

ALTUERA

# Описание функциональных характеристик ПО «VideoChat»

Настоящий документ содержит конфиденциальную информацию и является собственностью ООО "Альтуэра". Копирование, распространение и воспроизведение, как полностью, так и частично, без письменного согласия со стороны ООО "Альтуэра" запрещено.



ALTUERA

# Содержание

<b>1. ВВЕДЕНИЕ</b> .....	<b>2</b>
1.1. Назначение документа .....	2
1.2. Назначение Системы .....	2
<b>2. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ</b> .....	<b>2</b>
1.3. Функциональность решения .....	2
1.3.1. Поддерживаемые Браузеры .....	4
<b>3. ОПИСАНИЕ ПРОЦЕССА ОБРАБОТКИ ВЗАИМОДЕЙСТВИЙ</b> .....	<b>4</b>
3.1. Описание потоков данных взаимодействий .....	6
<b>4. ОТКАЗОУСТОЙЧИВОСТЬ</b> .....	<b>7</b>
<b>5. БЕЗОПАСНОСТЬ</b> .....	<b>7</b>
5.1. Матрица внешних соединений .....	7
5.2. Защита клиент-серверных соединений .....	9
<b>6. НАЗНАЧЕНИЕ КОМПОНЕНТ VIDEOCHAT</b> .....	<b>9</b>
6.1. Компоненты VideoChat .....	9
6.2. Сторонние компоненты VideoChat .....	10
6.3. Матрица соединений компонент .....	10

# 1. Введение

## 1.1. Назначение документа

Содержание документа направлено на формирование у пользователя понимания возможностей и назначения ПО VideoChat.

## 1.2. Назначение Системы

VideoChat – это комплекс ПО, предназначенный для оперативной организации аудио-визуального взаимодействия между представителями Бизнеса и их Клиентами.

Отличительной особенностью ПО VideoChat является отсутствие необходимости установки какого-либо ПО на оборудование как Агентов, так и Клиентов. Для организации соединения достаточно современного браузера на любом мобильном или стационарном устройстве (смартфон, планшет, ноутбук, персональный компьютер) и доступа в сеть Интернет.

# 2. Назначение и условия применения

## 1.3. Функциональность решения

VideoChat предоставляет следующую функциональность для обслуживания аудио-визуальных взаимодействий, а также для управления решением:

- Отправка ссылки Клиенту для подключения к взаимодействию
  - Через СМС провайдера SMSC.ru
  - Через телеграм-бот
  - Через специально разработанные коннекторы по запросу Заказчиков
  - Возможность скопировать ссылку Клиента в буфер обмена
- Двусторонняя передача аудио между Агентом и Клиентом
- Передача и отображение видео между Агентом и Клиентом
  - Односторонняя Клиент->Агент
  - Двусторонняя
- Передача отображения указателя мыши на видео Клиента. Агент может указывать на объекты на видео Клиента
  - Визуализация «клика» мыши
  - Визуализация «шлейфа» мыши
- «зеркальное» отражение фронтальной камеры
- Установка максимального качества видео
  - Низкое (640x480)
  - Среднее (1280x720)
  - Высокое (1920x1080)

- 4K (3840x2160)
- Автоматический выбор качества видео в соответствии с пропускной способностью сети между клиентскими устройствами и сервером VideoChat
- Возможность удаленной демонстрации экрана (только для браузеров ПК)
- Возможность управления состоянием передаваемого видео
  - Переключение камеры (фронтальная, тыловая)
  - Включение/выключение видео
  - Включение/выключение аудио
- Двусторонний чат с возможностью передачи файлов (до 100 Мбайт)
  - передача файлов в чате (до 100 Мбайт)
- Снятие скриншотов с передаваемого и/или получаемого видео
  - Двусторонняя передача скриншотов
  - Редактирование (рисование) скриншота Агентом с последующей отправкой Клиенту
- Функциональность запроса геолокации
  - Отправка Клиентом своей геолокации в чат
  - Запрос Агентом геолокации Клиента
  - Автоматическое получение геолокации Клиента
  - Геокодирование (преобразование географических данных геолокации в текстовый адресе)
  - Формирование ссылки с точкой геолокации на Яндекс.карты
  - Отображение превью месторасположения в чате
  - Присоединение данных геолокации Клиента к сделанным скриншотам (EXIF)
- Создание дополнительных полей с данными взаимодействия
- Формирования поствызовного опроса Клиента
- Запись вызовов (аудио Клиента в левом канале, аудио Агента в правом канале, видео - \_только\_ Клиента)
- Планирование Агентом будущих взаимодействий (определенному Клиенту на определенное время)
  - Управление запланированными взаимодействиями административным персоналом (Администраторы, Супервизоры)
- Брендирования интерфейсов Агента и Клиента
  - Логотип компании
  - Название компании
  - Текст согласия Клиента на подключение
  - Автоматический переход на сайт по завершению взаимодействия
- Управление пользователями их ролями
- Управлять доступной функциональностью вызовов
- Управлять занимаемыми файлами и записями местом на дисках
  - Автоматизация их удаления по расписанию
- Статистическая и историческая информация о вызовах
  - Факт и данные вызовов
  - Временные показатели вызовов
  - Данные геолокации
  - История чата

- Данные поствызовного опроса
- Файл с записью вызова
- Данные об устройстве пользователей
- Выгрузка отчетности в формате excel

### 1.3.1. Поддерживаемые Браузеры

Браузер и версия	Android Mobile	Mo-	iOS Mobile	MacOS	Windows, Linux
Яндекс.Браузер (chromium 92+)	+		+	+	+
Google Chrome (chromium 92+)	+		+	+	+
Microsoft Edge (chromium 92+)	+		+	+	+
Apple Safari 14+	-		+	+	-
Samsung Internet Browser 15.1+	+		-	-	-

## 3. Описание процесса обработки взаимодействий

- 1) После успешной авторизации веб-интерфейс агента открывает websocket соединение с сервером VideoChat для асинхронного взаимодействия с платформой
- 2) Инициализация нового взаимодействия происходит вызовом метода REST API
- 3) Сервер VideoChat
  - a) Создает запись в БД о новом взаимодействии
  - b) Создает сущность конференц-комнаты на медиа-шлюзе janus-gateway и отправляет ответ на REST запрос об успешном создании взаимодействия
- 4) Интерфейс агента по факту успешного создания взаимодействия отправляет через вебсокеты команду серверу VideoChat на отправку сообщения Клиенту
- 5) Интерфейс агента открывает второй вебсокеты для передачи WebRTC сигнализации:

- a) происходит подключение браузера агента к конференц-комнате на медиа-шлюзе
- b) медиа-шлюз и браузер Агента обмениваются данными для подключения по протоколу ICE
- 6) WebRTC компонент браузера устанавливает RTCP соединение к медиа-серверу
- 7) Медиа-сервер устанавливает RTCP соединение к браузеру Агента транзитом через TURN сервер
- 8) По факту установки обоюдного соединения медиа-сервер начинает запись аудио Агента
- 9) Клиент получает ссылку от агента и открывает её в браузере
- 10) Браузер Клиент делает синхронный REST запрос к API, чтобы получить параметры брендирования страницы и настройки вызова
- 11) Браузер Клиента возвращает страницу подключения
- 12) После нажатия кнопки «Подключиться» браузер клиента открывает web-socket соединение с сервером VideoChat для асинхронного взаимодействия с платформой
- 13) Интерфейс Клиента открывает второй вебсокет для передачи WebRTC сигнализации:
  - a) происходит подключение браузера Клиента к конференц-комнате на медиа-шлюзе
  - b) медиа-шлюз и браузер Клиента обмениваются данными для подключения по протоколу ICE
- 14) WebRTC компонент браузера устанавливает RTCP соединение к медиа-серверу
- 15) Медиа-сервер устанавливает RTCP соединение к браузеру Клиента транзитом через TURN сервер
- 16) По факту установки обоюдного соединения медиа-сервер начинает запись аудио и видео Клиента
- 17) С этого момента медиа потоки данных, аудио и видео Клиента передаются на устройство Агента и наоборот.
  - a) Медиа поток данных используется для:
    - i) Передачи сообщений в чате
    - ii) Передачи сигнализации (запросов-ответов) непосредственно от браузера Агента браузеру Клиенту
    - iii) Передача координат курсора мыши
- 18) По факту завершения взаимодействия Агентом компонент vc-server генерирует событие об окончании взаимодействия, при получении которого
  - a) Компонент vc-stat обращается в БД, откуда выбирает все события данного взаимодействия, записанные туда медиа-шлюзом, высчитывает из них метрики данных подключения и записывает в БД
  - b) Компонент vc-record определяет наличие файлов записи вызовов и вызывает их обработку компонентом vc-mediamerge. По факту успешной обработки записи, vc-record загружает файл на vc-fileserver и записывает в БД данные о записи вызова

### 3.1. ОПИСАНИЕ ПОТОКОВ ДАННЫХ ВЗАИМОДЕЙСТВИЙ

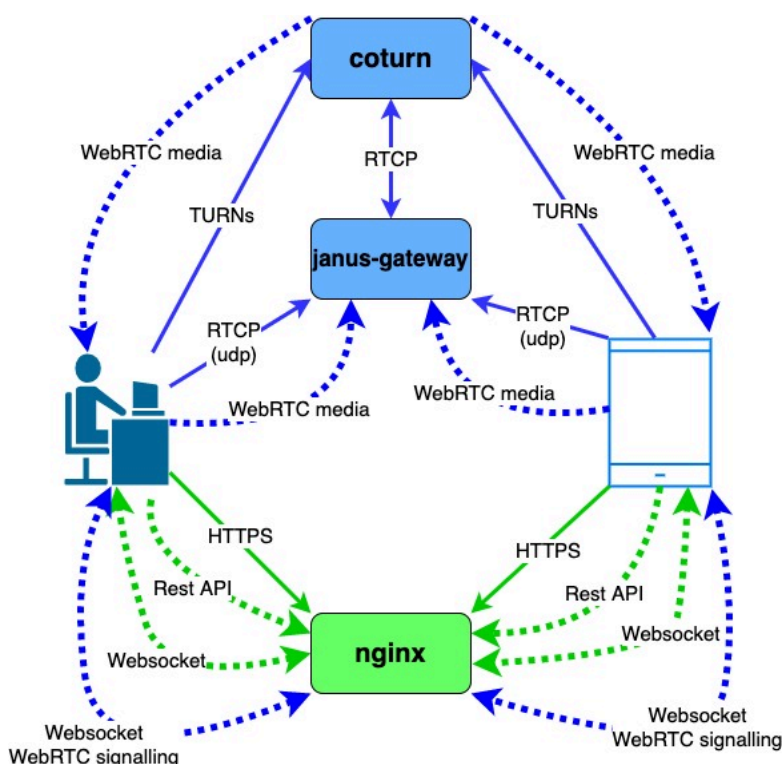
Непосредственного соединения устройств Агента и Клиента не происходит. Вся передача данных осуществляется «транзитом» через сервер VideoChat. Оба участника взаимодействия осуществляют подключение только к платформе VideoChat.

Пользователи подключаются к Системе VideoChat по протоколу HTTPS:

- Для вызовов методов синхронного REST API
- Основной вебсокет – для асинхронного API управления взаимодействием
- Дополнительный вебсокет – для передачи сигнализации WebRTC (управляющие команды, протокол ICE)

Для передачи медиа (аудио, видео, дата-канал):

- Подключение клиентских к TURN (coturn) серверу, чтобы обойти возможные ограничения NAT. По этому соединению на клиентские устройство поступают входящие потоки медиа данных
- Подключение клиентских устройств к медиа-серверу (janus-gateway). По этому соединению клиентские устройства отправляют исходящие потоки медиа данных



Сплошными линиями: сетевые соединения

Пунктирными линиями: потоки данных

## 4. Отказоустойчивость

Платформа VideoChat предоставляет возможность резервирования «cold-standby» - холодный запуск нового сервера VideoChat. Для поддержки режима резервирования cold-standby должны быть соблюдены следующие требования:

- СУБД должна работать в распределенном отказоустойчивом режиме.
- Хранилище файлов должно работать в распределенном отказоустойчивом режиме.

## 5. Безопасность

Специфика реализации протокола WebRTC в браузерах ТРЕБУЕТ обязательного использования безопасного контролирующего соединения с обязательной валидацией сертификатов. В связи с этим, все подключения к платформе VideoChat должны осуществляться по протоколу https и использовать подающиеся публичной верификации сертификаты. Взаимодействие локальных компонент VideoChat между собой осуществляется только на внутреннем (loopback) сетевом интерфейсе сервера и недоступно для подключения как из локальных, так из публичных сетей.

Инсталлятор также устанавливает и настраивает брандмауэр со следующими параметрами:

Зона **public** – источник: все публичные ip адреса вне зависимости от сетевого интерфейса. Разрешены только соединения в соответствии с разделом 3.3 [Матрица внешних соединений](#). Прочие соединения блокируются.

Зона **trusted** – источник: все приватные ip адреса (10/8, 172.16/12, 192.168/16) вне зависимости от сетевого интерфейса. Разрешены любые соединения.

### 5.1. Матрица внешних соединений

Вектор ствия	взаимодей-	Протокол	Адрес источ-	Адрес назна-	Порт назначе-
			ника	чения	ния
Администраторы	Сервер VideoChat	TCP (ssh)	На усмотрение администраторов сервера	Internal_ip, external_ip (опционально)	22
Агенты, Клиенты, Администраторы	Admin-panel, frontend agent, frontend user,	TCP (https)	Любой интернет адрес	external_ip	443



	frontend desk- top_user				
Агенты, клиенты	frontend agent, frontend user, frontend desk- top_user	TCP,UDP (STUN, TURNs сигнализа- ция)	Любой интер- нет адрес	external_ip	5349
Агенты, Клиенты	frontend agent, frontend user, frontend desk- top_user	UDP (TURN- server RTP, DTLS+STRP, DTLS+SCTP)	Любой интер- нет адрес	external_ip	20000- 29999 50000- 59999
Сервер VideoCha t	Шлюзы до- ставки со- общений	TCP (https)	internal_ip	Telegram Bot API ( <a href="https://api.telegram.org">https://api.tele- gram.org</a> ) SMSC.ru API ( <a href="https://smsc.ru">https://smsc.ru</a> )	443
Telegram Bot API	Сервер VideoChat	TCP (https)	<a href="https://api.telegram.org">https://api.tele- gram.org</a>	external_ip	443
Сервер VideoCha t	Сервис гео- кодирова- ния	TCP (https)	internal_ip	<a href="https://nomina-&lt;br/&gt;tim.open-&lt;br/&gt;streetmap.org/r&lt;br/&gt;everse">https://nomina- tim.open- streetmap.org/r everse</a>	443
Сервер Vide- oChat	Letsencrypt API	TCP (https)	Internal_ip	acme- v02.api.letsencr ypt.org	443
Letsen- crypt API	Сервер Vid- eoChat	TCP (http)	acme- v02.api.letsencr ypt.org	external_ip	80

## 5.2. Защита клиент-серверных соединений

Соединения пользователей с сервером VideoChat

Протокол	Порт	ПО	Комментарий
HTTPS	tcp 433	nginx	TLS v1.3
TURNs	tcp 3478, 5349	coturn	TLS v1.2 (сигнализация)
STUN	udp 3478, 5349	coturn	без шифрования согласно спецификации протокола STUN
RTCP	udp 20000-29000	coturn	RTCP+DTLS v1.2 (аудио, видео) SCTP+DTLC v1.2 (дата-канал)
RTCP	udp 50000-59999	janus-gateway	RTCP+DTLS v1.2 (аудио, видео) SCTP+DTLC v1.2 (дата-канал)

## 6. Назначение компонент VideoChat

### 6.1. Компоненты VideoChat

**vc-admin** – компонент VideoChat, предоставляющий API для конфигурации ПО

**vc-auth** – компонент VideoChat, отвечающий за аутентификацию пользователей и валидацию запросов к API

**vc-eventserver** – компонент VideoChat, агрегирующий статистические метрики WebRTC для системы мониторинга

**vc-fileserver** – компонент VideoChat, предоставляющий API для работы с файлами, а также осуществляющий контроль занятого файлами места на дисках и выполняющий регулярные работы по удалению старых файлов

**vc-geo** – компонент VideoChat, отвечающий за преобразования координат в текстовые данные геолокации с использованием внешних GEO сервисов

**vc-history** – компонент VideoChat, осуществляющий хранение и восстановление истории чатов, а также предоставляющий API для доступа к исторической отчетности

**vc-license** – компонент VideoChat, осуществляющий контроль лицензионных ограничений решения

**vc-mediamerge** – компонент VideoChat, отвечающий за конвертации аудио и видео записей взаимодействий в единый файл

**vc-msg** – компонент VideoChat, осуществляющий доставку ссылок до Клиентов и (опционально) контроль за доставкой

**vc-records** – компонент VideoChat, управляющий процессом конвертации и загрузки записей взаимодействий

**vc-server** – основной компонент VideoChat, осуществляющий: контроль жизненного цикла взаимодействий, контроль подключения пользователей к платформе, а также обеспечивающий взаимодействие пользовательских интерфейсов с прочими компонентами

**vc-stat** – компонент VideoChat, отвечающий за агрегацию статистических данных взаимодействий и предоставляющий API для доступа к исторической отчетности

**vc-statchart** – вспомогательный компонент VideoChat, предоставляющий возможность отображения информации о качественных показателях WebRTC для взаимодействий

**vc-surl** – компонент VideoChat, ответственный за укорачивание URL ссылок для отправки Клиентам

**frontend admin-panel** – административный веб-интерфейс системы, предоставляющий возможность осуществлять настройку, а также предоставляющий доступ к статистике

**frontend agent** – веб интерфейс Агента

**frontend user** – веб интерфейс Клиента для браузеров мобильных устройств

**frontend desktop\_user** – веб интерфейс Клиента для браузеров ПК

**storage** – хранилище файлов, (опционально) подключаемое к серверу и используемое компонентом **vc-fileserver**

## 6.2. Сторонние компоненты VideoChat

**activemq** – Брокер шины обмена данными между компонентами VideoChat по протоколу MQTT

**coturn** – STUN/TURN сервер, помогающий устанавливать WebRTC соединения Клиентов, находящихся за NAT, с сервером VideoChat

**janus** – медиа-шлюз, обеспечивающий установку и контроль WebRTC соединений и RTP медиа потоков для всех пользователей VideoChat

**logstash** – вспомогательный компонент VideoChat, осуществляющий сбор логирования с браузеров клиентов VideoChat в целях поиска ошибок

**mongod** – СУБД решения

**nginx** – веб сервер решения, являющейся единой точкой доступа ко всем пользовательским веб-интерфейсам, а также предоставляющий защищенный и авторизованный доступ веб-интерфейсов ко всем API решения.

**ansible** – система автоматического управления конфигурациями, используется для инсталляции, обновления и изменения конфигурации компонент VideoChat.

**java** – среда выполнения JAVA-приложений

**firewalld** – фреймворк ОС Linux для управления брандмауэром

**prometheus** – система хранения статистических метрик в timeseries database, используется как система мониторинга решения

**grafana** – система визуализации метрик Prometheus, используется как система мониторинга решения

**prometheus-node-exporter** – коллектор статистических метрик ОС для Prometheus

**ffmpeg** – набор утилит, кодеков и библиотек для работы с аудио и видео данными как в виде файлов, так в виде потоков в реальном времени

## 6.3. Матрица соединений компонент

Инициатор	Получатель	хост:порт	Протокол/тип соединения
nginx	vc-admin	127.0.0.1:9105	http
	vc-auth	127.0.0.1:9101	http
	vc-server	127.0.0.1:9103	http
	vc-msg	127.0.0.1:9106	http
	vc-fileserver	127.0.0.1:9108	http
	vc-stat	127.0.0.1:9104	http
	vc-startchart	127.0.0.1:9113	http
	vc-history	127.0.0.1:9107	http
	vc-surl	127.0.0.1:9102	http
	logstash	127.0.0.1:13001	http
grafana	127.0.0.1:3000	http	
vc-admin	mongodb	127.0.0.1:27017	mongo
vc-auth	mongodb	127.0.0.1:27017	mongo
	activemq	127.0.0.1:1883	mqtt
vc-eventserver	activemq	127.0.0.1:1883	mqtt
vc-fileserver	mongodb	127.0.0.1:27017	mongo
	activemq	127.0.0.1:1883	mqtt
vc-geo	activemq	127.0.0.1:1883	mqtt
vc-history	mongodb	127.0.0.1:27017	mongo
vc-license	activemq	127.0.0.1:1883	mqtt
vc-mediamerge	activemq	127.0.0.1:1883	mqtt
vc-msg	mongodb	127.0.0.1:27017	mongo
	activemq	127.0.0.1:1883	mqtt
vc-records	mongodb	127.0.0.1:27017	mongo
	activemq	127.0.0.1:1883	mqtt
	vc-fileserver	127.0.0.1:9118	http
vc-server	mongodb	127.0.0.1:27017	mongo
	activemq	127.0.0.1:1883	mqtt
vc-stat	mongodb	127.0.0.1:27017	mongo
	activemq	127.0.0.1:1883	mqtt
vc-startchart	mongodb	127.0.0.1:27017	mongo
vc-surl	activemq	127.0.0.1:1883	mqtt
	mongodb	127.0.0.1:27017	mongo
prometheus	prometheus- node-exporter	127.0.0.1:9090	http
	vc-eventserver	127.0.0.1:9095	http
grafana	prometheus	127.0.0.1:9100	http
janus-gateway	activemq	127.0.0.1:1883	mqtt
	vc-history	127.0.0.1:9117	http