Avhan Technologies Pvt. Ltd.

Jodo API - Customer Functionality documentation

AVHAN TECHNOLOGIES PVT LTD   www.avhan.com

Document History

Date	Document Version	Prepared By	Verified By	Approved By
05-September-2025	1.0.6	Abhishek	Satish Mokashi	

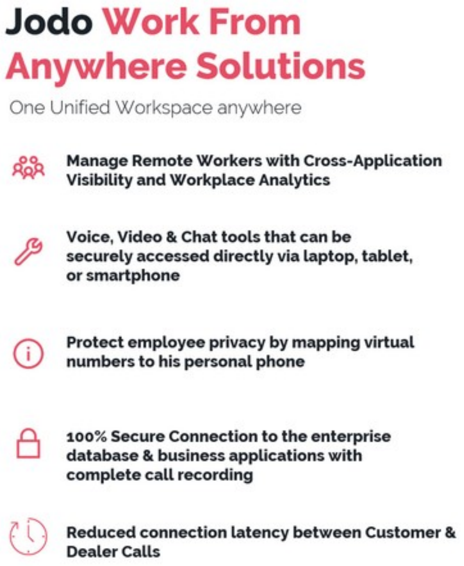
Jodo Introduction

Jodo World (“Jodo”) is a Cloud Workplace Technology Framework designed to handle all present and future business communication media and channels by directly embedding communications within software applications and platforms (browser-driven or mobile apps).

Jodo handles communication traffic distribution globally, via multi-lingual automation tools and monitors traffic handling via dispersed, multi-device global teams.

Jodo handles business phone calls from multiple geographies (distributed architecture), social media channels, email, chat, digital voice and video channels and SMS.

Jodo handles all communication media on the Cloud and communication media events, actions and controls are offered within any business applications via Jodo API.



Jodo Work From Anywhere Solutions

One Unified Workspace anywhere

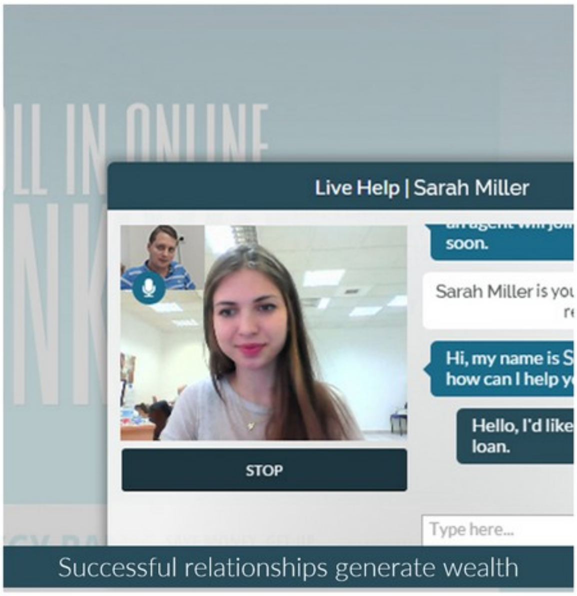
Manage Remote Workers with Cross-Application Visibility and Workplace Analytics

Voice, Video & Chat tools that can be securely accessed directly via laptop, tablet, or smartphone

Protect employee privacy by mapping virtual numbers to his personal phone

100% Secure Connection to the enterprise database & business applications with complete call recording

Reduced connection latency between Customer & Dealer Calls



soon.

Sarah Miller is you

Hi, my name is S
how can I help y

Hello, I'd like
loan.

Successful relationships generate wealth

JODO WORLD

Jodo Video, Voice, & Chat Support Solutions

Create Uniform Service Delivery – Globally

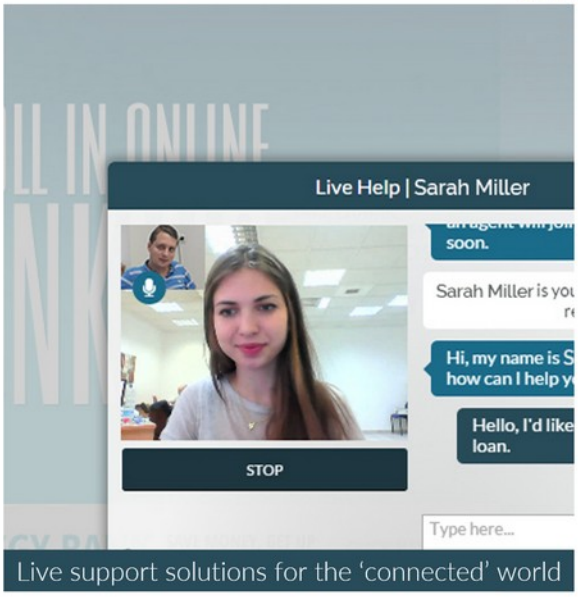
- **Create support processes in the 'connected' universe – Globally.**
- **Provide support using relevant or multiple media channels – Voice, Video & Chat - directly from your website or helpdesk application.**
- **Enhanced routing options based on employee skill ratings, Time zone, Location, Customer history, ID etc.**
- **Enhance the power of your Support quality using ScreenShare, DocumentShare & GPSShare**
- **Monitor your global support operations and resolution status in real-time, via visual dashboards, reports, KPI's, and more.**
- **Jodo Support solution is accessible from any 'connected' device on all popular browsers, supports multi lingual delivery, expanding your access globally.**

JODO WORLD

Jodo Live Support Solutions

Create digital Live Support Workspaces – its easy

- **Maximize customer and employee satisfaction.**
- **Connect customers to skilled business resources from across the world.**
- **Lower your support costs.**
- **Connect Customers to dispersed and diverse Team, of employees, outsourced agencies and specialists, in a #WorkFromAnywhere mode**
- **Monitor your dispersed and diverse Teams in a #WorkFromAnywhere mode**



soon.

Sarah Miller is you

Hi, my name is S
how can I help y

Hello, I'd like
loan.

Live support solutions for the 'connected' world

Partial list of Jodo Use Cases possible via Jodo API:

Jodo Customer Web Service

Import Process Data (Push API)

Purpose: The Push API in the Jodo Customer Web Service allows for the seamless transfer of customer data to the Jodo application. This functionality is particularly useful when initiating calls or handling incoming calls from CRM systems or other business applications, enabling integration with external platforms such as marketing automation tools. The primary objective is to enable Jodo to process customer data and trigger actions such as initiating calls, sending automated messages, and enhancing the customer experience through integration with third-party services (such as TTS engines).

Use Cases:

1. Initiate and Handle Customer Call from CRM / Business Application Screen:

This use case is designed for scenarios where an agent or system needs to initiate a call from a CRM or business application. The **Push API** is invoked when the user clicks on a specific contact entry or triggers the call in the background, prompting Jodo to initiate the call.

- **Technical Flow**:
  - CRM or business application sends customer data (such as the contact number, first name, last name, etc.) to Jodo via the Push API.
  - The contact number is validated for correctness and format (e.g., country code, length).
  - If the number is valid, the system proceeds with placing a call to the contact using the **Jodo Platform**.
  - This function is often employed in **Click-to-Call** campaigns, where an agent or system can initiate calls across multiple media channels (e.g., voice, SMS, email, etc.).
- **Key Steps**:
  - Send contact data (name, phone number, etc.) via API.
  - Validate the phone number.
  - Initiate call to contact from Jodo platform.

2. Incoming Calls Screen-Pop as Response to Multi-Media Campaigns:

In this scenario, external platforms, typically marketing automation tools, can use the Push API to pre-send customer data to Jodo. This ensures that when an incoming call is triggered as a part of a multi-channel marketing campaign (e.g., SMS or email), Jodo will present a **screen pop-up** to the agent or system with relevant customer information, allowing for quick and personalized responses.

- **Technical Flow**:
  - External platform (e.g., marketing automation system) triggers the **Push API** to send customer data into Jodo, typically before the scheduled campaign time.
  - The customer data includes important identifiers, such as the customer's name, contact number, campaign details, etc.
  - Jodo's **Call Guide** receives this data and presents it as a **screen-pop** to the agent when the call comes in, allowing the agent to act on the data with minimal delay.
- **Key Steps**:
  - External platform sends customer data via Push API before scheduled campaign time.
  - Data is received by Jodo Call Guide, which prepares for the incoming call.
  - When the call arrives, the agent sees a screen-pop with customer information.

3. Make Multi-lingual Calls Using Automated Voice (Engineered Voice) via Text-to-Speech Engine:

This use case is designed for **automated voice dialing** where the customer's details, including their name, are used to generate a personalized voice message via a Text-to-Speech (TTS) engine. Jodo integrates with cloud-based TTS services like Google or Amazon TTS to provide **multi-lingual support** for customer interactions, delivering dynamic and localized messages to the customer.

- **Technical Flow**:
  - Jodo receives the customer's first name and last name via the Push API in real-time.
  - The customer's data is passed to a TTS engine (which can be configured by the user to use services like **Google TTS** or **Amazon Polly**).
  - The TTS engine generates an audio file based on the customer's name and other dynamic data.

    - Once the audio response is generated, Jodo uses this response to initiate a call to the contact number passed in the request.
    - The call is scheduled and delivered at the appointed time, with the customer hearing the personalized TTS message. The flow can also include an **interactive message flow** for further engagement (e.g., press 1 for sales, press 2 for support).

- **Key Steps**:
    - Customer data (first name, last name) is pushed to Jodo via API.
    - Jodo sends this data to the TTS engine configured by the user.
    - TTS engine generates an audio response.
    - Jodo calls the customer at the scheduled time, playing the interactive voice message.

Using Postman :
Request URL: https://api.jodo.com/CustomerWebServices/handleCustomerWS
(Note: This base URL may change based on the deployment environment
eg. Server IP Address `172.16.17.168`*)*

Method : POST
Header : 'Content-Type: text/html'
Request Body (Text) :

```
data={"method":"ImportProcessData","servicekey":"MUMBAI_CRM_OBD","apikey":"test_org","
media":"12","updateparam":"mobilenumber","data":
[{"sr_number":"22856","campaign_name":"NPS","comments":"Customer
comment","feedback_score":"1","mobilenumber":"08852245996"}]}
```

It is also possible to use CURL command to test API:

```
curl --location --request POST    'https://api.jodo.com/CustomerWebServices/handleCustomerWS'
\
--header 'Content-Type: text/html' \
--data-raw
'data={"method":"ImportProcessData","servicekey":"MUMBAI_CRM_OBD","apikey":"test_org","
media":"12","updateparam":"mobilenumber","data":
[{"sr_number":"22856","campaign_name":"NPS","comments":"Customer
comment","feedback_score":"1","mobilenumber":"08852245996"}]}'
```

It is also possible to use Shell-wget to test API

```
wget --no-check-certificate --quiet \
  --method POST \
  --timeout=0 \
  --header 'Content-Type: text/html' \
  --body-data
'data={"method":"ImportProcessData","servicekey":"MUMBAI_CRM_OBD","apikey":"test_org","
media":"12","updateparam":"mobilenumber","data":
[{"sr_number":"22856","campaign_name":"NPS","comments":"Customer
comment","feedback_score":"1","mobilenumber":"08852245996"}]}
' \
   'https://api.jodo.com/CustomerWebServices/handleCustomerWS'
```

SOAP UI, Postman are other tools that can be used to test API

Common Status Code Information

Status	Status Description
1107	API key not found
1108	Service key not found
1243	Service Key API Key Not matching

User Input Parameters:

Node	M/O	Data type size	Value	Description
Method	M	Text	ImportProcessData	Unique Command for JSON Request. (This method value cannot be changed)
apikey	M	Text	test_org	Unique key mapped to an organisation. (API key will change according to client organisation)
servicekey	M	Text	VoiceProcess	Unique key mapped to the voice media process. (Service key will change according to client process)
Media	M	Text	11	Media for the received data. (e.g., 1=Email In (Incoming / received email) 2=Email Out (Outgoing / Sent email) 3=SMS In (Incoming / received SMS) 4=SMS Out (Outgoing / Sent SMS)

				8=Chat (Website / messenger Chat conversation) 11=Voice In (Incoming / received Voice Call / session) **12=Voice Out (outgoing / Sent Voice call / session)** 21=Social Media – Twitter channel 22=Social Media -Facebook channel 31=Video Refer table below for Media Information values

Data	M	Text	[
			{
			"sr_number":"1458",
			"campaign_name": "NPS",
			"comments": "Customer comment",
			"feedback_score": "8",
			"mobilenumber": "8850014998"
			}
			]
				Customer data in JSON Array
				(Needs mobile number, name, email ID).
				All fields to be stored in Jodo DB should be passed to API

updateparam	M	Text	mobilenumber	The parameter used for data check to update and insert data.
reqdatetime	M	Text	2024-10-30 12:14:46	Current Date and Time of the request. (Date format: YYYY-MM-DD hh:mm:ss)

Multiple trunks handle prefix for domestic and international calling differently and based on the Trunks used in your operations you will need to standardise the data in your applications with the appropriate prefix.
e.g., GSM trunks / Gateways can process + for international dialing as a prefix while TELCO trunks need you to assign 00 as prefix for international dialing. Similarly domestic dialing code prefix may be different in parts of the world.

Developer should replace + with dialing prefix based on trunk configuration

Output Parameters:

Node	Value	Description
method	ImportProcessData	Unique Command for JSON Response
resdatetime	2024-10-30 12:14:46	Date and Time of the response. (Date format: YYYY-MM-DD hh:mm:ss)
status	1	Refer the status parameter table
statusdesc	success	status parameter description

Return Value:

This API returns one of responses listed below in JSON format.

Request with updateparam mobilenumber:

Request :

```
{
  "method": "ImportProcessData",
  "servicekey": "clicktocall_457_23122019105710",
  "apikey": "e763f560e120b316ea7ea4f2de646d780e64457c160c",
  "media": "12",
  "updateparam": "mobilenumber",
  "data": [
    {
      "sr_number":"1458",
      "campaign_name": "NPS",
      "comments": "Customer comment",
      "feedback_score": "8",
      "mobilenumber": "8850014998"
    }
  ]
}
```

ImportProcessData JSON Success Response

```
{
  "resdatetime": "2024-10-30 12:14:46",
  "statusdata": [
    {
      "statuscode": "001",
      "mobilenumber": "8850014998",
      "customercode": 125,
      "calllistid": 796,
      "statusdesc": "Success",
      "info": "##NEW_CUSTOMER"
    }
  ],
  "method": "ImportProcessData",
  "status": 1
}
```

ImportProcessData JSON Fail Response

```
{
  "method": "ImportProcessData",
  "statusdesc": "Invalid Media",
  "status": "017",
  "timestamp": "2024-10-30 13:55:47"
}
```

NOTE : Api Key And Service Key will different for different environment

Data Information

Json Parameter	M/O	Data type size	Value	Description
sr_number	O	Text - 15	1458	Unique SR Number
campaign_name	O	Text - 100	NPS	Source For Dialing
comments	O	Text - 200	Customer comment	Customer comment
feedback_score	O	Text - 10	8	Feedback Score
mobilenumber	M	Text - 10	08852245996	10 digit Contact number (Do not add Country Code)

Example :

```
"data": [
  {
    "sr_number":"1458",
    "campaign_name": "NPS",
    "comments": "Customer comment",
    "feedback_score": "8",
    "mobilenumber": "8850014998"
  }
]
```

Data Status Code Information

Status	Status Description
001	Success
050	Mobile Number not found

049	Mobile Number found blank
043	Invalid Mobilenumber
024	Update parameter not found
036	Session id not found
069	Field List not found
037	Scripttemplate not defined with process

Response Status Data Information

Json Parameter	Data type size	Value	Description
statuscode	Text	001	Status Code for each request
statusdesc	Text	Success	Status Description for each request
mobilenumber	Text	8850014998	Source For Dialing
customercode	Numeric	**125**	Unique Customer Code for each request. It’s advisable to store for future reference.
calllistid	Numeric	796	Incremental Call List Id
info	Text	##NEW_CUSTOMER	Additional Information to support status description

Example

```
"statusdata": [
  {
    "statuscode": "001",
    "mobilenumber": "8850014998",
    "customercode": 125,
    "calllistid": 796,
    "statusdesc": "Success",
    "info": "##NEW_CUSTOMER"
  }
 ],
```

Status Code Information

Status	Status Description
1	Success
14	Customer data not found
15	Date range not proper
16	Invalid To Date format
17	Invalid Media
18	Media Not found
19	Media Found blank
24	Update Parameter not found
25	Update Parameter blank
26	Invalid Update Parameter
32	DATA parameter not found
36	Invalid Session
38	Error while processing data
40	search param key not found
42	Invalid Search Param key
43	Invalid Mobile Number
44	Invalid Customer Code
45	Invalid CRM Unique Code
46	Invalid From Date format
48	API key expired
49	Mobile Number found blank
50	Mobile Number not found
51	Customer code found blank
52	Customer Code not found
53	CRM Unique field found blank
54	CRM Unique field not found
55	Invalid Call Date Time
61	service_key_API_key_matching_not_found
62	Custom filter not found
63	Custom filter found blank
66	Filter values are invalid

67	Invalid operator found
69	Field List not found

CALL Type Code Information

Call Type	Call Type Description
5	Hot Lead
21	Callback Call
31	Preview Callback Call
41	General call back
71	Preview Call
1001	Fresh Call

Media type:

Media Id	Media Information
1	Email In
2	Email Out
3	SMS In
4	SMS Out
8	Chat
11	Voice In
12	Voice Out
21	Social-Twitter
22	Social-Facebook
31	Video