

LiraX API für Telefonie

Allgemeine API-Informationen

Die API ermöglicht es, jedes Ihrer Softwareprodukte (externes Softwareprodukt) an LiraX anzubinden.

Die Integration mit LiraX löst mehrere Geschäftsaufgaben:

- Abrufen des Anrufernamens anhand der Telefonnummer;
- Abrufen der Nummer des Managers, der dem Kunden zugeordnet ist;
- Anzeige von Informationen über einen eingehenden Anruf direkt im externen Softwareprodukt;
- Speichern der Historie aller Anrufe und der Links zu Gesprächsaufzeichnungen im externen Softwareprodukt;
- Erstellen eines Kontakts;
- Erstellen einer Aufgabe für Kontakt und Manager;
- Erstellen einer Notiz für den Kontakt;
- Erstellen eines Geschäfts für den Kontakt;
- Aktualisieren eines Geschäfts
- Ausführen ausgehender Anrufe direkt aus der Oberfläche des externen Softwareprodukts.

Integration muss sein bidirektional. daher Teil Anfragen LiraX sendet in Seite externes Softwareprodukt an angegeben Einstiegspunkte, und Teil Anfragen, umgekehrt umgekehrt, externes Softwareprodukt sendet in Seite LiraX an einziger entry Punkt.

Die Interaktion erfolgt über das HTTPS-Protokoll. Die Autorisierung erfolgt entsprechend über die Adresse von LiraX bzw. des externen Softwareprodukts und den Autorisierungsschlüssel, der bei der Einrichtung der Integration erhalten wurde.

Prinzip der Autorisierung und Interaktion

HTTPS:

- Anfragen an LiraX werden nur über das HTTPS-Protokoll akzeptiert. This bietet ausreichend Sicherheit Niveau für Kommunikation Systeme over der Internet.
- für der Sicherheit von Ihr Daten, bitte, implementieren Empfang Anfragen an Seite externes Softwareprodukt also unter Verwendung von HTTPS.

Schlüssel (token):

- Zusätzlich für Autorisierung jede Anfrage an LiraX ist verwendet spezieller Schlüssel (token). empfangen it kann in Integrationseinstellungen.
- bitte, implementieren Empfang Anfragen an Seite externes Softwareprodukt also unter Verwendung von Schlüssel (token). Generieren Schlüssel und einfügen it in entsprechendes Feld in Abschnitt Integrationseinstellungen.
- Keys sind erstellt einmal wenn Konfiguration integration. If erforderlich, you kann ändern der Schlüssel an Ihr Seite und aktualisieren it in der Schnittstelle LiraX.

Anfragen von LiraX an das externe Softwareprodukt:

- Anfragen muss senden an Adresse in den Integrationseinstellungen angegeben für Empfang Benachrichtigungen, Beispiel <https://domain/LiraX>.
- in der Anfrage-Body in speziell Feld «token» muss immer sein übergeben spezieller Schlüssel.
- Anfragen werden im Format application/x-www-form-urlencoded übertragen

Anfragen vom externen Softwareprodukt an LiraX:

- externes Softwareprodukt muss senden alle Anfragen an Adresse in den Integrationseinstellungen angegeben
- in der Nachrichten-Body externes Softwareprodukt muss übergeben angegeben durch you in Web-Konto Schlüssel (token).

- Anfragen werden im Format application/x-www-form-urlencoded übertragen

Antworten

- alle Antworten an Anfragen LiraX externes Softwareprodukt muss senden in JSON Format in der Antwort Body.
- alle Antworten an Anfragen externes Softwareprodukt LiraX sendet in JSON Format in der Antwort Body.

Liste der API-Befehle

Vom externen Softwareprodukt an LiraX:

- makeCall (POST,GET)
- killCall (POST,GET)t
- make2Calls (POST)
- AskQuestion (POST)
- AddTag (POST)
- DelTag (POST)
- get_sip_route_in (POST)
- set_sip_route_in (POST)
- getUserSips (POST)
- IsFreeUsers (POST)
- IsCalling (POST)
- AddCampaign (POST)
- AddPhoneCampaign (POST)
- checkKontakt (POST)
- getKontakt (POST)
- getStatInfo (POST)
- createAufgabe (POST)
- AddAufgabeErgebnis (POST)
- GetAufgabe (POST)
- createHinweis (POST)
- createGeschäft (POST)
- updateGeschäft (POST)
- addBlackPhone (POST,GET)
- addBlackIP (POST,GET)
- getStages (POST)
- getShops (POST)
- getUserStatuses (POST)
- getUsers (POST)
- initStatuses (POST)

- sendMsg (POST)
- send_cloud_message (POST)
- sendSMS (POST)
- checkSMS (POST)
- EncodePhone (POST)
- DecodePhone (POST)
- set_call_lost (POST)
- get_makecall_data(POST)
- get_calls
- set_Webhook
- Loading Anrufhistorie

Von LiraX an das externe Softwareprodukt:

- staton (POST)
- smsDelivered (POST)
- smsReceived (POST)
- Kontakt (POST)
- event (POST)
- Aufzeichnung (POST)

*Alle Anfragen mit Content-Type: application/x-www-form-urlencoded

API Befehle und Examples von verfügbar Szenarien

Befehle vom externen Softwareprodukt an LiraX

makeCall

der Befehl ist erforderlich an initiate Anruf von der Manager an der Kunde. As ein Ergebnis von erfolgreich Befehl execution, LiraX wird erste make ein Anruf an der Manager Telefon, und then verbinden der Manager mit der Kunde.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall makeCall	string	
from	Mitarbeiter Nebenstelle Ausführen ausgehender Anruf	string	
to	Nummer an which der ausgehender Anruf ist made	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
idshop	Optional parameter, indicates id Shop, von which wird sein Anruf	integer	(Liste id kann empfangen function GetShops)

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=makeCall
von=101
an=19101234567
token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"id_makecall":"\$id_makecall"}	OK, \$id_makecall – ist verwendet wenn Empfang Webhook makecall_finished
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

killCall

der Befehl ist erforderlich an forcibly abschließen Anruf von der Manager an der Kunde.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall killCall	string	
Call_id	id Anruf, empfangen in event makecall_finished	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=killCall
Call_id=101234234234s3
token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

make2Calls

der Befehl ist erforderlich an verbinden zwischen jede other zwei Teilnehmer. As ein Ergebnis von erfolgreich Befehl execution, LiraX wird erste make ein Anruf an Telefon to1, und then verbinden it mit to2. von erforderlich für proper Anruf Routing.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall make2Calls	string	
from	Mitarbeiter Nebenstelle, von auf behalf von which sind made ausgehender Anrufs	string	
to1	Nummer, an which ist made erster Anruf	string	
to2	Nummer, an which ist made zweiter Anruf oder URL Audiodatei(mp3,wav) if nur ein Audiodatei needs an sein played	string	
speech	Text which muss sein spoken after answering der Anruf to1. Text muss consist von Sprache("uk ", "en ", "ru ") und itself text	string	Optional
atempo	Optional parameter, indicates speed playback von 0.5 until 2.0	string	Optional
timeout	Zeit in Minuten (if mehr als 10 every 10 Minuten wird sein retries until answering der Anruf both Teilnehmers). Standard 0 – nur one Versuch	string	Optional
successtime	Zeit in Sekunden – How many Sekunden, muss take place Gespräch, an consider, that Anruf erfolgreich. Standard 0	string	Optional
notbefore	Zeit in Stunden – Until what Stunde muss nicht Anruf. Standard 08	string	Optional
notafter	Zeit in Stunden – Nach what Stunde muss nicht Anruf, Standard 20	string	Optional
try	Nummer von Versuche Anruf Versuche an erste Seite mit ein Intervall 10 Minuten	integer	Optional =1
vtime	Zeit in Stunden und Minuten – hh:mm. Parameter kann sein verwendet für IVR	string	Optional
vdate	Datum – "dd.mm.YYYY", "today" oder "tomorrow". Parameter kann sein verwendet für IVR	string	Optional
vmoney	Money value von Typ float 0.00 mit zwei digits after der decimal points. Parameter kann sein verwendet für IVR	string	Optional
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
idshop	Optional parameter, indicates id Shop, von which wird sein Anruf	integer	(Liste id kann empfangen function GetShops)

FirstInternal	Optional parameter, indicates, that der erste leg ist ein internal Anruf if =1	integer	Optional
SecondInternal	Optional parameter, indicates, that der zweite leg ist ein internal Anruf if =1	integer	Optional
SpeechNoWait	Optional parameter, if =1, indicates, that speech ist played instead von ringback tones wenn Anruf Seite to2	integer	Optional

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

```
cmd=make2Calls
von=101
to1=19101234561
to2=19101234567
speech=en Hello Kunde
Timeout=60
successtime=30
notbefore=18
notafter=21
```

token=202cb962ac59075b964b07152d234b70

```
cmd=make2Calls
von=201
to1=19101234567
to2=https://www.test.mp3/tttt.mp3
Timeout=60
successtime=30
notbefore=18
notafter=21
try=4
token=202cb962ac59075b964b07152d234b70
```

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"id_make2calls": "\$id_make2calls"}	OK, \$id_make2calls – ist verwendet wenn Empfang Webhook make2calls_finished
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

AskQuestion

der Befehl ist erforderlich an ask der Teilnehmer ein question und empfangen ein Antwort. As ein Ergebnis von erfolgreich Befehl execution, LiraX wird erste make ein Anruf an Telefon to1, und then wird play der greeting «hello», if erforderlich possible additionally play until 4 Nachrichten «text1-4» selected Sprache (erforderlich if pauses sind needed). Then der question ist played, if aktiviert option Sprache Erkennung, wartet korrekt Wörter in Antwort «ok», if there sind no Erkennung, wartet für ein Taste Drücken. «von» erforderlich für proper Anruf Routing. der behavior ist similar an der make2calls Befehl

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall AskQuestion	string	
from	Mitarbeiter Nebenstelle, von auf behalf von which sind made ausgehender Anrufs	string	
to1	Nummer, an which ist made erster Anruf	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
hello	Optional parameter, sentence muss as greeting (Typ Hello)	string	
text1	Optional parameter, Nachricht 0	string	
text2	Optional parameter, Nachricht 0	string	
text3	Optional parameter, Nachricht 0	string	
text4	Optional parameter, Nachricht 0	string	
bye	Optional parameter, Nachricht Typ goodbye, works after der question	string	
ask	Question Nachricht	string	
ok	Space-separated Wörter, which we expect empfangen in positiv Antwort (Beispiel: yes OK)	string	
cburl	Optional parameter, URL für Webhook mit Nachricht make2calls_finished, in which wird sein übergeben Daten mit Antwort Teilnehmer	string	

Anfrage Beispiel:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=AskQuestion
```

```
von=101
```

```
cburl=https://mydomain.co/
```

```
to1=19101234561
```

```
token=202cb962ac59075b964b07152d234b70
```

```
hello=en Hello
```

```
text1=en ein neu service ist offered
```

ask=en If you want an empfangen it, say yes

ok=en yes want need

bye=en Goodbye

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"id_make2calls": "\$id_make2calls"}	OK, \$id_make2calls – ist verwendet wenn Empfang Webhook make2calls_finished
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

AddTag

der Befehl adds ein Tag an ein Kontakt

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall AddTag	string	
ani	Kontakt Telefonnummer	string	Optional
tag	tag	string	Optional
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=AddTag

ani=19101234567

Tag=OK

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

DelTag

der Befehl removes ein Tag von ein Kontakt

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall DelTag	string	
ani	Kontakt Telefonnummer	string	Optional
tag	tag	string	Optional
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=DelTag

ani=19101234567

Tag=OK

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

get_sip_route_in

der Befehl ist erforderlich an determine Liste SIP-Nummers für Nebenstelle Telefon

Anfrageparameter:

Name	Beschreibung	Typ/ Datenform at	Hinweis
cmd	Operation Typ, in diesem Fall get_sip_route_in	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
phone	Extension	string	Beispiel: 302

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=**get_sip_route_in**

Telefon=302

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	<pre>{ "N":2, "description":"route_in_sip.", "Ergebnisse":[{ "sip_name":"Test", "sip_number":"1115", "Benutzer_name":"User", "ext":"222", "Priorität":"0", "time_plan":"1" }, { "sip_name":"Test1", "sip_number":"1156", "Benutzer_name":"User1", "ext":"223", "Priorität":"0", "time_plan":"1" }] }</pre>	OK, if there sind no SIPs we get “N”:0, N-number SIPs, sip_name Name SIP, sip_number – SIP-Nummer, Benutzer_name – Name Benutzer, ext – Extension Benutzer, Priorität – current Priorität SIP, time_plan – id Wochenplan (business-hours=1, Abend=2, Nacht=3, Feiertag=4)
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben
402		Interner Fehler

set_sip_route_in

der Befehl ist erforderlich an setzen Priorität SIP-Nummers für Nebenstelle Telefon

Anfrageparameter:

Name	Beschreibung	Typ/ Datenform at	Hinweis
cmd	Operation Typ, in diesem Fall set_sip_route_in	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
phone	Extension	string	Beispiel: 302

sip_number	SIP-Nummer	string	Beispiel: 1115
time_plan	id Wochenplan (business-hours=1, Abend=2, Nacht=3, Feiertag=4)	string	Beispiel: 1
priority	Priorität SIP in Nebenstelle Nummer	string	Beispiel: 1

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=set_sip_route_in

Telefon=302

sip_number=1115

Priorität=0

time_plan=1

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"Ergebnis":"1"}	OK, Upon erfolgreich aktualisieren, Ergebnis=1, If there ist no SIP Ergebnis=0
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben
402		Interner Fehler

getUserSips

der Befehl ist erforderlich an determine there ist whether verfügbar Telefons für Nebenstelle ext.

Befehl caches Daten und updates them 1 Zeit in Stunde

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall getUserSips	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
ext	Extension	string	Beispiel: 302
web	If setzen > 0, then sind returned nur web Telefons	integer	Optional

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=getUserSips

ext=302

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"Benutzer_sips": [{"Telefon": "10000", "Benutzer_id": "1111", "ext": "302"}]}	OK, if there sind no verfügbar SIPs we get {"Benutzer_sips": []}
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

IsFreeUsers

der Befehl ist erforderlich an determine, kann Benutzer an Web Telefon accept Anruf.

Befehl prüft: - aktiviert whether Web Telefon; - no whether Gespräch mit Benutzer und gibt zurück erste Nummer von frei

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall IsFreeUsers	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
phones	Liste von Nebenstellen oder SIP Telefonnummern	string	Beispiel: 1212121, 1212123, 302
amo	If setzen in 1, then Nummern sind verwendet as amocrm Benutzer_id	string	Optional

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=IsFreeUsers

Phones=1212121,1212123,302

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"Ergebnis": "TelefonFromPhones", "SIP": "sip_number", "ext": "Nummer"}	OK, if alle Nummern besetzt we get {"Ergebnis": null}
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

IsCalling

Befehl prüft, ob ein Gespräch mit SIP-Nummer und gibt zurück erste Nummer ohne Gespräch (prüft Registrierung)

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall IsFreeUsers	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
phones	Liste von Nebenstellen oder SIP Telefonnummern	string	Beispiel: 1212121, 1212123

Anfrage Beispiel:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=IsCalling
```

```
Phones=1212121,1212123
```

```
token=202cb962ac59075b964b07152d234b70
```

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"Ergebnis":"TelefonFromPhones","ext":"Nummer"}	OK, if alle Nummern besetzt we get {"Ergebnis":null}
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

AddCampaign

der Befehl ist erforderlich an erstellen oder setzen ein Kampagne für Anruf Teilnehmers oder sending SMS.

wenn Anruf Teilnehmers erster Anruf goes an Benutzer, und der stream goes an der Teilnehmer

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall AddCampaign	string	
from	Mitarbeiter Nebenstelle, von auf behalf von which ist performed Kampagne	string	
ext	Mitarbeiter Nebenstelle, an which wird sein sein akzeptiert Anruf Mitarbeiters (Standard equal an von)	string	Optional
days	Liste days(0-Sunday,6-Saturday)	string	Standard: 1,2,3,4,5
time	Range Zeit	string	durch Standard:

			10:00-18:00
pdd	Zeit an post-processing(Sekunden)	integer	Standard: 5
preview_timeout	Verzögerung ausgehender Anruf, after answering der Anruf Mitarbeiter (Standard 0)	integer	Optional
try	Nummer von Versuche	integer	Standard: 1
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
type	Kampagne Typ 1 - Phone, 2 - SMS	integer	Standard: 1
phones	Liste von Telefonnummers(if Liste wird sein leer, Kampagne stops)	string	Beispiel: 12121234567, 12121234568
message	Nachricht text, für Typ=1 muss sein erste word Sprache {en,uk,ru,...} (in diesem Fall wird sein erster Anruf Teilnehmer und preview_timeout ist ignored)	string	Required für Typ=2
idshop	Optional parameter, indicates id Shop, von which wird sein Anruf	integer	Optional (Liste id kann empfangen function GetShops)

Hinweis: für Typ=1 Nummer simultaneous Anrufe ist determined in Einstellungen Kampagne in account Benutzer

für Vorschau mode der der Anruf erste goes an der Benutzer, und der stream goes an der Teilnehmer, wenn Nachricht != NULL, umgekehrt umgekehrt

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=AddCampaign

von=101

days=1,2

Zeit=10:00-11:00

Timeout=60

pdd=30

try=2

idShop=2

Typ=2

Nachricht=Hello

Phones=12121234567,12121234568,12121234569

token=202cb962ac59075b964b07152d234b70

oder

cmd=AddCampaign

von=101

days=1,2

Zeit=10:00-11:00

Timeout=60

pdd=30

try=2

idShop=2

Typ=1

Nachricht=en Hello

Phones=12121234567, 12121234568, 12121234569

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

AddPhoneCampaign

der Befehl ist erforderlich an hinzufügen Telefonnummers in Kampagne für Anruf Teilnehmers oder sending SMS.

wenn Anruf Teilnehmers der der Anruf erste goes an der Benutzer, und der stream goes an der Teilnehmer für Vorschau mode, wenn Nachricht != NULL, umgekehrt umgekehrt

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall AddPhoneCampaign	string	
from	Mitarbeiter Nebenstelle, von auf behalf von which ist performed Kampagne	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
phones	Liste von Telefonnummers	string	Beispiel: 12121234567, 12121234568

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=AddPhoneCampaign

von=101

Phones=12121234567, 12121234568, 12121234569

token=202cb962ac59075b964b07152d234b70

checkKontakt

der Befehl ist erforderlich an einfügen neu oder aktualisieren ein existing Kontakt, ist aktualisiert Name Kontakt und zuständig Mitarbeiter.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall checkKontakt	string	
ext	Mitarbeiter Nebenstelle zuständig für Kontakt	string	
ani	Primary Kontakt Telefonnummer	string	
Name	Name Kontakt	string	
add_phone	Liste additional Telefons	string	Optional
emails	Liste e-mail	string	Optional
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
trackid	Google Tracking id für Website	string	Optional
clientid	Google Kunde id	string	Optional
sitename	Name Website	string	Optional
source	Quelle	string	Optional
stype	Typ Quelle (Medium	string	Optional
campain	Kampagne	string	Optional
Term	Keyword word	string	Optional
tag	Tag (wenn presence ist inserted Tag)	string	Optional

Anfrage Beispiel:

```
POST https://lira.voip.com.ua:8482/general
cmd=checkKontakt
ext=101
ani=19101234567
Name=Dima
token=202cb962ac59075b964b07152d234b70
add_Telefon=380123456789,380234567890,380112345678
E-Mails=ein@ein.com,b@b.com
```

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

getKontakt

der Befehl ist erforderlich an empfangen information about existing Kontakt und it zuständig Mitarbeiter.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall getKontakt	string	
ani	Kontakt Telefonnummer	string	Optional
email	e-mail	string	Optional
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=getKontakt

ani=19101234567

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{ "Ergebnis": { "Name": "Dmitry", "mainTelefon": "12345678901", "zuständig": "209", "SIP": "111" } }	OK Name - Name Kontakt, mainTelefon – erste Kontakt Telefonnummer, zuständig - Extension zuständig, SIP - SIP-Nummer zuständig
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

getStatInfo

der Befehl ist erforderlich an empfangen Statistics Anrufe durch Mitarbeiters für Zeitraum nicht mehr als Monat. Befehl kann verwenden nicht mehr oft 1 Zeit an Stunde

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall getStatInfo	string	
Start	Datum in Format “YYYY-MM-DD HH:MM:SS” - Start Intervall	string	
Stop	Datum in Format “YYYY-MM-DD HH:MM:SS” - Ende Intervall	string	Optional, wenn nicht vorhanden, then für 24 Stunde
token	LiraX Schlüssel (token) setzen in der	string	

	Integrationseinstellungen		
--	---------------------------	--	--

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=getStatInfo

Start=2021:01:01 00:00:00

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	<pre>{ "Ergebnisse":[{ "description":"stat_info.", "stat":[{ "Login":"alex", "Name":"Alexey", "count_incoming":"0", "count_incoming_unique":"0", "count_incoming_answered":"0", "count_incoming_answered_unique":"0", "count_incoming_lost":"0", "count_incoming_lost_unique":"0", "count_outgoing":"10", "count_outgoing_unique":"4", "count_outgoing_answered":"5", "count_outgoing_answered_unique":"2", "count_outgoing_lost":"5", "count_outgoing_lost_unique":"4", "incoming_duration_talk":"0", "outgoing_duration_talk":"765" }] }] }</pre>	OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

createAufgabe

der Befehl ist erforderlich an erstellen neu Aufgabe für zuständig Mitarbeiter und Kontakt. If Kontakt nicht exists, ist erstellt neu

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall createAufgabe	string	
ext	Mitarbeiter Nebenstelle zuständig für Kontakt	string	
Abteilung	Name Department	string	Optional
ani	Nummer, Telefon Kontakt	string	Required ani oder E-Mail
email	e-mail Kontakt	string	Required ani oder E-Mail
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
webhook	URL für notification about Ergebnis	string	in Ende wird sein appended text
text	Text Aufgabes	string	
date	Datum execution Aufgabes	Datum	Optional, Standard current Zeit +1sec. Beispiel: "2017-01-01 17:00:00", muss sein mehr als current dates und weniger zwei year forward
type	Typ	integer	1,2,3 (Follow-up, Meeting, Anruf)

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=createAufgabe

ext=101

Abteilung=Sales

ani=19101234567

text=Prepare Aufgabe

Datum=2017-01-01 17:00:01

Typ=1

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"idAufgabe":"22"}	OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

AddAufgabeErgebnis

der Befehl ist erforderlich an hinzufügen Ergebnis an Aufgabe, abschließen Aufgabe oder ändern zuständig

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall AddAufgabeErgebnis	string	Required
ext	Mitarbeiter Nebenstelle who hinzugefügt Ergebnis	string	Required
idtask	Id Aufgabes	integer	Required
newext	neu zuständig	string	Optional
finish	Abschluss Aufgabes	bool	Optional
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
text	Text Ergebnis	string	Required

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=AddAufgabeErgebnis

ext=101

idAufgabe=123123

newext=191

text=Aufgabe at Stufe 1

Abschluss=0

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"idAufgabe":"123123"}	OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

GetAufgabe

der Befehl ist erforderlich an hinzufügen Ergebnis an Aufgabe, abschließen Aufgabe oder ändern zuständig

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall AddAufgabeErgebnis	string	Required

task_id	Id Aufgabes	integer	Required
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=GetAufgabe

Aufgabe_id=123123

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	<pre>{ "Aufgabe_id":"609", "Typ":"Follow-up", "date_insert":"2014-12-09 14:15:55", "date_end":"2014-12-09 15:12:07", "text":"test Aufgabe!", "ext":"101", "Name":"System", "coauthors":"1", "Liste":[{ "ext":"504", "Name":"Sergey Rybalkchenko" }] }</pre>	Here Typ – Typ Aufgabes; date_insert – Datum creation; date_end – Datum execution; text – text Aufgabes; ext – zuständig; Name - Name zuständig; coauthors - Nummer co-assignees; Liste – Liste co-assignees
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

createHinweis

der Befehl ist erforderlich an erstellen Notiz für existing Kontakt von zuständig Mitarbeiter.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall createHinweis	string	
ext	Mitarbeiter Nebenstelle zuständig für Kontakt	string	
ani	Nummer, Telefon Kontakt	string	

token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
text	Text Notiz	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=createHinweis

ext=101

ani=19101234567

text=News

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

createGeschäft

der Befehl ist erforderlich an erstellen neu Geschäft für zuständig Mitarbeiter und existing Kontakt.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall createGeschäft	string	
ext	Mitarbeiter Nebenstelle zuständig für Kontakt	string	
ani	Nummer, Telefon Kontakt	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
Name	Name Geschäft	string	
sum	Amount Geschäft	integer	
Stufe	Stufe Geschäft	integer	Allowed nur existing Stufen
Status	Status Geschäft (Optional)	integer	0-active, 1- erfolgreich Completed, 2- Completed Unsuccessfully

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

```
cmd=createGeschäft
ext=101
ani=19101234567
Name=Prepare Aufgabe
sum=3030
Stufe=1
token=202cb962ac59075b964b07152d234b70
```

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"id_Geschäft":\$integer}	OK (\$integer - integer Nummer)
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

updateGeschäft

der Befehl ist erforderlich an aktualisieren existing Geschäft mit id_Geschäft für zuständig Mitarbeiter und existing Kontakt.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall updateGeschäft	string	
ext	Mitarbeiter Nebenstelle zuständig für Kontakt	string	
ani	Nummer, Telefon Kontakt	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
id_Geschäft	Id Geschäft	integer	
Name	Name Geschäft	string	
sum	Amount Geschäft	integer	
Stufe	Stufe Geschäft	integer	Allowed nur existing Stufen
Status	Status Geschäft	integer	0-active, 1- erfolgreich Completed, 2- Completed Unsuccessfully

Anfrage Beispiel:

```
POST https://lira.voip.com.ua:8482/general
cmd=updateGeschäft
ext=101
ani=19101234567
id_Geschäft=1910
```

Name=Prepare Aufgabe

sum=3030

Stufe=2

Status=0

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben, oder Parameter nicht aktualisiert
401		Ungültig Schlüssel (token) war übergeben

addBlackPhone

der Befehl ist erforderlich an hinzufügen in schwarz Liste Nummer Telefon (nur digits).

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall addBlackPhone	string	
phone	Nummer Telefon	string	nur digits
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=addBlackPhone

Telefon=100111000101

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

listBlackPhone

der Befehl ist erforderlich an empfangen schwarz Liste Telefons.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall listBlackPhone	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=listBlackPhone

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"Sperrliste": ["Telefon1","Telefon2",,...,"TelefonN"]}	OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

delBlackPhone

der Befehl ist erforderlich an löschen in schwarz Liste Nummer Telefon (nur digits).

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall delBlackPhone	string	
phone	Nummer Telefon	string	nur digits
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=delBlackPhone

Telefon=100111000101

`token=202cb962ac59075b964b07152d234b70`

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

listBlackIP

der Befehl ist erforderlich an empfangen schwarz Liste IP addresses

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall listBlackIP	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST `https://lira.voip.com.ua:8482/general`

`cmd=listBlackIP`

`token=202cb962ac59075b964b07152d234b70`

Antwort options:

HTTP-Code	Body	Beschreibung
200	<code>{"Sperrliste":["IP1","IP2",..., "IPN"]}</code>	OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

addBlackIP

der Befehl ist erforderlich an hinzufügen in schwarz Liste IP Adresse.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
------	--------------	-------------------------	---------

cmd	Operation Typ, in diesem Fall addBlackIP	string	
IP	IP Adresse	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

```
POST https://lira.voip.com.ua:8482/general

cmd=addBlackPhone
IP=100.111.000.101
token=202cb962ac59075b964b07152d234b70
```

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

delBlackIP

der Befehl ist erforderlich an löschen in schwarz Liste IP Adresse.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall delBlackIP	string	
IP	IP Adresse	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

```
POST https://lira.voip.com.ua:8482/general

cmd=delBlackPhone
IP=100.111.000.101
token=202cb962ac59075b964b07152d234b70
```

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK

400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

getStages

der Befehl ist erforderlich an empfangen Liste Stufen Geschäfts, sorted According an Pipeline.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall getStages	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=getStages

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"Stufen":[{"Stufe":"\$Stufe","title":"..."},...]}	OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

getShops

der Befehl ist erforderlich an empfangen Liste Shops.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall getShops	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=getShops

`token=202cb962ac59075b964b07152d234b70`

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"Shops":[{"id":\$id_Shop,"Name":"..."},...]}	OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

getUserStatuses

der Befehl ist erforderlich an empfangen Liste Statuswerte Statuswerte Benutzer.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall getUserStatuses	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST `https://lira.voip.com.ua:8482/general`

`cmd=getUserStatuses`

`token=202cb962ac59075b964b07152d234b70`

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"Statuswerte":[{"Status":\$Status,"title":"..."},...]}	OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

getUsers

der Befehl ist erforderlich an empfangen Liste Benutzers

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall getUsers	string	

token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
-------	---	--------	--

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=getUsers

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"Benutzers":[{"id":\$id,"Name":"...", "ext":"...", "active":"0 1"},...]}	OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

initStatuses

der Befehl ist erforderlich an empfangen current Statuswerte operators, through Webhooks. sind verwendet wenn enabling integration. Further Status wird sein arrive durch them ändern, previous Statuswerte muss ignore

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall initStatuses	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=initStatuses

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

sendMsg

der Befehl ist erforderlich an senden Nachricht Mitarbeiter in corporate messenger.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall sendMsg	string	
ext	Mitarbeiter Nebenstelle, an whom ist gesendet Nachricht	string	
ani	Name Kontakt, which gesendet Nachricht	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
text	Nachricht text	string	

Anfrage Beispiel:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=sendMsg
```

```
ext=101
```

```
ani=19101234567
```

```
text=Prepare Aufgabe
```

```
token=202cb962ac59075b964b07152d234b70
```

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben, oder Parameter nicht aktualisiert
401		Ungültig Schlüssel (token) war übergeben

send_cloud_message

der Befehl ist erforderlich an senden Nachricht in cloud messenger (Telegram, Viber, Facebook).

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall send_cloud_message	string	
Kunde	Kunde ID cloud messenger	string	Required wenn absence ani
ani	Nummer Telefon Kunde(ist verwendet nur Telegram)		Required wenn

			absence Kunde
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
text	Nachricht text	string	

Anfrage Beispiel:

```
POST https://lira.voip.com.ua:8482/general
cmd=send_cloud_message
Kunde=web_t123123123_00000
text=Prepare Aufgabe
token=202cb962ac59075b964b07152d234b70
```

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben, oder Parameter nicht aktualisiert
401		Ungültig Schlüssel (token) war übergeben
402		Nachricht nicht übergeben

sendSMS

der Befehl ist erforderlich an senden SMS through GSM modem. Anfragen need senden nicht mehr often 1 Zeit in 5 Sekunden an one modem.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall sendSMS	string	
ext	Mitarbeiter Nebenstelle, who sendet SMS	string	nur digits
provider	Identifier provider (Nummer Telefon in GSM modem)	string	nur digits
phone	Nummer Phone, an whom senden Nachricht	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	
text	Nachricht text	string	

Anfrage Beispiel:

```
POST https://lira.voip.com.ua:8482/general
cmd=sendSMS
ext=101
Telefon=19101234567
provider=380123456789
```

text=Prepare Aufgabe

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"id_sms":\$id_sms}	OK
400		Ungültig Parameter waren übergeben, oder Parameter nicht aktualisiert
401		Ungültig Schlüssel (token) war übergeben
402		Modem nicht connected
403		Modem besetzt, repeat later

checkSMS

der Befehl ist erforderlich an check Status delivery SMS through GSM modem. Anfragen need senden nicht mehr often 1 Zeit in 5 Sekunden an one modem.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall checkSMS	string	
ext	Mitarbeiter Nebenstelle, who sendet SMS	string	nur digits
provider	Identifier provider (Nummer Telefon in GSM modem)	string	nur digits
id_sms	Id empfangen Befehl sendSMS	integer	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=checkSMS

ext=101

id_sms=1910123

provider=380123456789

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"sms_Status":\$id_Status}	OK, 1- übergeben provider, Status unknown; 2- übergeben provider, goes

		delivery; 3- delivered; 4- Zeit delivery expired
400		Ungültig Parameter waren übergeben, oder Parameter nicht aktualisiert
401		Ungültig Schlüssel (token) war übergeben

EncodePhone

der Befehl ist erforderlich an check empfangen encrypted Nummer.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall EncodePhone	string	
phone	Nummer Telefon	string	nur digits
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=EncodePhone

Telefon=12123123123

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"Telefon":\$encoded}	OK
400		Ungültig Parameter waren übergeben, oder Parameter nicht aktualisiert
401		Ungültig Schlüssel (token) war übergeben

DecodePhone

der Befehl ist erforderlich an empfangen decrypted Nummer.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall EncodePhone	string	
phone	Nummer Telefon	string	nur digits
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=DecodePhone

Telefon=12123120960170061170799

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{"Telefon":\$decoded}	OK
400		Ungültig Parameter waren übergeben, oder Parameter nicht aktualisiert
401		Ungültig Schlüssel (token) war übergeben

set_call_lost

der Befehl ist erforderlich an setzen Nummer Telefon, as missed Anruf, für CallBack

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall set_call_lost	string	erforderlich
phone	Nummer Telefon	string	nur digits, erforderlich
ext	Nummer Telefon Mitarbeiters für CallBack	string	nur digits, erforderlich
time_plan	1 - Working Zeit, 2 - Evening, 3 - Night, 4 - Days off	digits	Zeit Intervall ist determined in Wochenplan, Standard 1
max_try	Maximum Nummer von Versuche connection mit Teilnehmer (mit Operator nicht ist considered)	digits	Standard 3
Intervall	Initial Intervall zwischen Versuche in Minuten, multiple 6. Subsequent Intervall equal an initial multiply an Nummer von Versuche	digits	Standard 6
Minuten	Nummer Minuten Verzögerung	digits	Standard 0
Stunden	Nummer Stunden Verzögerung	digits	Standard 0
days	Nummer days Verzögerung	digits	Standard 0
weeks	Nummer weeks Verzögerung	digits	Standard 0
months	Nummer months Verzögerung	digits	Standard 0
info	Text, which wird sein sein spoken Mitarbeiter. Format: Sprache und text. Beispiel: "ru Anruf Back service an missed Anruf, wait connection mit Teilnehmer"	string	
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	erforderlich

Anfrage Beispiel:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=set_call_lost
```

```
Telefon=12123120960
```

```
ext=222
```

```
token=202cb962ac59075b964b07152d234b70
```

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK

400		Ungültig Parameter waren übergeben, oder Parameter nicht aktualisiert
401		Ungültig Schlüssel (token) war übergeben
402		Problems mit server

get_makecall_data

Befehl ermöglicht empfangen information about Anruf unter Verwendung von id_makecall, empfangen Befehl makecall

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall get_makecall_data	string	erforderlich
id_makecall	empfangen in Ergebnis makecall	string	nur digits, erforderlich
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	erforderlich

Anfrage Beispiel:

POST https://lira.voip.com.ua:8482/general

cmd=get_makecall_data

id_makecall=e5251de22f99e071

token=202cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200	{ "id_makecall":"e5251de22f99e071", "call_start":"2021-10-28 16:02:32", "call_record":"https://Dateien.lirax.net/voicerec.w av", "Dauer":"5" }	OK
400		Ungültig Parameter waren übergeben, oder Parameter nicht aktualisiert
401		Ungültig Schlüssel (token) war übergeben
402		Problems mit server

get_calls

Befehl ermöglicht log Anrufe in json Format (maximum 5000) für range Zeit. für Befehle possible verwenden method GET

Anfrageparameter:

Name	Beschreibung	Typ/ Datenforma t	Hinweis
cmd	Operation Typ, in diesem Fall get_calls	string	erforderlich
date_start	Start range	string	in SQL Format, erforderlich
date_finish	Ende range	string	in SQL Format, erforderlich Intervall muss sein weniger 48hours
call_type	Typ Anruf, {-1,0,1}, entsprechend {Nebenstelle, eingehend, ausgehend }, if ist absent, then alle	numerical	optional
ani	Who angerufen	string	optional
dnis	Where angerufen	string	optional
offset	Skip in output Nummer rows	numerical	optional
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	erforderlich

Anfrage Beispiel:

```
GET https://api.lirax.net/general?cmd=get_calls&date_start=2023-10-04 10:00:00&date_finish=2023-10-04 11:00:00&token=-----
```

Antwort options:

HTTP-Code	Body	Beschreibung
200	<pre>[{"call_id":"6708","ani":"203" , "dnis":"380123123", "Start":"2023-10-04 10:21:14", "verbinden":"2023-10-04 10:21:25", "Aufzeichnung": "" , "Typ":"1","Dauer": "163","disconnect_side":"0", "call_call_center_id":"6" }, {"call_id":"8326","ani":"380111111" ,"dnis":"380 123123", "Start":"2023-10-04 10:04:07", "verbinden":"","Aufzeichnung":"","Typ":"1","Dau er":"0", "disconnect_side":"- 1","call_call_center_id":"0" } https://files.lirax.net/voice_record2/x_10_21_14. wav</pre>	OK verbinden – if nicht beantwortet wird sein leer row und Aufzeichnung – URL recording Gespräch disconnect_side – who hung up handset, -1 system(possible durch Timeout), 0 – who angerufen, 1 – an whom angerufen call_call_center_id – what Anruf center processed Anruf (if there ist)
400		Ungültig Parameter waren übergeben, oder Parameter nicht

		aktualisiert
401		Ungültig Schlüssel (token) war übergeben
402		Problems mit server

set_Webhook

Befehl ermöglicht log Anrufe in json Format (maximum 5000) für range Zeit. für Befehle possible verwenden method GET

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall set_Webhook	string	erforderlich
type	«0» — eingehend, «1» — ausgehend	numerical	erforderlich
event_type	init - wenn Beginn Anruf, trying - wenn Beginn ausgehend leg, verbinden - wenn answering handset Teilnehmer, release - wenn Ende von ausgehend leg, disconnect - wenn Ende von Anruf	string	erforderlich
json	Template json, which wird sein verwendet in Body Webhook. Here kann verwenden macros [ani] — who Anrufe, [dnis] — where, [Datum] — current Zeit, [starto] — Start eingehend leg, [starta] — Start answering der Anruf Teilnehmer, [durationo] — Dauer eingehend leg, [durationa] — Dauer ausgehend leg, [anidnis] — Nummer Operator für eingehender Anruf, [Typ] — Typ Anruf «-1» — Nebenstelle, «0» — eingehend, «1» — ausgehend Beispiel: { "call_type": "[Typ]", "event_date": "[Datum]", "number_a": "[ani]", "number_b": "[dnis]", "call_duration": "[durationa]" }	string	kann sein absent, if method POST, then Body wird sein verwendet as template
header_field	Additional Feld header, which LiraX wird sein einfügen in Webhook Beispiel: Authorisation: 123123123	string	optional
method	GET POST	string	Optional Standard GET
URL	URL für Webhook Beispiel: https://test.test.test.com/Webhook?ein=1&b=2	string	erforderlich
token	LiraX Schlüssel (token) setzen in der Integrationseinstellungen	string	erforderlich

Anfrage Beispiel:

```
GET 0
&json={"call_type":"[Typ]","event_date":"[Datum]","number_a":"[ani]","number_b":"[dnis]","call_duration":"[durationa]"}
&token=-----https://api.lirax.net/general?cmd=set\_webhook&type=&event\_type=disconnect&url=https://test.test.test.com/webhook?
a=1&b=2
```

Antwort options:

HTTP-Code	Body	Beschreibung
200		erfolgreich
400		Ungültig Parameter waren übergeben, oder Parameter nicht aktualisiert
401		Ungültig Schlüssel (token) war übergeben
402		Problems mit server

Loading Anrufhistorie

Autorisierung through API

Performs Autorisierung Benutzer in system. alle methods API kann sein verwendet nur after Autorisierung.

in Antwort an Anfrage, wenn erfolgreich Autorisierung, except Antwort Body ist returned cookie Datei, containing Schlüssel session, similarly work mit WEB-browser. wenn further Anfragen an API-methods need back übergeben empfangen cookie. Zeit lifetime session - 15 Minuten.

alle Anfragen an API occur von under Benutzer, attributes which waren verwendet wenn Autorisierung through this method. wenn this we consider alle rights Benutzer, i.e. through API muss nicht empfangen mehr als Daten, als itself Benutzer kann view through interfaces Systeme. We recommend für API erstellen separate Benutzer für mehr specific Einstellungen rights connected application.

ReQuelle URL

POST

<https://lira.lirax.net/api/private/api/auth.php>

Parameter

PARAMETER	DESCRIPTION
USER_LOGIN	require Login Benutzer.
USER_HASH	require Password Benutzer.

Anrufe

https://lira.lirax.net/services/call_show.php?action=show&date_start=2016-12-06

00:00:00&date_finish=2016-12-13

23:59:59&filter=call_ani_dnis&filter_value=&call_type=&call_connect=&call_export=1

date_start

Datum von

date_finish

Datum Until

filter

Filter durch Feld

possible values:

"call_ani", "call_dnis", "call_ani_dnis", "call_originip", "call_destination", "call_origination",
"call_utm_Quelle", "call_utm_medium", "call_utm_Kampagne", "call_utm_term"

filter_value

Values filter

call_type

possible values:

0 eingehend, 1 ausgehend

call_connect

possible values:

0 nicht war Gespräch, 1 war Gespräch

Befehle von LiraX an der externes Softwareprodukt

makecall_finished

Befehl notifies about Abschluss Aufgabe an creation Anruf für 2 Teilnehmers.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall makecall_finished	string	
id_makecall	Id - makecall	string	
Call_id	Call_id ausgehend leg Anruf	string	wenn success=0 leer
success	beantwortet handset Teilnehmer	integer	0 1
from_LiraX_token	Schlüssel (token) externes Softwareprodukt, angegeben in Integrationseinstellungen	string	

Anfrage Beispiel:

```
POST https://domain/LiraX
cmd=makecall_finished
id_makecall=ef232234feb12
from_LiraX_token=303cb962ac59075b964b07152d234b70
Call_id=b964b07152d
success=1
```

Antwort options:

HTTP- Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

make2calls_finished

Befehl notifies about Abschluss Aufgabe an creation Anruf für 2 Teilnehmers.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall make2calls_finished	string	
id_make2calls	Id - make2calls	string	
success	war whether answering handset	integer	0- war nicht, 1-

			beantwortet handset
duration_success	Dauer Gespräch war nicht weniger, als expected in parameter successtime	integer	0- weniger, 1- mehr als equal an
keys	Json array gedrückt buttons, ivr_name – Name IVR, ivr_entry – item IVR menu, Schlüssel – selected Taste	Json	kann sein absent
from_LiraX_token	Schlüssel (token) externes Softwareprodukt, angegeben in Integrationseinstellungen	string	

Anfrage Beispiel:

```
POST https://domain/LiraX
cmd=make2calls_finished
id_make2calls=ef232234feb12
from_LiraX_token=303cb962ac59075b964b07152d234b70
success=1
keys=[{"ivr_name":"Test","ivr_entry":"0","Schlüssel":"9"},{"ivr_name":"Test","ivr_entry":"09","Schlüssel":"7"}]
duration_success=0
```

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

staton

Befehl notifies about about Mitarbeiter Statuswerte.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall staton	string	
ext	Extension Mitarbeiter LiraX	string	
Status	identifiziert Status Mitarbeiter in personal account	integer	0- disconnected, 1-frei, >1 -besetzt, value according an Einstellungen
from_LiraX_token	Schlüssel (token) externes Softwareprodukt, angegeben in Integrationseinstellungen	string	

Anfrage Beispiel:

```
POST https://domain/LiraX
```

```
cmd=staton
ext=910
from_LiraX_token=303cb962ac59075b964b07152d234b70
Status=1
```

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

smsDelivered

Befehl notifies about about delivery SMS.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall smsDelivered	string	
id_sms	Id SMS, empfangen Befehl sendSMS	integer	
from_LiraX_token	Schlüssel (token) externes Softwareprodukt, angegeben in Integrationseinstellungen	string	

Anfrage Beispiel:

```
POST https://domain/LiraX
cmd=smsDelivered
from_LiraX_token=303cb962ac59075b964b07152d234b70
id_sms=1212
```

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

smsReceived

Befehl notifies about about Empfang SMS.

Anfrageparameter:

Name	Beschreibung	Typ/	Hinweis
------	--------------	------	---------

		Datenformat	
cmd	Operation Typ, in diesem Fall smsReceived	string	
ani	Nummer Telefon Teilnehmer	string	
id	Id SMS, empfangen von provider	integer	
provider	Name provider (Nummer Telefon)	string	
text	Nachricht text	string	in UTF8
from_LiraX_token	Schlüssel (token) externes Softwareprodukt, angegeben in Integrationseinstellungen	string	

Anfrage Beispiel:

```
POST https://domain/LiraX

cmd=smsReceived
from_LiraX_token=303cb962ac59075b964b07152d234b70
id_sms=121
ani=38012345679
provider=380112345678
text=Hello
```

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400		Ungültig Parameter waren übergeben
401		Ungültig Schlüssel (token) war übergeben

Kontakt

Befehl für Empfang information about der Name von Kunde und it zuständiger Mitarbeiter durch its Telefonnummer.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenformat	Hinweis
cmd	Operation Typ, in diesem Fall Kontakt	string	
phone	Nummer Kontakt	string	
callid	eindeutig id Anruf	string	
diversion	external Nummer Telefon LiraX through which arrived Anruf	string	wenn ausgehend equal an Nummer Teilnehmer
from_LiraX_token	Schlüssel (token) externes Softwareprodukt, angegeben in Integrationseinstellungen	string	

Anfrage Beispiel:

```
POST https://domain/LiraX
```

cmd=Kontakt
Telefon=19101234567
diversion=12101234567
from_LiraX_token=303cb962ac59075b964b07152d234b70
callid=D12D0EB124F4E64AF4EA-1511

Antwort options:

HTTP-Code	Body	Beschreibung
200	{ "Kontakt_name": "Ivanov Ivan Ivanovich", "zuständig": 103 }	zuständig - Kurzwahl zuständig Benutzer, kann sein absent, if such no oder at it nicht setzen Kurzwahls
400	{ "Fehler": "Ungültig Parameter" }	Ungültig Parameter waren übergeben
401	{ "Fehler": "Ungültig token" }	Ungültig Schlüssel (token) war übergeben

event

LiraX sendet in externes Softwareprodukt Benachrichtigungen about events Anrufe: appearance, acceptance oder Abschluss Anruf, und also information about Anruf wenn it Abschluss.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenfor mat	Hinweis
cmd	Operation Typ, in diesem Fall event	string	
type	Typ Anruf	in/out	eingehend / ausgehend
event	Typ events, related mit Anruf: eingehend - arrived eingehender Anruf (in this Zeit der Manager Telefon sollte Start ringing). akzeptiert - Anruf erfolgreich akzeptiert (Manager hung up handset). COMPLETED - Anruf completed (Manager oder Kunde hung up handset after Gespräch). CALL_COMPLETED – Anruf completely completed.	string	
phone	Nummer Telefon Kunde	string	
diversion	external Nummer Telefon LiraX through which arrived oder disconnected Anruf	string	
ext	Extension Mitarbeiter LiraX, in Fall CALL_COMPLETED und Typ=in, contains Nummer, an which arrived Anruf	string	
callid	eindeutig id Anruf	string	
Dauer	Dauer Gespräch in Sekunden	int	wenn event=COMPLETED, CALL_COMPLETED
call_duration	Dauer Anruf an moment events in Sekunden	int	wenn event=COMPLETED, CALL_COMPLETED
is_recorded	aufgezeichnet whether Anruf	0/1	wenn event=COMPLETED, CALL_COMPLETED
Status	Status eingehender Anruf: Success - erfolgreich eingehender Anruf Missed – missed eingehender Anruf Status ausgehender Anruf: Success - erfolgreich ausgehender Anruf besetzt - LiraX empfangen Antwort besetzt NotAvailable - LiraX empfangen Antwort Subscriber unavailable NotAllowed - Anrufe an this direction forbidden	string	wenn event=COMPLETED, CALL_COMPLETED
record_link	Anrufaufzeichnung link, if LiraX kann	string	wenn event=COMPLETED,

	provide it sofort, if there sind no - link ist gesendet Befehl Aufzeichnung		CALL_COMPLETED
tid	Tracking Id		
cid	Kunde Id		
cs	Quelle		
cm	Medium		
cc	Kampagne		
ct	Term		
keys	Last selection IVR menu	json	wenn event= CALL_COMPLETED
From_LiraX _token	Schlüssel (token) externes Softwareprodukt, angegeben in Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://domain/LiraX

cmd=event

Typ=out

event=COMPLETED

Telefon=19101234567

ext=102

callid=D10D0EB124F4E64AF4EA-1511

Dauer=124

is_recorded=1

Status=Success

record_link=https://link/file.wav

keys=[{"ivr_name":"Test","ivr_entry":"09","Schlüssel":"4"}]

from_LiraX_token=303cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400	{ "Fehler": "Ungültig Parameter" }	Ungültig Parameter waren übergeben
401	{ "Fehler": "Ungültig token" }	Ungültig Schlüssel (token) war übergeben

Aufzeichnung

Notification contains information about recording Gespräch.

Anfrageparameter:

Name	Beschreibung	Typ/ Datenfor mat	Hinweis
cmd	Operation Typ, in diesem Fall Aufzeichnung	string	
callid	eindeutig id Anruf	string	
record_link	Anrufaufzeichnung link	string	
from_LiraX_token	Schlüssel (token) externes Softwareprodukt, angegeben in Integrationseinstellungen	string	

Anfrage Beispiel:

POST https://domain/LiraX

cmd=Aufzeichnung

callid=D10D0EB124F4E64AF4EA01511

record_link=https://link/Datei.wav

from_LiraX_token=303cb962ac59075b964b07152d234b70

Antwort options:

HTTP-Code	Body	Beschreibung
200		OK
400	{ "Fehler": "Ungültig Parameter" }	Ungültig Parameter waren übergeben
401	{ "Fehler": "Ungültig token" }	Ungültig Schlüssel (token) war übergeben

ChangeLog:

2021-01-28

make2calls- hinzugefügt Feld FirstInternal

2021-02-18

getKontakt- hinzugefügt function

2021-02-19

getStatInfo- hinzugefügt function

2021-04-12

AskQuestion- hinzugefügt function

2021-04-13

AddTag- hinzugefügt function

2021-04-13

DelTag- hinzugefügt function

2021-04-14

checkKontakt — hinzugefügt Feld Tag

2021-08-13

get_sip_route_in — hinzugefügt function

2021-08-13

set_sip_route_in — hinzugefügt function

2021-09-12

SecondInternal in function make2calls

2021-09-12

SpeechNoWait in function make2calls

2021-10-26

set_call_lost- hinzugefügt function

2021-10-28

get_makecall_data - hinzugefügt function

2021-11-19

set_call_lost - fields hinzugefügt info, Minuten, Stunden, days, weeks, months

2021-12-30

createAufgabe — hinzugefügt parameter E-Mail, if Kontakt nicht exists, ist erstellt neu

2022-08-25

Event — hinzugefügt parameter call_duration

2022-11-10

send_cloud_message - hinzugefügt function

2023-10-04

get_calls- hinzugefügt function

2023-10-20

AddAufgabeErgebnis- hinzugefügt function

2024-09-24

GetAufgabe- hinzugefügt function

--