

СОДЕРЖАНИЕ

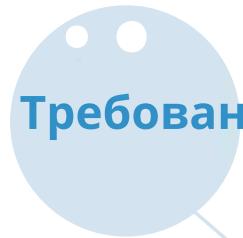
IFORM	• 01
КРАТКИЙ СЛОВАРЬ ИСПОЛЬЗУЕМЫХ ТЕРМИНОВ	• 01
ТРЕБОВАНИЯ	• 02
ПРИНЦИП РАБОТЫ	• 02
ОСНОВНЫЕ ВОЗМОЖНОСТИ	• 03
ПРЕДВАРИТЕЛЬНОЕ ТЕСТИРОВАНИЕ	• 04
ПОДКЛЮЧЕНИЕ	• 05
ЭКСПРЕСС ПОДКЛЮЧЕНИЕ	• 06
ВАРИАНТЫ ПОДКЛЮЧЕНИЯ ФОРМ	• 07
ДРУГОЙ ВИД РАЗМЕТКИ	• 07
РАЗМЕТКА ЧЕРЕЗ КЛАССЫ	• 08
ПРОДВИНУТЫЕ ВОЗМОЖНОСТИ IFORM	• 09
ПЕРЕДАЧА КОДА СТРАНЫ	• 10
ИСПОЛЬЗОВАНИЕ УДОБНОГО ВРЕМЕНИ	• 11
ИСПОЛЬЗОВАНИЕ ЭЛЕКТРОННОЙ ПОЧТЫ	• 12
ИГНОРИРОВАНИЕ ЭЛЕМЕНТОВ	• 13
ДОПУСТИМЫЕ ЗНАЧЕНИЯ	• 14
СПИСОК ОБЪЯВЛЕННЫХ АТРИБУТОВ	• 15
ОСОБЕННОСТИ УСТАНОВКИ АТРИБУТОВ	• 15
ПРИОРИТЕТЫ ВЫБОРКИ ПО УБЫВАНИЮ	• 16
СЦЕНАРИЙ ДЛЯ ОТПРАВКИ ДАННЫХ	• 17
АНТИСПАМ	• 18
РУКОВОДСТВО ОПЕРАТОРУ «КОЛЛ-ЦЕНТРА»	• 18
ПАРАЛЛЕЛЬНЫЙ ДОЗВОН, ВЫБОР ОФИСА И IFORM	• 19
ВОЗМОЖНОСТЬ ОГРАНИЧИТЬ АВТОМАТИЧЕСКОЕ ПОЯВЛЕНИЕ ИНСТРУМЕНТОВ ПРИ РАБОТЕ С IFORM	• 19

• • •

iForm — один из вариантов использования возможности инструмента обратного звонка. Инструмент iForm позволяют собирать данные и переводить заявки в звонки с уже существующих форм на сайте. Форма — это набор полей и элементов для ввода информации посетителем на сайте. Чаще всего на сайте уже существует некоторый набор форм.

Краткий словарь используемых терминов

- Тег — единичный элемент языка разметки, который обязательно имеет своё название, которое определяет его поведение и внешний вид;
- Открывающий тег — такая часть тега, которая находится между «первыми» двумя угловыми скобками;
- Закрывающий тег — такая часть тега, которая находится между «вторыми» двумя угловыми скобками;
- Атрибут — любое значение, которое находится внутри открывающего тега;
- Селектор — набор атрибутов, которые могут указать на элементы или список элементов;
- Class — класс, атрибут, который содержит список классов, к которым принадлежит данный элемент;
- id — идентификатор, уникальный атрибут для каждого элемента на странице;
- Метки — общее понятие об атрибуатах тегов;
- Асинхронный способ отправки — метод, который позволяют отправить данные, не влияя на общую работу страницы (обычно используется);
- Мгновенный синхронный способ — метод, который позволяют отправить данные почти мгновенно (используется только в определённых условиях);
- Событие подтверждения — это такая сущность внутри языка JavaScript, которая генерируется перед отправкой формы (влияет на отправку заявок с сайта);
- Проговариваемые параметры – значения тегов, отмеченные определенными атрибутами, которые будут проговорены роботом при поступлении звонка.



Требования

- 01.
- 02.

Сайт с установленным кодом CallKeeper

Положительный баланс минут на вашем аккаунте



Принцип работы

После подключения на странице будет доступен специальный объект JavaScript — ckForms.

Такой объект позволяет отправлять данные в CallKeeper. После отправки происходит мгновенное соединение вашего колл-центра и посетителя, который заполнил форму.

Если форма заполнена в нерабочее время, то звонок встанет в очередь «ночных звонков» и будет инициирован в ближайшее рабочее время (данные берутся из настроек инструмента обратного звонка).

Ниже описаны этапы работы с ним, от простого к сложному.

Инструмент iForm позволяет обрабатывать номера телефонов с любых форм на сайте. Позволяет выбирать удобное время для звонка. Позволяет указать название формы, имя посетителя и два других параметра, которые оператор «колл-центра» может прослушать во время звонка



Основные возможности

- Объект ckForms может собирать данные с любой формы, где есть поле для ввода телефона;
- Полная статистика по визиту посетителя, который заполнил форму;
- Возможность запускать звонки с форм, которые были заполнены в нерабочее время;
- Возможность совершать звонки сразу с нескольких форм, расположенных на одной странице;
- Возможность выбора номера телефона, с которым соединяют посетителя сайта;
- Во время звонка, синтезатор речи может проговорить следующие данные: имя посетителя, название формы и два любых дополнительных параметра;
- При необходимости можно полностью отключить проговаривание данных с формы;
- Проверка корректного ввода номера посетителя перед звонком;
- Антиспам: один посетитель не может совершить звонок с формы чаще, чем 1 раз в 5 минут;
- Возможность выбрать удобное время для звонка с формы;
- Возможность выбрать удобный способ отправки данных;

Предварительное тестирование

Все перечисленные возможности служат для того, чтобы произвести звонок с формы на сайте. Любая форма может быть конвертирована в звонок.

После того как все требования были выполнены, можно начинать тестирование. Для этого нет необходимости связываться с менеджером CallKeeper. Любое тестирование можно провести самостоятельно. Оно выглядит, как заказ звонка. Чтобы убедиться, что всё работает, просто закажите звонок.

Когда первые попытки дали неудовлетворительные результаты, нет необходимости связываться с менеджером CallKeeper. Это займет слишком много времени. Сначала подрядчики сообщают проблему менеджеру по маркетингу, тот в свою очередь связывается с менеджером CallKeeper, который в свою очередь пишет письмо в техподдержку. Когда техподдержка примет решение, делегирование пойдёт по обратному кругу. Это слишком долго.

Итак, возникли сложности. Нужно ещё раз обратиться к документации CallKeeper. Там, вероятно, уже есть ответ. Обычно пропущена скобка или последняя буква в названии атрибута или класса, или нарушены неочевидные правила, описанные в документации. По поводу её улучшения вы всегда можете написать в CallKeeper.

Итак, тестирование. Единственный возможный вариант, когда всё работает, и не осталось вопросов, это если происходит звонок после отправки формы. Отправляете форму – происходит звонок, только в этом случае всё хорошо. В обратном случае – что-то пошло не так, и лучше ещё раз вернуться к документации.

Подключение

Необходимо, чтобы все элементы для ввода данных, такие как `select` и `input`, и остальные, находились внутри тега `<form>` в соответствии со стандартом HTML5.

Кнопка отправки данных должна быть обрамлена тегом `<input>` или тегом `<button>`. С указанными внутри них типами `type="image"` или `type="submit"` или `type="button"`.

И ни в коем случае кнопка не должна содержать атрибутов типа: `onclick="return false"` или `onsubmit="return false"`, или подобных, приводящих к остановке обработки события.

Если данные настройки сделаны и подтверждены со стороны технических специалистов, то менеджер должен в панели администратора включить все формы. В этом случае активируется работа инструмента iForm.

Все остальные методы являются надстройками над общим подключением. Они необходимы для более точной обработки форм, и применяются лишь в отдельных случаях.

Экспресс подключение

Данный метод использует устаревшие технологии, но является наиболее отказоустойчивым, поэтому он приведён первым. Существует метод ckForms.trigger, который позволяет позвонить с любой формы, если соблюдены HTML стандарты.

```
1 <form onsubmit="ckForms.trigger(event)">
2   <input name="tel">
3   <input type="submit" value="Отправить">
4 </form>
```



В самой форме указан вызов метода по событию. В него обязательно передать объект event. При отправке такой формы произойдёт звонок на номер указанный в одном из полей формы. Возможно, установить такую же функцию обработчик и для нажатия на клавишу Enter. Она уже готова к такому использованию и можно указать любое событие keyup, keypress, keydown. Но лучше использовать keyup.

```
1 <form onsubmit="ckForms.trigger(event)">
2   <input name="tel">
3   <input type="submit" value="Отправить">
4   onclick="ckForms.trigger(event)"
5 </form>
```

Также можно использовать событие click, и установить обработчик прямо на кнопку. Можно не бояться укладывать обработчики по несколько штук, они защищены от повторения. Звонок будет только один.

Используется любое событие JavaScript описанное в стандарте в ECMAScript и специальный метод ckForms.trigger с параметром event.

Варианты подключения форм

Если тариф позволяет использовать возможности сбора номеров с форм, тогда возможен ещё один вариант их использования. Для его работы требуется активация модуля сборщика с форм. По умолчанию он включён.

Чтобы позволить сервису CallKeeper производить звонки на номера посетителей сайта, которые оставляют заявки, достаточно поместить класс CKi-Form в тег form. Вышеуказанный класс обязательно должен присутствовать у формы.

```
1 <form class="CKiForm">
2   <input type="submit" value="Отправить">
3 </form>
```

Другой вид разметки



```
1 <form class="CKiForm">
2   <input name="name">
3   <input name="tel">
4   <input type="submit" value="Отправить">
5 </form>
```

Форма реагирует на нажатие кнопки, точнее на событие submit, которое создаёт браузер перед отправкой формы. Поэтому форма обязательно должна генерировать событие submit, чтобы отправка прошла успешно.

Кнопка отправки имеет type="submit" или type="image", поле мобильного телефона имеет type="tel", поле электронной почты имеет тип type="email".

Существует общая договорённость о зарезервированных словах для обозначения атрибута name у каждого элемента формы, для упрощения разметки. Для имени посетителя лучше использовать name="name" или name="person", для телефона лучше использовать name="tel" или name="phone", для электронной почты name="email". Никто не требует от разработчика дополнительной разметки и безукоризненного следования стандартам и договорённостям. Форма сможет превращать заявки в звонки в любом случае, если она имеет класс CKiForm.

Разметка через классы

В случае если форма не является тегом form или кнопка не является тегами button или input, можно использовать разметку классами. Это проще, чем стандартные возможности iForm и поэтому предпочтительнее.

Для использования разметки классами у контейнера формы, в котором находятся все остальные активные элементы формы, нужно указать класс CKiForm, этот класс является обязательным для разметки формы, без него работать не будет, остальные являются опциональным. Если кнопка, которая инициирует отправку данных, не является button или input, у кнопки нужно указать класс CKFormTrigger. При нажатии на неё форма будет производить звонок.

```
1 <div class="CKiForm">
2   <input type="text" name="name">
3   <span href="#" class="CKFormTrigger">Отправить</span>
4 </div>
```

Этот способ позволяет сделать форму из любого тега на странице, который хранит поля ввода. Способ отправки и внутренняя структура формы не имеют значения. Важно только то, что кнопка должна находиться внутри кон-тейнера формы.

```
1 <div class=" CKFormTrigger">
2   <button>Отправить</button>
3 </div>
```

В приведённом примере кнопка находится внутри родительского тега с требуемым классом, это тоже сработает. Но инициирует событие элемент, находящийся глубже всех. Дальше он делегирует его своим родителям. Это делегирование в браузерах иногда даёт неожиданные результаты. Поэтому нужно быть осторожными в таких случаях. При использовании разметки через классы, виджет не будет автоматически появляться пока форма заполняется.

Используются классы CKFormTrigger и CKiForm

Продвинутые возможности iForm

Если необходимо, чтобы звонок с формы был соединен с конкретным подразделением, в этом случае потребуется соблюдать определённые правила разметки.

Более сложный вариант разметки формы:

```
1  <form class="CKiForm">
2      <input name="person">
3      <input name="phone">
4      <select>
5          <option value="0(000)000-00-00">
6              Дилерский центр 1
7          </option>
8          <option value="1(111)111-11-11">
9              Дилерский центр 2
10         </option>
11
12     </select>
13     <input type="image" value="Отправить">
14 </form>
```



Такой вариант разметки позволяет посетителю сайта выбрать номер телефона, на который поступит звонок. Для этого потребуется стандартный тег select. В нем находятся названия подразделений и номера телефонов. Каждый тег option, вложенный в тег select, хранит в атрибуте value номер телефона, к примеру: value="0(000)000-00-00". В таком случае, при выборе подразделения, звонок поступит именно туда.



Передача кода страны

- **data-callkeeper_countryCode**

С некоторых пор наш сервис работает и за рубежом. Поэтому крайне желательно передавать код страны явно. То есть нельзя гарантировать правильную работу сервиса, если в номере телефона отсутствует телефонный код страны. В целом, всегда передавать телефон в полном виде – это хорошая практика.

В случае если невозможно в одном поле хранить полный номер, и его обязательно нужно разбивать на несколько, тогда возможно хранить в определённом поле ввода код страны.

По умолчанию считается, что звонок пройдёт на территории Российской локации, и кодом страны будет считаться «+7».

В случае если код отличен от предустановленного, тогда он будет выбран из набора допустимых кодов стран, сохранённых в виджете. Следует обязательно согласовывать с менеджером данный параметр. Это может выглядеть так:

```
1 <input type="hidden" value="+52">>
```



Во избежании ошибочных случаев можно использовать специальный атрибут, который явно укажет, что в этом поле находится код страны. Возможны варианты:

```
1 <input type="text" data-callkeeper_countryCode value="45">
```

```
1 <input type="hidden" data-callkeeper_countryCode="+7">
```

- для хранения обязательно использовать тег input
- наличие плюса перед кодом не влияет на обработку
- значение атрибута type не влияет на обработку
- используется data-callkeeper_countryCode
- допустимые коды стран должны быть согласованы

Использование удобного времени

- **data-callkeeper="nice_time"**

В случае если форма заполняется в нерабочее время или посетители сайта хотят, чтобы звонок произошел позже, или в любом другом случае, когда требуется отложить звонок, можно дать посетителю возможность выбрать удобное время.

Разметка выбора удобного времени:

```
1 <form>
2   <select>
3     <option value="15">15:00</option>
4     <option value="11">11:00</option>
5     <option value="20">20:00</option>
6   </select>
7   <input type="image" value="Отправить">
8 </form>
```

Для упрощения процесса выбора времени используется ряд вещей. Первое — это контейнер в виде тега form. Второе — это набор элементов типа option внутри тега select, то есть обычный выпадающий список.

Каждый элемент option должен в атрибуте value хранить комбинацию из двух цифр - значение часа для звонка. Значения не могут быть меньше 0 или больше 24, то есть возможными значениями является любой час в сутках.

- для хранения обязательно использовать тег select
- ключевое значение имеет содержание атрибута value

В случаях, когда возникают проблемы с определением удобного времени, следует использовать специальный атрибут **data-callkeeper="nice_time"**. Он явно укажет на содержимое элемента.

Может выглядеть так:

```
1 <input type="hidden" data-callkeeper="nice_time" value="23">
```

```
1 <select data-callkeeper="nice_time">
2   <option value="20">20:00</option>
3 </select>
```

Использование электронной почты

- **data-callkeeper_email**

Если необходимо отправить письмо с результатом звонка на определённый почтовый ящик, тогда возможно разметить специальный элемент атрибутом data-callkeeper_email.

Данное поле содержит адрес или адреса электронной почты, на которые поступит письмо с результатом звонка, данный атрибут имеет самый высокий приоритет из возможного выбора адресов.

На странице это выглядит так:

```
1 <select>
2   <option data-callkeeper_email="adress@your.mail">
3     Дилерский центр 1
4   </option>
5   <option data-callkeeper_email="adress@your.mail">
6     Дилерский центр 2
7   </option>
8 </select>
```

```
1 <input data-callkeeper_email value="adress@your.mail">
```

```
1 <input data-callkeeper_email="adress@your.mail">
```

- письма о результатах звонков распределяются согласно офисам, указанным для них в виджете
- для iForm можно передать адреса почты явно
- письма различаются на аудитории: успешные, неуспешные, по офисам и для всех
- для хранения обязательно использовать тег select или input

В случае, когда требуется отправлять уведомления на несколько адресов одного дилерского центра, следует воспользоваться конструкцией вида:

```
1 <input data-callkeeper_email="["rere@rrew.ru","fdgsd@fe.ru"]">
```

Игнорирование элементов

- **data-callkeeper_ignore**

Любой элемент внутри формы можно исключить из сбора. Для этого нужно указать специальный атрибут, чтобы сборщик не учитывал данное поле при создании звонка.

```
1 <input type="text" data-callkeeper_ignore value="45">
```

Любой элемент формы, размеченный таким образом, станет невидимым для сборщика. Часто элементы на странице имеют вложения.

Данная возможность работает только для элемента, у которого указан атрибут, но не для его вложения. Следует быть осторожным и проверить работу

Так же не учитываются при поиске телефона посетителя, его электрон-ной почты и имени поля с типами hidden, submit, image, checkbox, button, любой элемент, не имеющий атрибута value и не имеющий HTML содержания внутри.

Крайне важное замечание по поводу расположение всех элементов, среди которых будет производиться поиск. Они должны находиться внутри контейнера. Если это не так, то предусмотрены отдельные решения для подобных случаев.



Допустимые значения

Менеджеры CallKeeper могут установить оптимальные настройки форм своими руками, без дополнительных действий со стороны клиента.

Поле iForm находится на странице настроек виджета. Состояние этого поля регулирует алгоритм работы iForm. Оно может принимать следующие значения:

- формы отключены совсем, iForm не реагирует на любые метки на странице;
- формы включены, iForm реагирует только на метки на странице, использует асинхронный способ отправки, не останавливает событие подтверждения;
- сбор со всех форм на сайте, с остановкой события подтверждения, использует асинхронный способ отправки данных;
- сбор со всех форм на сайте, без остановки события, использует синхронную отправку на выходе;
- сбор со всех форм на сайте, без остановки события, использует мгновенную синхронную отправку;
- серверная интеграция, формы не отправляются;
- сбор со всех форм, асинхронный, без остановки события;
- сбор со всех форм, звонок происходит, когда посетитель сайта ввёл последний символ своего телефонного номера
Не рекомендуется для использования;
- сбор со всех форм, используются устаревшие синхронные обработчики. Наиболее отказоустойчивый метод;

Список объявленных атрибутов

data-callkeeper_name="Форма" — название формы
data-callkeeper_tel="9991112222" — телефон компании
data-callkeeper="tel" — номер телефона посетителя
data-callkeeper_office="0" — индекс офиса в виджете
data-callkeeper="person" — имя посетителя
data-callkeeper_email — почта, на которую придёт уведомление
data-callkeeper="nice_time" — удобное время звонка
data-callkeeper_param="1" — любая текстовая информация
data-callkeeper_param="2" — любая текстовая информация
data-callkeeper_countryCode — телефонный код страны

Особенности установки атрибутов

Атрибуты для номеров телефонов `data-callkeeper="tel"`, `data-callkeeper_tel="9991112222"` могут быть указаны в любом формате, и содержать любые символы. Вместо `data-callkeeper_tel` можно использовать атрибут `data-callkeeper_office`. В качестве значения этого параметра указывается индекс офиса в виджете, на который будет заказан звонок. Индексирование офисов начинается с 0.

Эти поля проходят валидацию перед отправкой. Также возможно указать добавочный номер менеджера в «колл-центре» в формате: «телефон»^«добавочный».

Такие атрибуты как `data-callkeeper_name="Форма"`, `data-callkeeper="person"`, `data-callkeeper_param="1"`, `data-callkeeper_param="2"` являются проговариваемыми. Это значит, что оператор «колл-центра» услышит то, что они содержат. Почти все атрибуты являются неизменяемыми, то есть их нужно повторить знак в знак с учётом регистра. И только `data-callkeeper_tel="9991112222",data-callkeeper_name="Форма"` могут изменять своё значение.

Полностью размеченная форма может выглядеть так:

```

1 <form id="form1" data-callkeeper_name="чудесная форма">
2   <input name="name" data-callkeeper="person">
3   <input name="phone" data-callkeeper="tel">
4   <select name="brand" data-callkeeper_param="1">
5     <option data-callkeeper_tel="7(916)916-65-34">BMW</option>
6     <option data-callkeeper_tel="+7 111 111 1111">
7       Mercedes</option>
8     <option data-callkeeper_tel="+7 222 222-22-22">Honda</option>
9     <option data-callkeeper_tel="+7 333 333-33-33">Audi</option>
10    </select>
11    <select data-callkeeper_param="2">
12      <option>Кредит</option>
13      <option>Спецпредложение</option>
14      <option>Страховка</option>
15      <option>Тест-драйв</option>
16    </select>
17    <select data-callkeeper="nice_time">
18      <option value="15">16:15</option>
19      <option value="11">10:30</option>
20    </select>
21  <input type="submit" value="отправить" onsubmit="ckForms.send('#form1'); return false;">
22 </form>
23
24 Или так:
25
26 <form id="form2" data-callkeeper_name="прекрасная форма" data-callkeeper_tel="7(916)916-96-18^12">
27   <input data-callkeeper="person">
28   <input data-callkeeper="tel">
29   <input type="image" value="отправить" on-click="ckForms.send('#form2'); return false;">
30 </form>

```

```

1 <form id="form1" data-callkeeper_name="чудесная форма">
2   <input name="name" data-callkeeper="person">
3   <input name="phone" data-callkeeper="tel">
4   <select name="brand" data-callkeeper_param="1">
5     <option data-callkeeper_office="1">BMW</option>
6     <option data-callkeeper_office="2">
7       Mercedes</option>
8     <option data-callkeeper_office="3">Honda</option>
9     <option data-callkeeper_office="4">Audi</option>
10    </select>
11    <select data-callkeeper_param="2">
12      <option>Кредит</option>
13      <option>Спецпредложение</option>
14      <option>Страховка</option>
15      <option>Тест-драйв</option>
16    </select>
17    <select data-callkeeper="nice_time">
18      <option value="15">16:15</option>
19      <option value="11">10:30</option>
20    </select>
21  <input type="submit" value="отправить" onsubmit="ckForms.send('#form1'); return false;">
22 </form>
23
24 Или так:
25
26 <form id="form2" data-callkeeper_name="прекрасная форма" data-callkeeper_tel="7(916)916-96-18^12">
27   <input data-callkeeper="person">

```

01. Порядок выбора поля с номером телефона менеджера:

- при включенной интеграции с сервисами «коллтрекинга» берется подставной номер;
- ищется атрибут data-callkeeper_tel в тегах option внутри тега select;
- ищется атрибут data-callkeeper_tel в любых элементах внутри контейнера формы;
- берется номер/номера из настроек вида CallKeeper;

02. Порядок выбора поля с номером телефона клиента:

- ищется элемент с атрибутом data-callkeeper="tel";
- ищется тег input с атрибутом type="tel";
- ищется тег input с атрибутом name="phone" или "tel";

03. Порядок выбора поля с именем клиента:

- ищется элемент с атрибутом data-callkeeper="person";
- ищется тег input с атрибутом name="name" или "person";

04. Порядок выбора поля с электронной почтой клиента::

- ищется элемент с атрибутом data-callkeeper="email";
- ищется тег input с атрибутом type="email";
- ищется тег input с атрибутом name="email";

Сценарий для отправки данных

Отправка данных из формы в звонок реализуется методом send объекта ckForms. Метод send принимает селекторы, элементы, массивы элементов или списки элементов, перечисленные через запятую.

Антиспам

При успешной отправке посетителю устанавливается cookie (ключ — селектор формы, переданный в ckForms.send(), значение — stop, время жизни — 5 минут), что не позволяет одному посетителю отправлять более одной формы чаще, чем 1 раз в течение 5 минут.

Также нельзя отправить одну и ту же форму дважды. Копия формы со-храняется, и второй раз её отправить уже нельзя. Нужно обязательно перезагрузить страницу.

Кроме того, на сервере CallKeeper существует «блэк-лист» с телефонами и ip-адресами недоброжелателей. Подобные списки позволяют обезопасить себя от «спам-атак».



Руководство оператору «колл-центра»

При поступлении звонка с формы в рабочее время, оператор услышит:

— Посетитель заполнил форму на сайте. Пожалуйста, нажмите 1 чтобы узнать подробности или нажмите 5 чтобы позвонить клиенту.

Далее оператор может нажать «5» чтобы совершить звонок сразу или нажать «1» чтобы прослушать данные, которые пришли из формы, в этом случае следующая фраза синтезатора речи будет такая:

— Посетитель сайта, [имя посетителя], заполнил форму [название формы]. [параметр1]. [параметр2]. Пожалуйста, нажмите 5 чтобы позвонить клиенту.

Эта фраза будет циклично повторяться 15 раз. Если со стороны оператора не будет нажата «5» в тоновом режиме, то звонок завершится автоматически.



Параллельный дозвон, выбор офиса и iForm

В случае, если номер менеджера передаётся явно, работа проходит штатно. Звонок происходит только на номер менеджера, чей номер был передан. Во всех остальных случаях работа iForm приравнивается к работе ви-джета без выбора офиса. И поведение зависит от настройки "параллельный дозвон" или "последовательный дозвон".

Параллельный дозвон производит звонок на все номера одновременно, до поднятия трубки, если номер не задан явно.

Последовательный дозвон перебирает по очереди все доступные номера в виджете, начиная с первого, если номер не задан явно.

В iForm обычно нет выбора офисов и процесс звона может быть неконтролируемым. Поэтому следует уделить повышенное внимание вопросу отправления звонка на определённый номер.



Возможность ограничить автоматическое появление инструментов при работе с iForm

В случае если необходимо ограничить автоматическое появление инструментов (например виджетов) во время заполнения формы можно воспользоваться разметкой классами (пункт 11) или специальными атрибутами:



1. Атрибут `data-lock_timeout` - ограничит появление виджета во время того как будет заполняться любое поле формы, при прекращении взаимодействия с формой появление будет перепланировано согласно настройкам. Данный атрибут можно использовать с любым элементом формы, но рекомендуется воспользоваться им на контейнере, для того чтобы захватить всю форму.

2. Атрибут `data-clear_element` - данный атрибут можно повесить на подтверждающий элемент заполнения формы (например кнопку "отправить") для того чтобы совсем выключить появление виджета после нажатия на этот элемент.

Проговариваемые параметры



Шаблон проговариваемой информации роботом —
‘Посетитель заполнил форму %название формы% сайт %название
виджета%. пожалуйста, нажмите %цифра установленная для
соединения с клиентом% чтобы позвонить клиенту . %цифра
установленная для прослушивания дополнительной информации%
чтобы узнать подробности . %цифра установленная для отмены
соединения% чтобы прекратить вызов’.

Шаблон проговариваемой дополнительной информации — имя .
%data-callkeeper_person%. форма . %data-callkeeper_name%.%da-
ta-callkeeper_param='1'%. %data-callkeeper_param='2'%.
Вместо значений в %% будут подставлены данные которые
заполнены в полях формы и размеченные параметрами.

Также можно включить проговаривание информации сразу, без
нажатия цифры «1». Данный функционал настраивается отдельно.

