vots1C\_Служебный.ЗаписатьПользовательскиеДанныеВКонтекст(КонтактЧатБота, СценарийСсылка, КлючШаг, Данные);*

### 9. Функция ЧатБотAPI. ПолучитьПользовательскиеДанныеИзКонтекста(КонтактЧатБота, СценарийСсылка, КлючШаг);

Данная функция является обратной к ЗаписатьДанныеВКонтекст, и позволяет получить записанные ранее данные.

*Пример:*

*Bots1C\_Служебный.ПолучитьПользовательскиеДанныеИзКонтекста(КонтактЧатБота, СценарийСсылка, КлючШаг);*

### 10. Процедура ЧатБотAPI. ЗаписатьНастройкуПользователя(РазрезПараметра, Значение);

Записывает любую пользовательскую настройку, позволяя не «хардкодить» в коде. К примеру, количество позиций для вывода, или данные внешней системы, и т.д..

Где параметры:

**РазрезПараметра** - означает конкретный параметр, который будет установлен или изменен для определенного пользователя.

**КлючПараметра** – <Строка> Ключ параметра из таблицы с параметрами

**Значение** – Значение параметра.

*Пример:*

*Bots1C\_Служебный.ЗаписатьНастройкуПользователя(РазрезПараметра, Значение);*

### 11. Процедура ЧатБотAPI. ПолучитьНастройкуПользователя(РазрезПараметра, ЗначениеПоУмолчанию);

Служит для получения пользовательской настройки. Является обратной к методу в п.10.

Где параметры:

**РазрезПараметра** - означает конкретный параметр, который будет установлен или изменен для определенного пользователя.

**ЗначениеПоУмолчанию** – Если значение не найдено, какое значение выводить, позволяет избежать лишнего дальнейшего условия

*Пример:*

*Bots1C\_Служебный.ПолучитьНастройкуПользователя(РазрезПараметра, ЗначениеПоУмолчанию);*

Работа с контактами:

### 12. Функция ПолучитьКонтактЧата(ЧатБотСсылка, IDЧата)

Метод получает структуру с данными чата, где его наименование, номер телефона, контрагент и т.д.

Где параметры:

**IDЧата** - <Число, Строка> ID пользователя во внешней системе.

**ЧатБотСсылка** - <Ссылка> на справочник чат ботов.

*Пример:*

*Bots1C\_Служебный.ПолучитьКонтактЧата(ЧатБотСсылка, IDЧата);*

### 13. Функция ПолучитьКонтрагентЧата(ЧатБотСсылка, IDЧата)

Метод получает ссылку элемента справочника связанного с контактом, это может быть контрагент, партнер, сотрудник или любая другая ссылка.

Где параметры:

**ЧатБот, IDЧата** – см. в пункте 1.

Возвращаемое значение: <Структура>

**Подписан** – <Булево> Признак подписан ли клиент на бота, не всегда может быть корректным

**Представление** – <Строка> Представление пользователя, может быть Имя и Фамилия или логин

**НомерТелефона** – <Строка> Номер телефона

**СсылкаНаАватар** – <Строка> URL ссылка на картинку аватара

**Контрагент** – <ЛюбаяСсылка > Ссылка на контрагента, партнер и т.д. если удалось связать

*Пример:*

*Контакт = ЧатБотAPI.ПолучитьКонтактЧата(ЧатБот, "1861315")*

### 14. Функция ПолучитьКонтактыЧатовПоКонтрагенту(Контрагент)

Метод получает все контакты чатов, связанных с переданным контрагентом (партнером, сотрудником).

Применяться может к примеру для получения контактов контрагента для рассылки.

Где параметры:

**Контрагент** – Ссылка на справочник элеменет.

Возвращаемое значение: <Структура>

*Пример:*

*Bots1C\_Служебный.ПолучитьКонтактыЧатовПоКонтрагенту(Контрагент);*

#### **15. Процедура УстановитьНомерТелефонаКонтакта(КонтактЧатБота, НомерТелефон, АвтопоискКонтрагента)**

Необходим для установки данных контакта

Где параметры:

**КонтактЧатБота** - см. в п.1

**НомерТелефона** – <Строка> Номер телефона

**АвтопоискКонтрагента**– <Булево> Указывает будет ли выполнен поиск контрагента по номеру телефона

*Пример:*

*Контакт = ЧатБотAPI. УстановитьНомерТелефонаКонтакта(КонтактЧатБота, НомерТелефона, АвтопоискКонтрагента)*

#### **16. АвтоматическиСоздатьКонтактЧатБотаПоНомеруТелефона(НомерТелефона, ВЧатБотах)**

Функция - Автоматически создать контакты чат ботов по номеру телефона в случае отсутствия

*Пример:*

*Bots1C\_Служебный.АвтоматическиСоздатьКонтактЧатБотаПоНомеруТелефона(НомерТелефона, Контрагент, ВЧатБотах);*

Механизмы сотрудника:

#### **16. Процедура НачатьДиалогССотрудником(КонтактЧатБота, ПользователиДиалога, ОписаниеОшибки);**

Переводит клиента в режим диалога с сотрудником. Для перевода необходим установленный в настройках сценарий диалога с сотрудником. Если клиент уже в режиме диалоге – ничего не происходит.

Где параметры:

**КонтактЧатБота** - см. в п.1

**ПользователиДиалога** - пользователь ИБ, для которого установить диалог. Если не передан – до будет выбран последний сотрудник который общался с клиентом.

**Описание ошибки** – описание ошибки

*Пример:*

*Bots1C\_Служебный.НачатьДиалогССотрудником(КонтактЧатБота, ПользователиДиалога, ОписаниеОшибки);*

#### **17. Процедура ЗавершитьДиалогССотрудником(КонтактЧатБота, ОписаниеОшибки);** Завершает диалог с сотрудником, закрывая текущий шаг с диалогом и переводя на следующий шаг в сценарии. Если это последний шаг в сценарии – то будет запущен сценарий с главным меню. Если клиент не находился в режиме диалога с сотрудником – ничего не произойдет.

Где параметры:

**КонтактЧатБота** - см. в п.1

*Пример:*

*Bots1C\_Служебный.ЗавершитьДиалогССотрудником(КонтактЧатБота, ОписаниеОшибки);*

#### **18. Процедура ПеревестиДиалогНаСотрудника(КонтактЧатБота, НаПользователяИБ, СообщениеПользователю);**

Данный метод позволяет перевести клиента на другого сотрудника. Сотруднику могут быть перечислены через запятую (без пробела). Если параметр сообщение заполнено – то переведенному пользователю оно будет отображено.

Где параметры:

**КонтактЧатБота** - см. в п.1

**НаПользователяИБ** – <Строка> Имя пользователя, или имена через запятую, как задано в конфигураторе, которому перенаправить диалог и клиента, если будет Неопределенно – то оповещение будет всем.

**Сообщение** – <Строка> Произвольный текст сообщения.

Пример:

*Процедура ПеревестиДиалогНаСотрудника(КонтактЧатБота, НаПользователяИБ, СообщениеПользователю = "")*

**19. Процедура ОповеститьСотрудника(КонтактЧатБота, Тэг, ПользовательИБ, Заголовок, ,  
ИмяКартинкиИзБиблиотеки, ДополнительныеДанные, ОповеститьВ, ОповещатьКаждыеСекунд,  
ПриАктивизацииПереноситьНа )**

Создает оповещение сотруднику, (у данного сотрудника должен быть открыт мессенджер чат-ботов). Если IDЗаписиСообщения заполнен, при нажатии на будет переход в мессенджер.

Где параметры:

<b>КонтактЧатБота</b>	-	-
<b>Тэг</b>	-	-
<b>ПользовательИБ</b>	-	-
<b>Заголовок</b>	-	-
<b>ТекстСообщения</b>	-	-
<b>ИмяКартинкиИзБиблиотеки</b>	-	-
<b>ДополнительныеДанные</b>	-	-
<b>ОповеститьВ</b>	-	Если неопределено - то оповестится сразу
<b>ОповещатьКаждыеСекунд</b>	-	Если >0 - после оповещения переносится следующее оповещение на сек, если 0 - остается с активным оповещением
<b>ПриАктивизацииПереноситьНа</b>	-	Если >0 - переносится на сек, если 0 - остается активно без переноса, если -1 – удаляется

Пример:

*Процедура ОповеститьСотрудника(КонтактЧатБота, Тэг = Неопределено, ПользовательИБ = Неопределено, Заголовок = Неопределено, ТекстСообщения = Неопределено, ИмяКартинкиИзБиблиотеки = Неопределено, ДополнительныеДанные = Неопределено, ОповеститьВ = Неопределено, ОповещатьКаждыеСекунд = Неопределено, ПриАктивизацииПереноситьНа = -1)*

**20. Функция НайтиПользователейОтдела(ИмяОтдела)**

Данный метод позволяет получить всех пользователей, как они указаны в настройках, по имени отдела.

Где параметры:

**ИмяОтдела** – Имя отдела.

**НаПользователяИБ** – <Строка> Имя пользователя, или имена через запятую, как задано в

Пример:

*ЧатБотAPI.НайтиПользователейОтдела(«ОТДЕЛ\_ПРОДАЖ»)*

**21. Функция ОпределитьОтделПользователя(ИмяПользователя)**

Данный метод возвращает отдел в который входит пользователь. Если отделов несколько – возвращается только первый найденный, если не найдено – Неопределенно.

Где параметры:

**ИмяПользователя** – <Строка> Имя пользователя, как задано в конфигураторе

Пример:

*ЧатБотAPI.ОпределитьОтделПользователя(«AlexDimitrov»)*