



API QuickStart

Integrate with RingPlan SANDBOX DEV API guide



Thank You For Selecting RingPlan as Your Service Provider

Learn how you can integrate your technologies with RingPlan.

Why is This Document Important?

Use this document to create a successful integration with RingPlan.

From Our Knowledge Base

SANDBOX DEV ENV API GUIDE

Initial Steps

To work with Open API, you need to get an `id_token`.
To do this, you need to register

This can be done via the web interface my.dev.ringplan.com

You can also register via the API:

Request Example:

```
curl --request POST 'https://b2clogin.dev.ringplan.com/api/user/register' \
  \
  --header 'Content-Type: application/json' \
  --data-raw '{
    "first_name": "SomeName",
    "last_name": "SomeLastname",
    "email": "someemail@gmail.com",
    "company": "Some Company Name",
    "key": "b6ae17b92f60d3110c2cDsI90!dK5!1P", (for dev)
    "pwd": "SomeStrongPassword" (optional, without it password will be
generated)
  }'
```

1 | Login to Your Portal: Before attempting the next step, first visit <https://my.ringplan.com> and “log in” with your account.

2 | ID Token: After registration or logging in, you can get `id_token` and `refresh_token` here: <https://b2clogin.dev.ringplan.com/show-token>
For `id_token`, the lifetime is 8 hours, for `refresh_token` - 336 hours
you should receive something similar to the following:

```

JSON  Raw Data  Headers
Save Copy Collapse All Expand All Filter JSON
_id: "5fa043012ebb5ad62af6f14a"
access_token: "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImtpZCI6IjNpUDRvM2cyZHpQOFQxcXJYWjFZQ
Uzwyf3X9VGH0rKhItPeA8WUL-C9oHVVpRSTA8-tUdEobPhg05fyQGcjqXq0X7rYtYQerKGaz4
expires_at: 1604367233.5812545
expires_in: 28800
expires_on: 1604367233
id_token: "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImtpZCI6IjNpUDRvM2cyZHpQOFQxcXJYWjFZQ
lbIt0Ec8X7AGzy33VpeYT2cvNj8ZqZaGeExBQT_rur3kZedRn8u_r9v8cJEe0VKLXJa_CSHNe9
id_token_expires_in: 28800
not_before: 1604338433
profile_info: "eyJ2ZXIiOiIxLjAiLCJ0aWQiOiJlN2JmZmU2YS03YzlkLTQ4NjQtOGZiZi02YjM2ZWMyMmQ4M
refresh_token: "eyJraWQiOiJ3Q0RtM0lnTjZaT0NRYzI2di1lbFZPUUtxcU85TXAwOUcwS3lZZk02MEswIiwid
ZE6Hh8MGDxBMz9C25CtVMciMMK7ny4vq83hEovDkrJD7wP47LKVh6960Gh6umIefGis31C00Y
refresh_token_expires_in: 1209600
resource: "dd07e555-e1b6-4e5b-ba8b-b52d2afed9b8"
scope:
  0: "/"
  1: "offline_access"
  2: "openid"
token_type: "Bearer"

```

If you don't then you need to log in with the same browser you're using to get the show-token url from.

After registration, a client account will be created for the primary user of the specified company, and for full-fledged work it will be preferable to go through “onboarding”, during which all the initial settings for the company and this user will be made.

You can also make the same settings yourself.

Refresh Tokens

If you know the refresh_token, then you can get a new id_token using it if the id_token has expired.

It Looks Like This:

```
curl --request GET
'https://b2clogin.dev.ringplan.com/refresh-token?refresh_token=eyJraWQiOi
J3Q0RtM0lnTjZaT0NRYzI2di1lbFZPUUtxcU85TXAwOUcWS3lZzk02MEswIiwidmVyIjo
iMS4wIiwieG90IjoiRGVmbGF0ZSIsInNlciI6IjEuMCJ9.peMK5J-KaVe9cM5nT6gFU5IX
in0eFbeA8Gs3VCRY75iCREFoJVRZnVUGGCrAl8_bIth3eg7ZF11Aic_MjI5B2vVbX8sr
WT21p2AbJjJiGOHk4J3Xfuq1ZG2VO6fZTHAqGtfjGKEH27KkHTCuLztrCrDtPQjBiIbbp
8B5pttu8eauVRPT4Mfr-IFl6qzEzSMc4vQWGH9rfSrp7az8mJOhKKddEYI7Y9sezdFW8
DDLcEamR_0tBeF8dWWd2CpPX_RsaDCYouCvI601gldg7wRDm69Pej85Eivt1npu2qFQj9
D-roy-KiGes5H1GcSFJn2_CVqtzOYZD-WobV8z8ojujg.1OYzMf-cZdjvZNS.l2RPhycem
afgLK94INBrQgP9jfpPYyoESkIIbOcwI5digCCeDklrPmvczr97ysnBtNC6SYnisYOtZ
YQ928S6lJX7vAKCt1FURMV1QKP1wT8mApGMTUB07jO9IsiBJe_0cL4t8FpcenOwLJm
myAW2tGcaTzF-FMKYMLTvCu7PhHdJVon39csMeEhpsG7ZTqzay45t69cW1PrUvWjs
98XraiGC6j9ceh9-5A-HIYamrsfm3MMk4dZxNVYdnyqbU33h3buroWtnLBab9jH3xmWp
025n-bcZOLmhRaEi5DiNe4ahErnyw4590KWrx1-EdtMWay31Tek3XcmqWAbUDxplneus
Rq-8FtnFH1eJ21hnOgbQA5xEuGg089ctvdc7kq6-LoSP1YHdgrUCpw7QV3sd03oYIi
fW7Y6JLLveaQMKP-yqSKT6Qtow-RZYJJK184axbfGyIWqu7oacdPr9ffRKWIaDiJr3
s3iNsFDF0Pqh08T6-pOCRXU5pvA0E_-ZoXtEmYk67G7s5ZpjhbCtHUoaxPJNcHVUFz
LynuDGNoX-GiO7HFC2lpGkJsBJ05p-fwdG-WAXibjDSly6HxHbZhadwCFddcWFLsEH1
ZwqM6cg8jgYVbAE5gfvFGdy3t8jdHyT2onkxwWRG8xA5A156zqrIkJo14A.CHqtVIfe
Fh1dbQfgnPdfbw'
```

Response Sample:

```
{
  "id_token":
  "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImtpZCI6IjNpUDRvM2cyZHpQOFQxcXJY
WjFZQXNzN1dhY19zSmNpclkaGRiRDBqa1EifQ.eyJleHAiOiJlMjMDQ0Mzc5MTQsIm5i
ZiI6IjE2MTYwNDQwOTExNCwidmVyIjoiaHR0cHM6Ly9kZXZyaW5ncGxhbi5iMmNsb2dp
bi5jb20vZTdiZmZlNmEtN2M5ZC00ODY0LTNmYmYtNmIzNmVjMjJkODExL3YyLjAvIiw
iwiOiI2MjNjZTMxMmMtODRhZC00ZDE5LTkzMGMtMTIyMjhmYWM0MTk4IiwiaWF0Ijoi
ZGQwN2U1NTUtZ"
```

```

TFiNi00ZTViLWJhOGItYjUyZDJhZmVkOWI4IiwiaWF0IjoxNjA0NDA5MTE0LCJhdXRoX3RpbW
UiOjE2MDQ0MDg5NzgsImdpdmVuX25hbWUiOiJEZXYiLCJmYW1pbHlfbmFtZSI6Ik5vRGVsZXR
lIiwizW1haWxzIjpbImRldnRlc3RAc2VudHJpdW0uaW8iXSwidGlkIjoiZTdiZmZlNmEtN2M5
ZC00ODY0LThmYmYtNmIzNmVjMjJkODExIiwiaXRFaGFzaCI6IjFnSWlYakxqcmlONXlybVBST
FRoZXcifQ.aEaXkWSTPoLjFGmTrezBPRlCvYIBqoa6vQ2BtIBWNf87isq0JK85f8F1_JV73LS
rH0nCDGgtb_X3jHAeUspkrPU-ReVOuwj8Uf34wUoPT81FWnCSHgo-HE6GvgC17L3E0YR-z1WJ
4f8E9dRF06DVP5PRFzx75FwImQhj76I9BLPefLMjzYDkbyOPTcqvLdRzdm3AL4h_HnaXLnjZb
BOSkd7UDiJqnkj8G7ghywgWkekFpmu58QcC2af2Q3K0Ff75t9XB7SMfgzrxwlizjoTpsubJAC
Toq_dewM6tmLAVlpBGzpYj0pKlJXSV-UGjvAJ9BpwmTdGXdUSN2ulMEwT92g",
  "refresh_token":
  "eyJraWQiOiJ3Q0RtM0lnTjZaT0NRYzI2di1lbFZPUUtxcU85TXAwOUcwS3lZzk02MEswIiwiaXNjaW50IjoiMS4wIiwiaW1lIjoiRGVmbGF0ZSIsInNlciI6IjEuMCJ9.aFLtmoOBGgpPeSYy3VFW
Yj38GUYQZ4rSazSo4aRH7VPjroOnG8r4yGSLZ-2tCUsd6BX9f6otq3yKUMqjjyW7P8iyKltbH
nsubC2aQEW_9gmx3G5qZg-99Vizcpcefbgds1xDSyVSuJnJXsnAIzSm3i_2xx6h7v32umate
LZgMQcJXR0452veWT9dvP60Ss5bXJ_QgrgTotlu90D2kn8vY5-RwZuQTWUIPjvXwANSJh_TUF
XX2VWLYDhEq9gZe35wQEr1qLtor8dVx4ZzT_m04wWB1BsVh8Kd_Euf0acyj-s3MPUh1TtmLdA
fTgGGG1w11q8DBh5caA3J0gZFVCnkQ.z0fr7OW5VptyFTpd.UfcXrsOxld3mybCNlovWEpZmY
pWVHQBmrsWWvlHCrvWvuvyQLbizIcRk6iNzhi5zvTmlA6XRgDYi_4vb2bPQ5rO2NphdZYcOuT
YpAa4Wir_nraURqjBDK2heTr-Nh3_vFXC8djtUGAHnPWoxYQDX1lMxsQXcf0jo_CvT7qI89E-
c2pvjh74Dvnf0VpyJKnS18d6HJazmVa0KBvs4vt1GcAEcBzG8-dOfaIac6b-vK6J2hSBsbCaI
C9HzkM-7fKdluBtjVbuqtg7pIGimZsVQfjeKiv4fGgbfkJFBWBqFbxqoZempfBHxBRjY0Y4qI
fdf76dirhxIdqpXrLi7AwLqKB5w_kyiv2xyHQ8UoAR9JKOJTKPfxhNMEdf-KEiMwT_QA0sDwX
A4OstDPCfCohYP9iKvIYG0B1S1ChKEcXXV9svgegswAgQ9HBulrMvSNH7xaXQ3wtgvh-hIEDB
ihxV9tiVQFVap1PpJF19b2A6uWx-rGruMVxDz0FCoByVNOVqDe5TH7aD3zvgUw8yjTkQ90d86
GvA6or2FkfAG_D824z4mXPZ0C6o2Qe0iGwfGeq6MCpqMNDpU2LYJUgAuhpyJQb2Lwcjq9qQMG
hEKJHgR-J2NdnDCrmlen8yj4wa6jyB2QSXcJtTIpyFT88dPanWPfavfjWPoYw.dZhx5Zdb43M
FrVYGkzuFDw"
}

```

For example, if an id_token is used in the API and you receive an error related to its expiration, you can use a request to update tokens. After that, the tokens must be saved for future use.

From Our Knowledge Base

Using Swagger UI

<https://ssp-backend.dev.ringplan.com/ui/#/>

To work in the Swagger UI, you need to log in using the received `id_token`

ssp-backend.dev.ringplan.com/ui/#!/Texting324532Contacts/controllers_texting_create_contact

Texting - Contacts Show/Hide List Operations Expand Operations

GET /texting/contacts Return filtered list of contacts

POST /texting/contacts Endpoint for creating a contact record

Implementation Notes
Contact creation. `contact_id` is required for conversation creation. After contact creation you can create conversation with this contact. `company_id` parameter will be added automatically

Response Class (Status 200) !
Contact already exist for company

Model | **Example Value**

```
{
  "company_id": "121d95391afa0914b47b9d54",
  "first_name": "Matt",
  "id": "9490637dafff2380dc05bbe0",
  "last_name": "Wiz",
  "number": "8123382823"
}
```

Response Content Type:

Parameters

Parameter	Value	Description	Parameter Type	Data Type
data	(required)		body	undefined

Parameter content type:

Response Messages

HTTP Status Code	Reason	Response Model	Headers
201	Contact created		

Click on the red icon on the upper right corner.



Texting - Contacts Show/Hide List Operations Expand Operations

GET /texting/contacts Return filtered list of contacts

POST /texting/contacts Endpoint for creating a contact record

Implementation Notes
 Contact creation. contact_id is required for conversation creation. After contact creation you can create conversation with this contact. company_id parameter will be added automatically

Response Class (Status 200) !
 Contact already exist for company

Model | **Example Value**

```
{
  "company_id": "121d9539",
  "first_name": "Matt",
  "id": "9490637dafff2380",
  "last_name": "Wiz",
  "number": "8123382823"
}
```

Available authorizations

Api key authorization

name: authorization
 in: header
 value:

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
data	(required)		body	undefined

Parameter content type:

Response Messages

HTTP Status Code	Reason	Response Model	Headers
201	Contact created		

Enter your id_token



The icon color changes to blue.

Now you can use the swagger web interface to learn the API

Texting

In order to be able to send and receive messages, you need to configure the main number for messages. This option is currently available from the web interface and will be added to the SSP Open API in the near future.

If the number is configured, you can get it here:

https://ssp-backend.dev.ringplan.com/ui/#!/Texting324532Numbers/controllers_texting_get_numbers_ownership

You will receive information back such as the following:

```
Curl
curl -X GET --header 'Accept: application/json' --header 'authorization: eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImtpZCI6Ij
```

```
Request URL
https://ssp-backend.dev.ringplan.com/texting/numbers
```

```
Response Body
[
  {
    "company_id": "5f74dee2e56bb5f12fa22f4c",
    "id": "5f75f385bf99409ca2294bbb",
    "number": "8583715117",
    "ownership": [
      "jjiao@ringplan.com",
      "jbartolli@ringplan.com",
      "jsweere@ztelco.com",
      "jsmith@ztelco.com",
      "jhansen@ztelco.com"
    ]
  }
]
```

```
Response Code
200
```

```
Response Headers
{
  "access-control-allow-origin": "*",
  "access-control-expose-headers": "*",
  "content-length": "291",
  "content-security-policy": "default-src https: data: 'unsafe-inline' 'unsafe-eval'",
  "content-type": "application/json",
  "date": "Mon, 02 Nov 2020 17:54:17 GMT",
  "server": "nginx",
  "strict-transport-security": "max-age=31536000; includeSubdomains",
  "x-content-type-options": "nosniff",
  "x-firefox-spdy": "h2",
  "x-frame-options": "SAMEORIGIN",
  "x-xss-protection": "1; mode=block"
}
```

Contact Creating

Before you can send a text, it is first necessary to create a contact object.

https://ssp-backend.dev.ringplan.com/ui/#!/Texting324532Contacts/controllers_texting_create_contact

Example of request parameters:

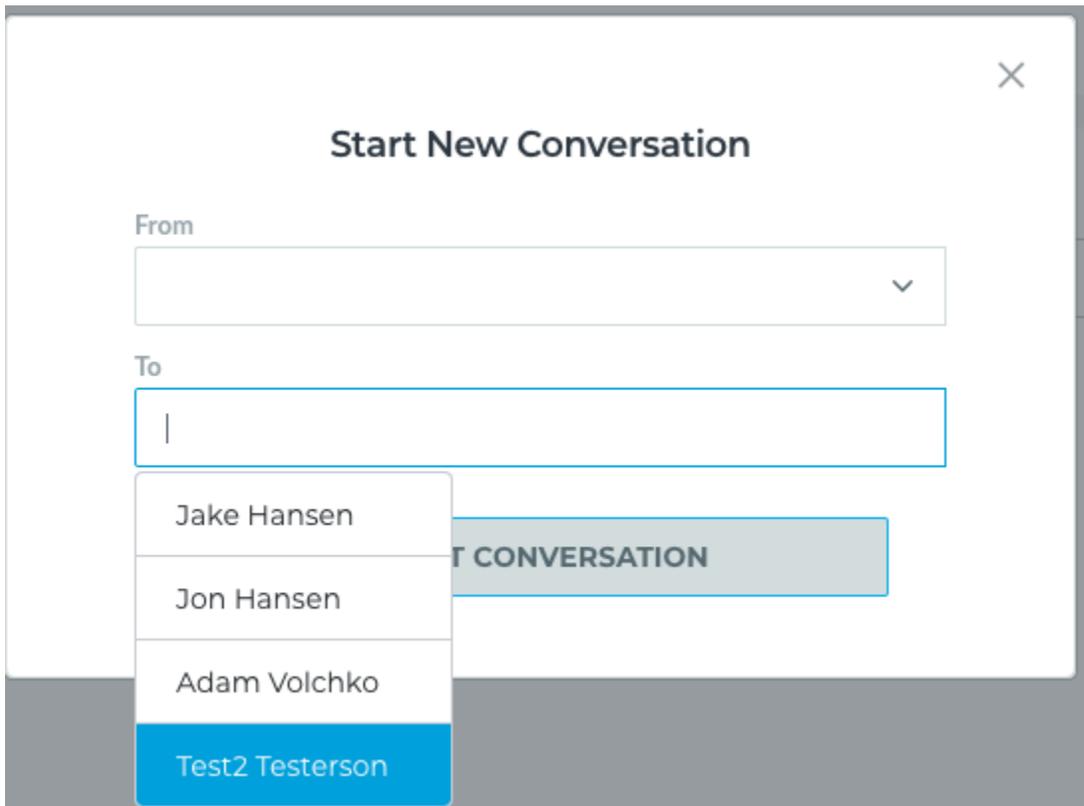
Parameter	Value
data	<pre>{ "first_name": "Test2", "last_name": "11111", "number": "8123382222" }</pre>

Parameter content type:

You will see something like the following:

HTTP Status Code	Reason	Response Model	Headers
201	Contact created		
Try it out! Hide Response			
Curl			
<pre>curl -X POST --header 'Content-Type: application/json' --header 'Accept: application/json' --header 'authorization: eyJ</pre> <pre>{</pre> <pre> "first_name": "Test2", \</pre> <pre> "last_name": "Testerson", \</pre> <pre> "number": "8586881741" \</pre> <pre>}' 'https://ssp-backend.dev.ringplan.com/texting/contacts'</pre>			
Request URL			
<pre>https://ssp-backend.dev.ringplan.com/texting/contacts</pre>			
Response Body			
<pre>{</pre> <pre> "company_id": "5f74dee2e56bb5f12fa22f4c",</pre> <pre> "first_name": "Test2",</pre> <pre> "id": "5fa04b0816ce9ade1c20ad60",</pre> <pre> "last_name": "Testerson",</pre> <pre> "number": "8586881741"</pre> <pre>}</pre>			
Response Code			
<pre>200</pre>			
Response Headers			
<pre>{</pre> <pre> "access-control-allow-origin": "https://ssp-backend.dev.ringplan.com",</pre> <pre> "access-control-expose-headers": "*",</pre> <pre> "content-length": "162",</pre> <pre> "content-security-policy": "default-src https: data: 'unsafe-inline' 'unsafe-eval'",</pre> <pre> "content-type": "application/json",</pre> <pre> "date": "Mon, 02 Nov 2020 18:09:48 GMT",</pre> <pre> "server": "nginx",</pre> <pre> "strict-transport-security": "max-age=31536000; includeSubdomains",</pre> <pre> "vary": "Origin",</pre> <pre> "x-content-type-options": "nosniff",</pre> <pre> "x-firefox-spdy": "h2",</pre> <pre> "x-frame-options": "SAMEORIGIN",</pre> <pre> "x-xss-protection": "1; mode=block"</pre> <pre>}</pre>			

You will also find in the SSP that you can now start a text conversation with this contact:



contact_id is required for conversation creation. After contact creation you can create conversation with this contact. Company_id parameter will be added automatically.

Contact List for Texting

https://ssp-backend.dev.ringplan.com/ui/#!/Texting324532Contacts/controllers_texting_get_contacts

contact_id is required for conversation creation. After contact creation you can create conversation with this contact.

Get Contact

https://ssp-backend.dev.ringplan.com/ui/#!/Texting324532Contacts/controllers_texting_get_contact

Only one parameter needs to be specified - `contact_id`

Update Contact:

https://ssp-backend.dev.ringplan.com/ui/#!/Texting324532Contacts/controllers_texting_update_contact

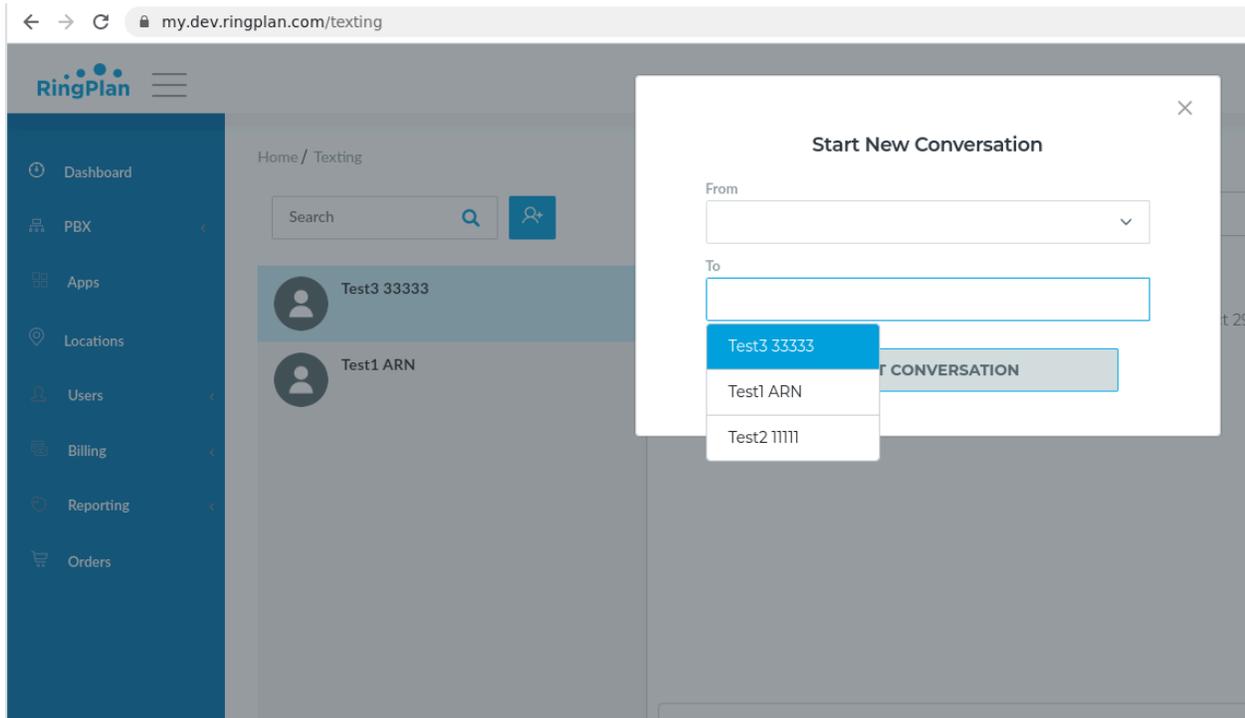
Example of request parameters:

Parameters	
Parameter	Value
contact_id	<input type="text" value="5f9ac44c660b1197e693295a"/>
data	<pre>{ "first_name": "Test3", "last_name": "33333", "number": "8123383333" }</pre>
Parameter content type: <input type="text" value="application/json"/>	

Get List of Conversations

https://ssp-backend.dev.ringplan.com/ui/#!/Texting324532Conversations/controllers_texting_get_conversations

Object represent information about communication between number and contact.
The same list in the Self Service Portal Web interface:



Create New Conversation:

https://ssp-backend.dev.ringplan.com/ui/#!/Texting324532Conversations/controllers_texting_create_conversation

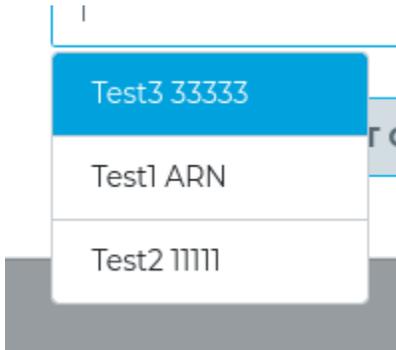
Example of request parameters:

Parameters

Parameter	Value
data	<pre>{ "contact_id": "5f9be654660b1197e69329a9", "from_number": "8123382222" }</pre>

Parameter content type:

Creating a new conversation will add it to the list in the Self Service Portal Web interface:



List of Messages in Conversation:

https://ssp-backend.dev.ringplan.com/ui/#!/Texting324532Messages/controllers_texting_message_list

Example of Request Parameters:

Parameters		
Parameter	Value	Description
conversation_id	<input type="text" value="5f9aca67660b1197e693295f"/>	
page	<input type="text" value="1"/>	current page number
per_page	<input type="text" value="20"/>	size of the pages in pagination

You will get a response similar to the following:

Parameter	Value	Description	Parameter Type	Data Type
conversation_id	5f7cffc81f4245401f63a506		path	string
page	1	current page number	query	integer
per_page	20	size of the pages in pagination	query	integer

Try it out! [Hide Response](#)

Curl

```
curl -X GET --header 'Accept: application/json' --header 'authorization: eyJ0eXAI0iJKV1QilCJhbGciOiJSUzI1NiIsImtpZCI6Ij
```

Request URL

```
https://ssp-backend.dev.ringplan.com/texting/conversations/5f7cffc81f4245401f63a506/messages?page=1&per_page=20
```

Response Body

```
{
  "conversation_id": "5f7cffc81f4245401f63a506",
  "created_at": "2020-11-02T18:41:03.447000",
  "direction": "outbound",
  "files": [],
  "frm": "8583715117",
  "id": "5fa052bf16ce9ade1c20ad62",
  "media_urls": [],
  "success": false,
  "text": "test",
  "to": "8586881740"
},
{
  "conversation_id": "5f7cffc81f4245401f63a506",
  "created_at": "2020-10-06T23:38:11.390000",
  "direction": "outbound",
  "files": [
    {
      "company_id": "5f74dee2e56bb5f12fa22f4c",
      "description": null,
      "id": "5f7c561c0e0175011c20ad62"
    }
  ]
}
```

Send an Outbound SMS or MMS Message:

https://ssp-backend.dev.ringplan.com/ui/#!/Texting324532Messages/controllers_texting_publish_message

Example of request parameters:

Parameters

Parameter	Value
conversation_id	<input type="text" value="5f9aca67660b1197e693295f"/>
data	<pre>{"text": "Lorem ipsum dolor sit amet"}</pre>

Parameter content type:

Meeting API

Create History event (not private)

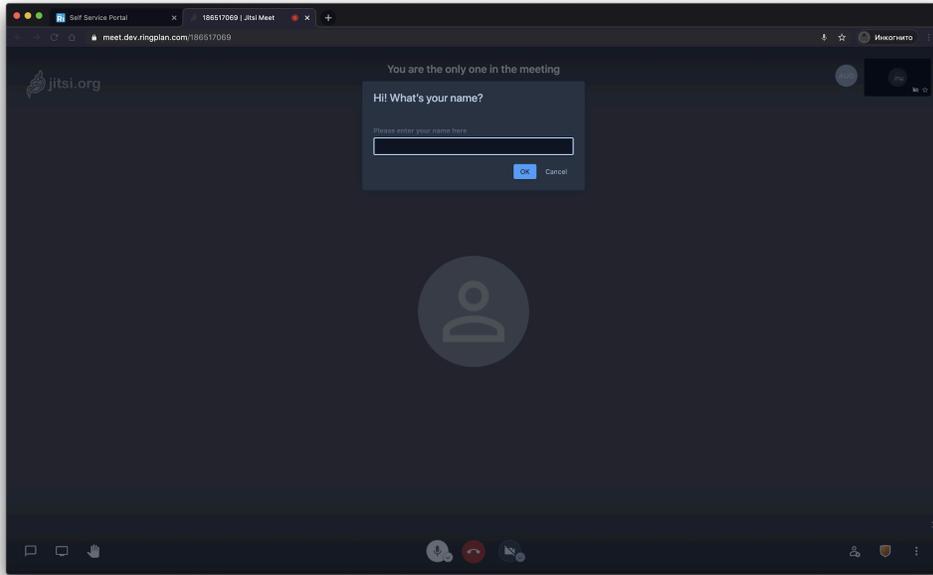
SSP Backend Example

[Go to Apps => Ringplan Meet](#)

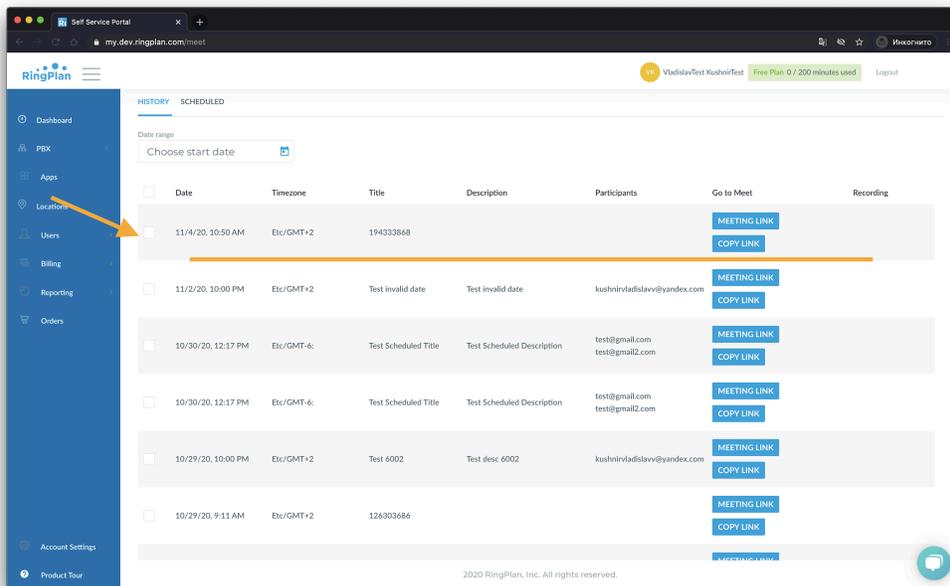
The screenshot shows the RingPlan Meet web application. At the top, there is a navigation bar with the user's name 'VladislavTest KushnirTest', a plan indicator 'Free Plan 0 / 200 minutes used', and a 'Logout' button. Below the navigation bar, the main content area is titled 'Meet' and contains a 'Welcome to RingPlan Meet!' message. It prompts the user to 'Go ahead and start a new video conference with your entire team.' and includes a 'Make it private' checkbox. There are two main buttons: 'START A NEW MEETING' (highlighted in blue) and 'SCHEDULE'. Below this, there are tabs for 'HISTORY' and 'SCHEDULED'. The 'HISTORY' tab is active, showing a table of meeting events. The table has columns for 'Date', 'Timezone', 'Title', 'Description', 'Participants', 'Go to Meet', and 'Recording'. One event is listed with the date '11/2/20, 10:00 PM', timezone 'Etc/GMT+2', and title 'Test invalid date'. The 'Go to Meet' column for this event contains 'MEETING LINK' and 'COPY LINK' buttons. At the bottom of the page, there is a footer with the copyright notice '© 2020 RingPlan, Inc. All rights reserved.' and a chat icon.

Push the button *“Start a new meeting”*

After clicking on the button, we will be redirected to the page in jitsi



The newly created meeting will be displayed in the list.



Swagger example

We used this endpoint for meeting create history event

POST [/meet/calendar/event/history](#)

We need to enter an authentication token in order to be able to send requests.

User_id on SSP-Backend is automatically substituted from the authentication token

Event Show/Hide List Operations Expand Operations

POST /meet/calendar/event/history Create a history event

Implementation Notes
Create a scheduled appointment in google calendar

Response Class (Status 201)
Successful

Model Example Value

```
{
  "_id": "5ed547b2ce4e5ac3876d7e26",
  "event": {
    "date_end": "2020-10-18 12:17:16.1532",
    "date_start": "2020-10-18 12:17:16.1532",
    "description": "Test Description",
    "event_type": "history",
    "hash_password": "3875034e17855bac83a3cc9e107b1d28a9b44313d381c3335588525b4e78b55b",
    "link_on_meet": "https://meet.ringplan.com/899717532",
    "meeting_id": "800717532"
  }
}
```

Response Content Type application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
data	(required)		body	Model Example Value

Parameter content type: application/json

Response Messages

HTTP Status	Reason	Response Model	Headers
-------------	--------	----------------	---------

Model Example Value

```
{
  "description": "Test History Description",
  "end_time": "2020-10-18T12:17:16.153321",
  "password": "TestPassword",
  "start_time": "2020-10-18T12:17:16.15332",
  "summary": "Test History Summary",
  "time_zone": "Etc/GMT+0",
  "title_meet": "Test History Title"
}
```

We fill Data for send:

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
data	<pre>{ "description": "Test History Description", "end_time": "2020-11-04T10:17:16.153321", "start_time": "2020-11-04T10:17:16.153321", "summary": "Test History Summary", "time_zone": "Etc/GMT+0", "title_meet": "Test History Title" }</pre>		body	Model Example Value

Parameter content type:

```
{
  "description": "Test History De
  "end_time": "2020-10-18T12:17:1
  "password": "TestPassword",
  "start_time": "2020-10-18T12:17
  "summary": "Test History Summar
  "time_zone": "Etc/GMT+0",
  "title_meet": "Test History Tit
}
```

Response Messages

HTTP Status Code	Reason	Response Model	Headers
------------------	--------	----------------	---------

example data:

```
{
  "description": "Test History Description",
  "end_time": "2020-11-04T10:17:16.153321",
  "start_time": "2020-11-04T10:17:16.153321",
  "summary": "Test History Summary",
  "time_zone": "Etc/GMT+0",
  "title_meet": "Test History Title"
}
```

start_time and end_time - should be the same

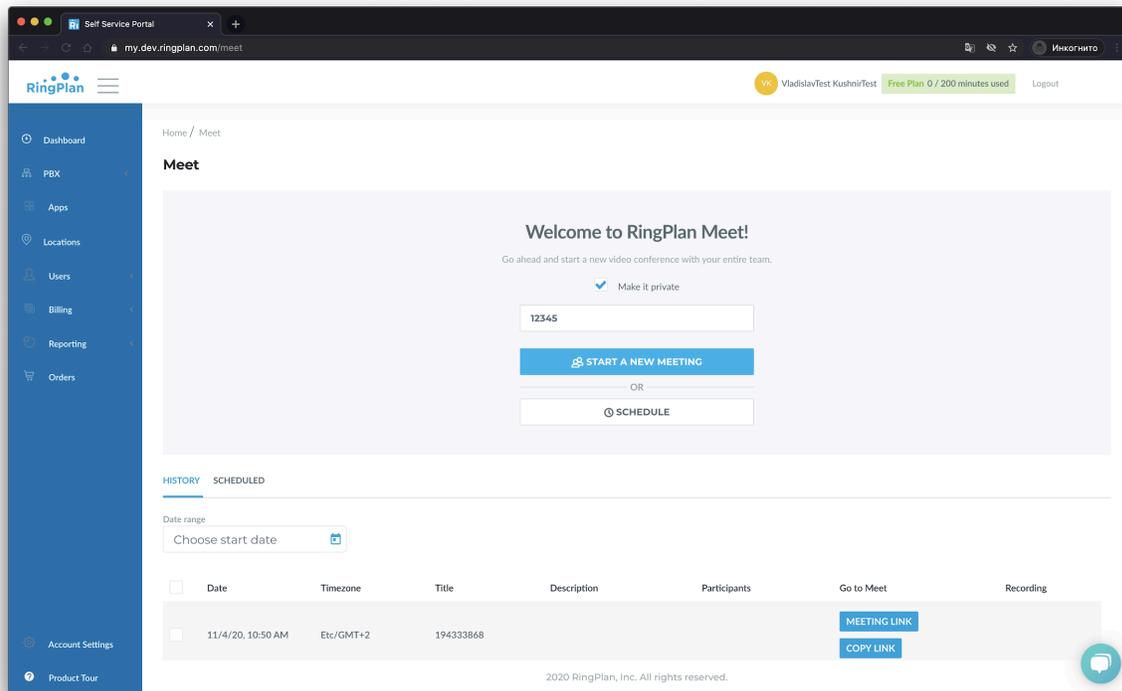
start_time should not be more than the date and time at the time of the event creation and should not be less than 48 hours from the moment the event was created

```
start_time < now_time_1
now_time_1 > now_time_2 - 24h
```

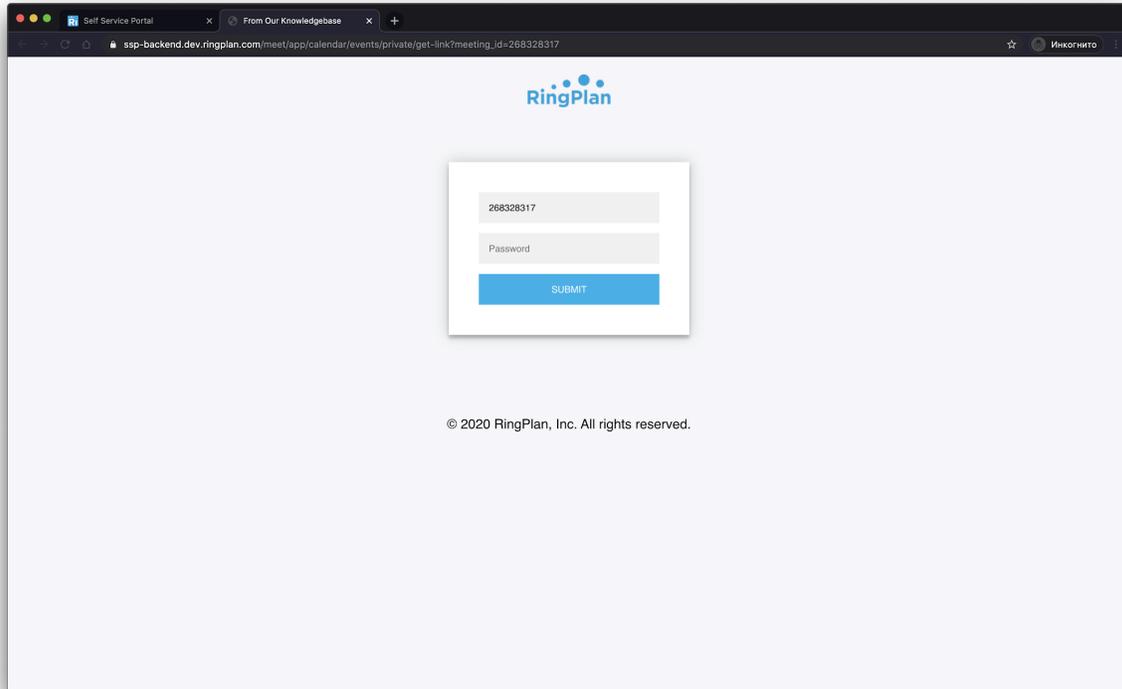

Create History event (private)

SSP Backend Example

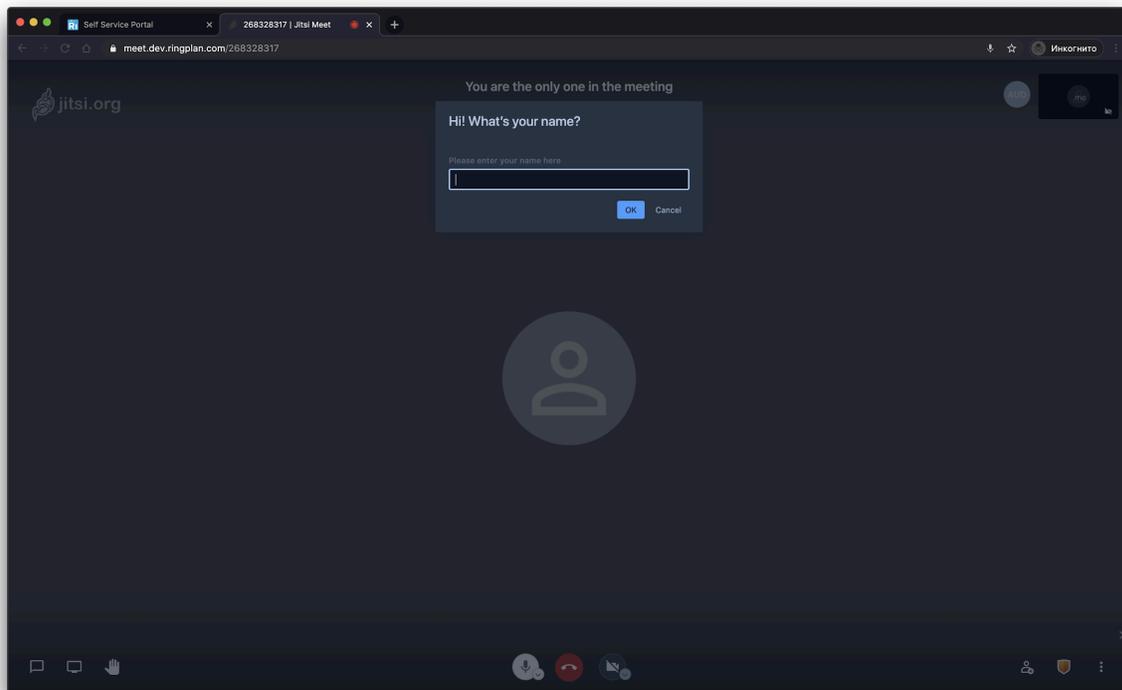
We must select the checkbox next to "Make it private" And enter the password
Then click on the "Start A New Meeting"



We will be redirected to the password entry page



After we enter the correct password, we will be redirected to the jitsi page



The newly created meeting will be displayed in the list

The screenshot shows the RingPlan Meet interface. At the top, there's a navigation bar with the RingPlan logo and a user profile for 'VladislavTest KushnirTest' with a 'Free Plan 0 / 200 minutes used' indicator. Below the navigation bar, a 'Welcome to RingPlan Meet!' message is displayed, followed by a 'Go ahead and start a new video conference with your entire team.' prompt. A 'Make it private' checkbox is checked. A meeting ID '12345' is entered in a text field. Below this, there are two buttons: 'START A NEW MEETING' and 'SCHEDULE'. The 'SCHEDULE' button is highlighted with a blue border. Below the buttons, there's a 'Date range' section with a 'Choose start date' dropdown. The main content area is a table with columns: Date, Timezone, Title, Description, Participants, Go to Meet, and Recording. The table contains three rows of meeting data. The first row is highlighted with an orange bar. The second row is also highlighted with an orange bar. The third row is not highlighted. The table footer contains the text '2020 RingPlan, Inc. All rights reserved.' and a chat icon.

<input type="checkbox"/>	Date	Timezone	Title	Description	Participants	Go to Meet	Recording
<input type="checkbox"/>	11/4/20, 11:21 AM	Etc/GMT+2	447205211			MEETING LINK COPY LINK	
<input type="checkbox"/>	11/4/20, 10:50 AM	Etc/GMT+2	194333868			MEETING LINK COPY LINK	
<input type="checkbox"/>	11/4/20, 10:17 AM	Etc/GMT-4:	250703883	Test History Description		MEETING LINK	

Swagger example

We used this endpoint for create history event

POST [/meet/calendar/event/history](#)

We need to enter an authentication token in order to be able to send requests.
User_id on SSP-Backend is automatically substituted from the authentication token

The screenshot displays the Swagger UI for the endpoint `POST /meet/calendar/event/history`. The main content area is titled "Event" and includes the following sections:

- Implementation Notes:** "Create a scheduled appointment in google calendar". A blue information icon (i) is located in the top right corner of this section, with an orange arrow pointing to it.
- Response Class (Status 201):** "Successful".
- Model Example Value:** A JSON object representing the response structure:

```
{  "_id": "5ed547b2ce4e5ac3876d7e26",  "event": {    "date_end": "2020-10-18 12:17:16.153Z",    "date_start": "2020-10-18 12:17:16.153Z",    "description": "Test Description",    "event_type": "history",    "hash_password": "3875034e17855bac83a3c9e107b1d28a9b44313d381c3335588525b4e70b55b",    "link_on_meet": "https://meet.ringplan.com/899717532",    "meeting_id": "800717532"  }}}
```
- Response Content Type:** `application/json`.
- Parameters:** A table with columns for Parameter, Value, Description, Parameter Type, and Data Type. It lists a required `data` parameter with a `body` type and `application/json` content type. An example JSON value is provided for the body parameter:

```
{  "description": "Test History Description",  "end_time": "2020-10-18T12:17:16.153Z",  "password": "TestPassword",  "start_time": "2020-10-18T12:17:16.153Z",  "summary": "Test History Summary",  "time_zone": "Etc/GMT+0",  "title_meet": "Test History Title"}
```
- Response Messages:** A section for HTTP status, reason, response model, and headers.

We fill Data for send:

The screenshot shows a REST client interface with the following sections:

- Response Content Type:** application/json
- Parameters:** A table with columns: Parameter, Value, Description, Parameter Type, Data Type. A parameter named 'data' is defined with a JSON body:

```
{ "description": "Test History Description", "end_time": "2020-11-04T10:17:16.153321", "start_time": "2020-11-04T10:17:16.153321", "summary": "Test History Summary", "time_zone": "Etc/GMT+0", "title_meet": "Test History Title", "password": "12345" }
```
- Parameter content type:** application/json
- Response Messages:** A table with columns: HTTP Status Code, Reason, Response Model, Headers.

example data:

```
{
  "description": "Test History Description",
  "end_time": "2020-11-04T10:17:16.153321",
  "start_time": "2020-11-04T10:17:16.153321",
  "summary": "Test History Summary",
  "time_zone": "Etc/GMT+0",
  "title_meet": "Test History Title",
  "password": "12345"
}
```

In order to create a private meeting, you need to add "password" to data

start_time and end_time - should be the same

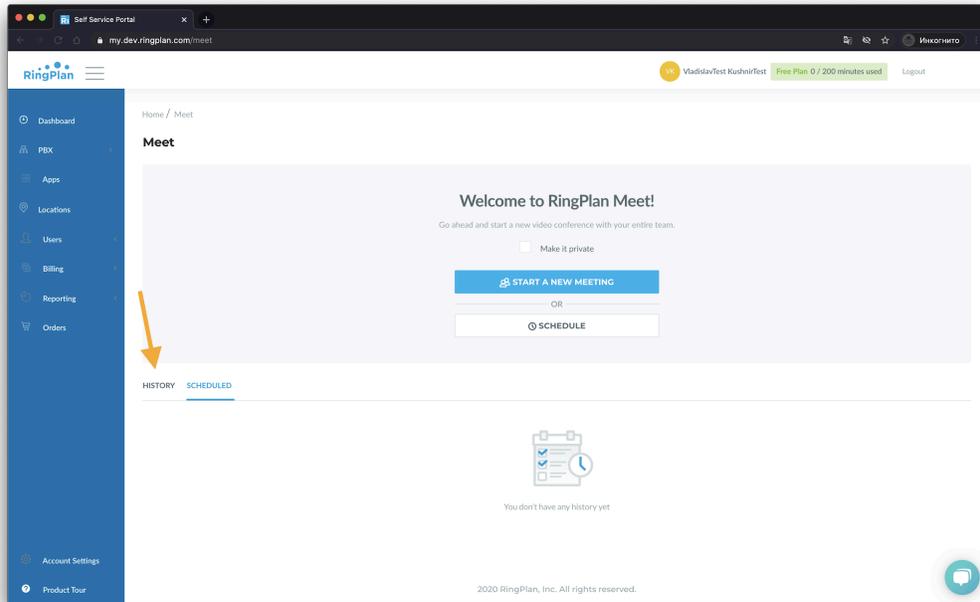
start_time should not be more than the date and time at the time of the event creation and should not be less than 48 hours from the moment the event was created

```
start_time < now_time_1
now_time_1 > now_time_2 - 24h
```


Get History Event (not private)

SSP Backend Example

Go to Apps => *Ringplan Meet*



Press “History”
And see all history event

Self Service Portal | my.dev.ringplan.com/meet

VladislavTest KushnirTest | Free Plan 0 / 200 minutes used | Logout

HISTORY | SCHEDULED

Date range: Choose start date

<input type="checkbox"/>	Date	Timezone	Title	Description	Participants	Go to Meet	Recording
<input type="checkbox"/>	11/4/20, 11:21 AM	Etc/GMT+2	447205211			MEETING LINK COPY LINK	
<input type="checkbox"/>	11/4/20, 10:50 AM	Etc/GMT+2	194333868			MEETING LINK COPY LINK	
<input type="checkbox"/>	11/4/20, 10:17 AM	Etc/GMT-8	250703883	Test History Description		MEETING LINK COPY LINK	
<input type="checkbox"/>	11/4/20, 10:17 AM	Etc/GMT-8	248660412	Test History Description		MEETING LINK COPY LINK	
<input type="checkbox"/>	11/2/20, 10:00 PM	Etc/GMT+2	Test invalid date	Test invalid date	kushnirvladislav@yandex.com	MEETING LINK COPY LINK	
<input type="checkbox"/>	10/30/20, 12:17 PM	Etc/GMT-8	Test Scheduled Title	Test Scheduled Description	test@gmail.com test@gmail2.com	MEETING LINK COPY LINK	

2020 RingPlan, Inc. All rights reserved.

Self Service Portal | my.dev.ringplan.com/meet

VladislavTest KushnirTest | Free Plan 0 / 200 minutes used | Logout

<input type="checkbox"/>	10/26/20, 12:17 PM	Etc/GMT-8	Test Scheduled Title	Test Scheduled Description	testing170120@gmail.com	MEETING LINK COPY LINK	
<input type="checkbox"/>	10/26/20, 11:11 AM	Etc/GMT+2	130671554			MEETING LINK COPY LINK	
<input type="checkbox"/>	10/25/20, 12:17 PM	Etc/GMT-8	Test Scheduled Title	Test Scheduled Description	testing170120@gmail.com	MEETING LINK COPY LINK	
<input type="checkbox"/>	10/23/20, 3:17 PM	Etc/GMT-8	Test History Title	Test History Description		MEETING LINK COPY LINK	
<input type="checkbox"/>	10/23/20, 12:17 PM	Etc/GMT-8	Test History Title	Test History Description		MEETING LINK COPY LINK	
<input type="checkbox"/>	10/22/20, 12:17 PM	Etc/GMT-8	Test History Title	Test History Description		MEETING LINK COPY LINK	
<input type="checkbox"/>	10/22/20, 12:17 PM	Etc/GMT-8	Test Scheduled Title1	Test Scheduled Description1	testing170120@gmail.com	MEETING LINK COPY LINK	

Show 20 | 1 | 2

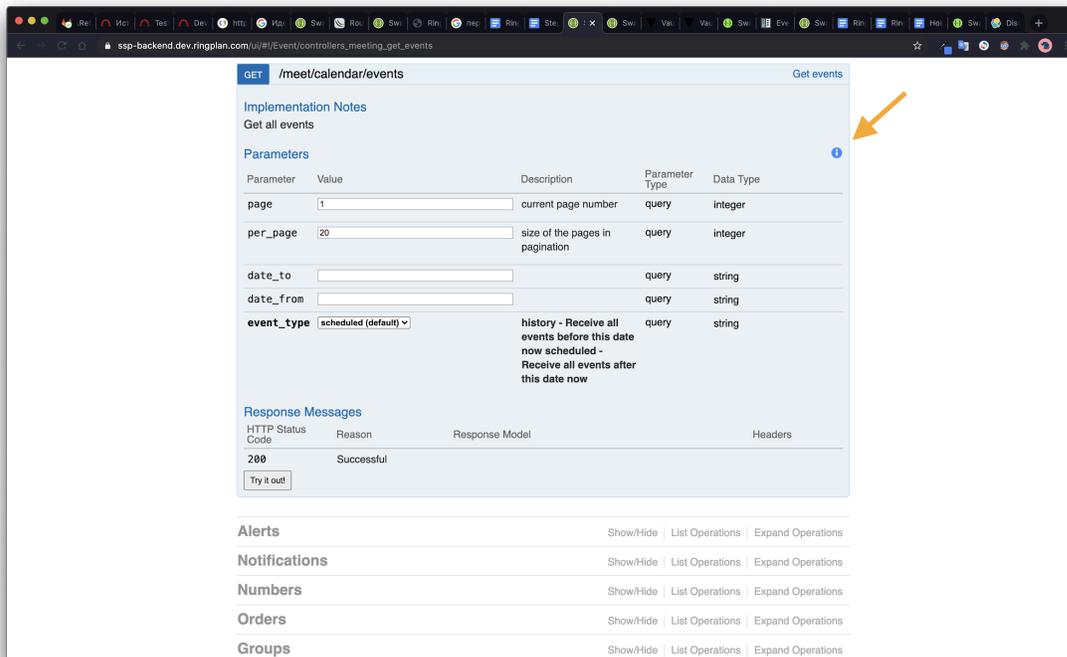
2020 RingPlan, Inc. All rights reserved.

Swagger Example

We used this endpoint for get history event

`GET /meet/calendar/events`

We need to enter an authentication token in order to be able to send requests.
User_id on SSP-Backend is automatically substituted from the authentication token



The screenshot displays the Swagger UI for the endpoint `GET /meet/calendar/events`. The interface includes the following sections:

- Implementation Notes:** "Get all events"
- Parameters:** A table with columns for Parameter, Value, Description, Parameter Type, and Data Type.

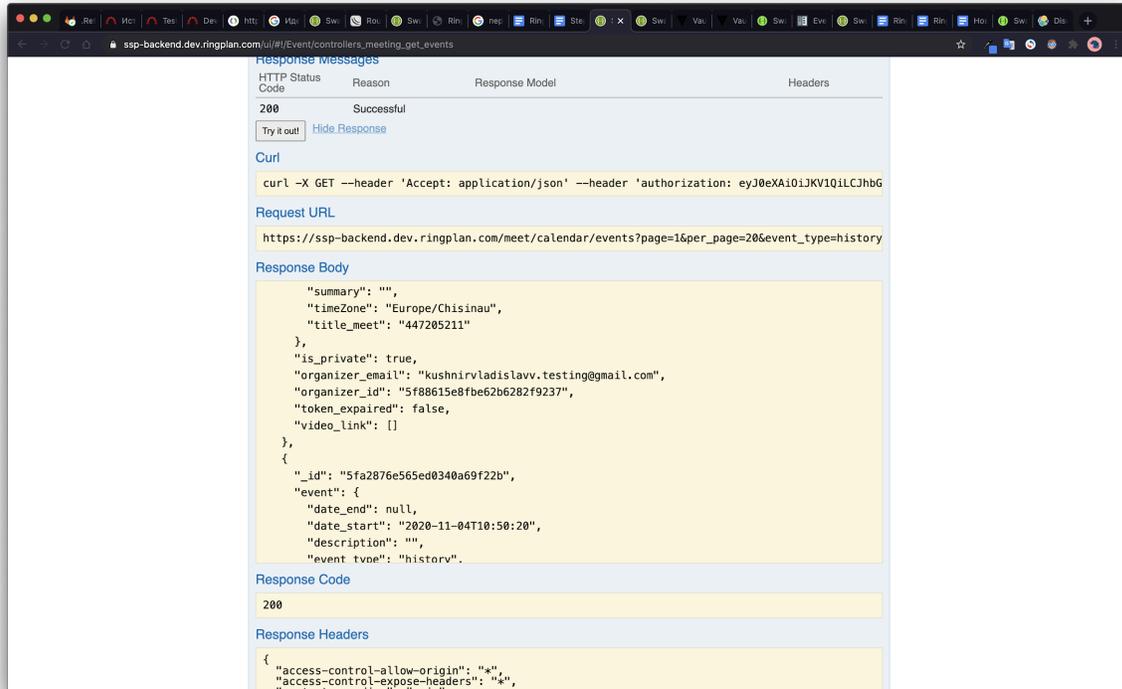
Parameter	Value	Description	Parameter Type	Data Type
page	1	current page number	query	integer
per_page	20	size of the pages in pagination	query	integer
date_to	<input type="text"/>		query	string
date_from	<input type="text"/>		query	string
event_type	scheduled (default)	history - Receive all events before this date now scheduled - Receive all events after this date now	query	string
- Response Messages:** A table with columns for HTTP Status Code, Reason, Response Model, and Headers.

HTTP Status Code	Reason	Response Model	Headers
200	Successful		

Below the main content area, there are sections for Alerts, Notifications, Numbers, Orders, and Groups, each with "Show/Hide", "List Operations", and "Expand Operations" links.

We must select the type "history"

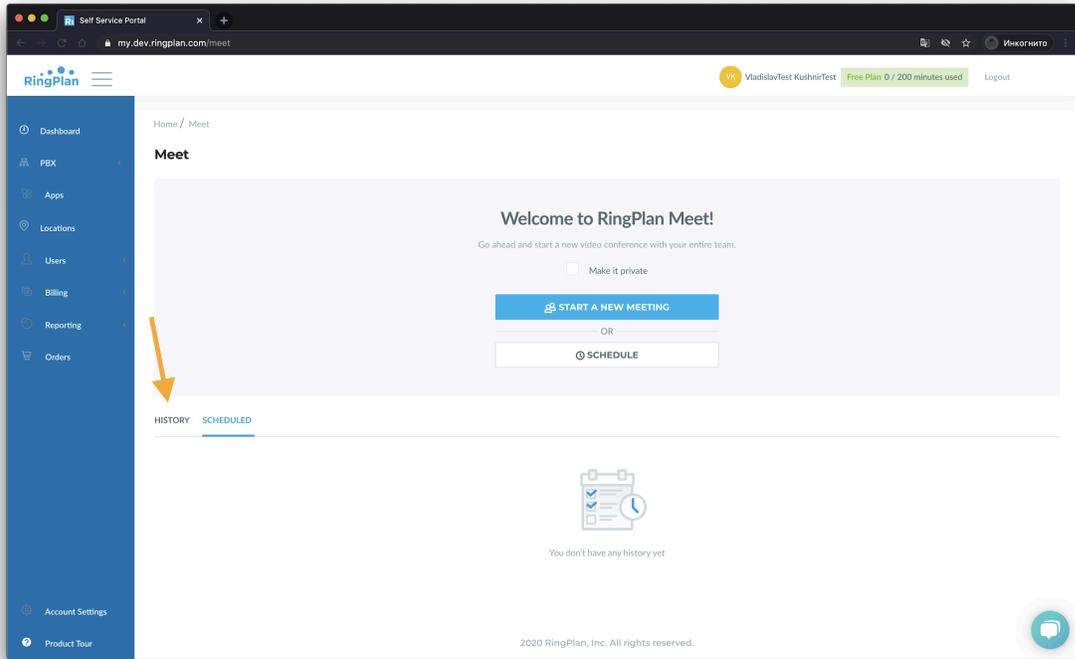
And after sending the request, we will receive a response with all history events



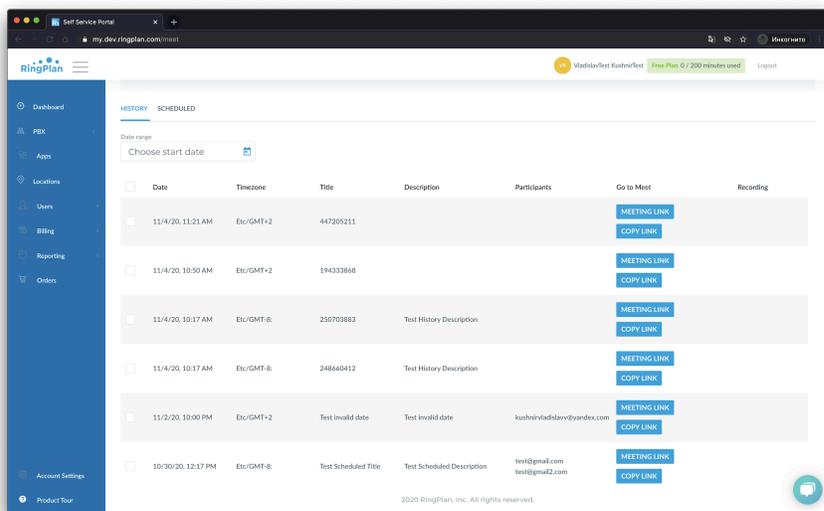
Get History event (private)

SSP Backend Example

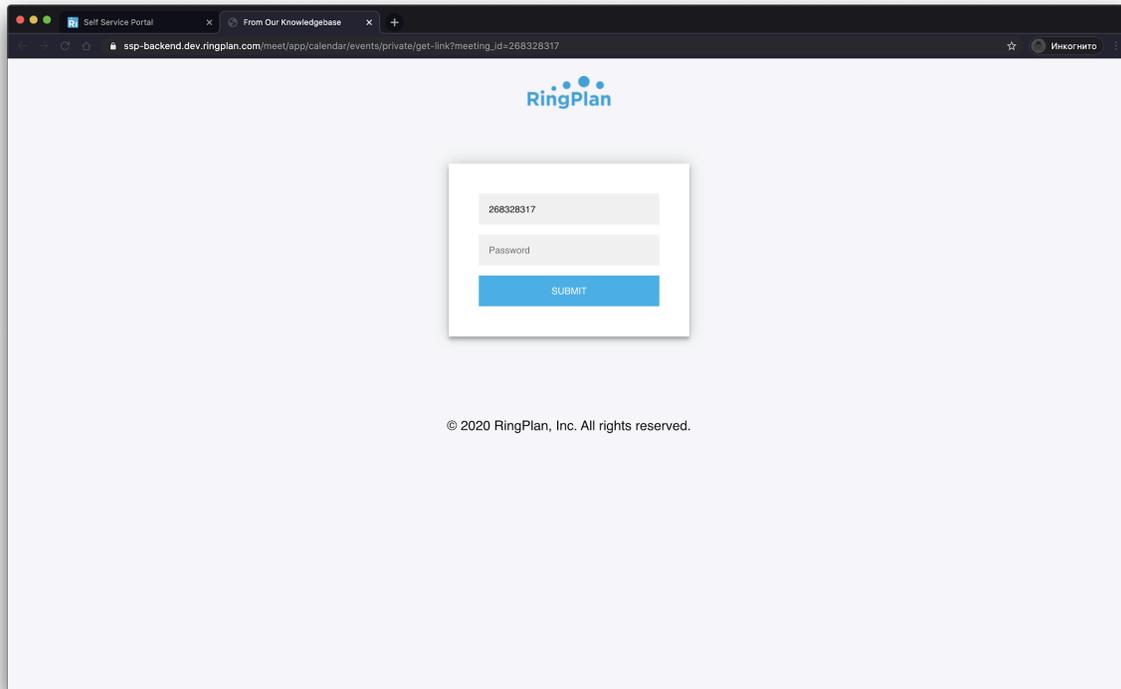
Go to Apps => *Ringplan Meet*



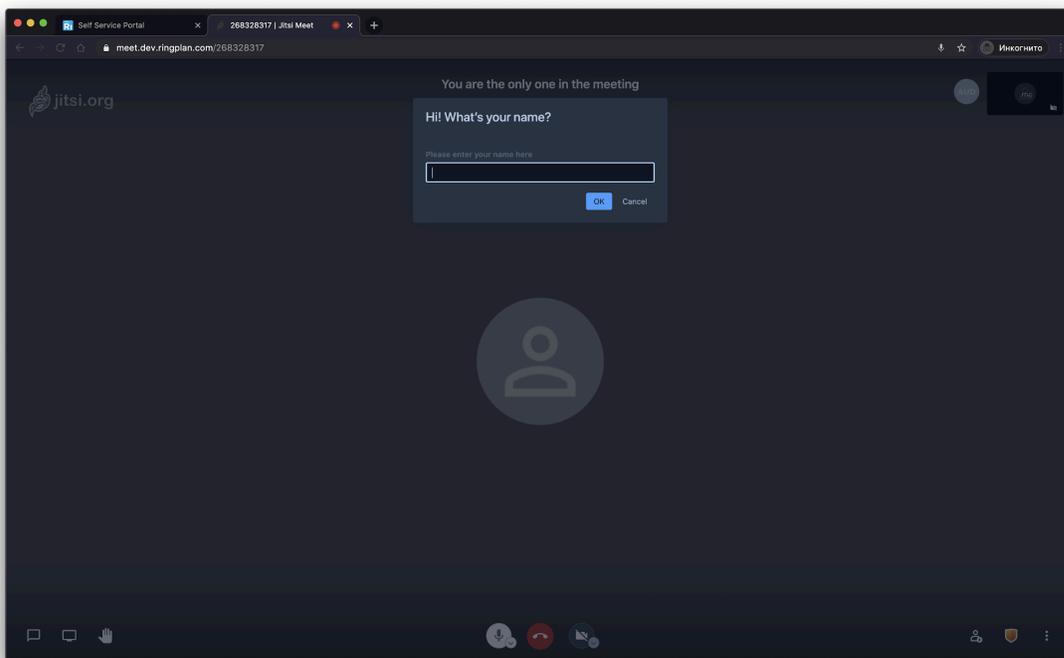
Press "History"
And see all history event



If we click the "Meeting Link" button and this meeting is private, We will be taken to the password entry page.



After we enter the correct password, we will be redirected to the jitsi page

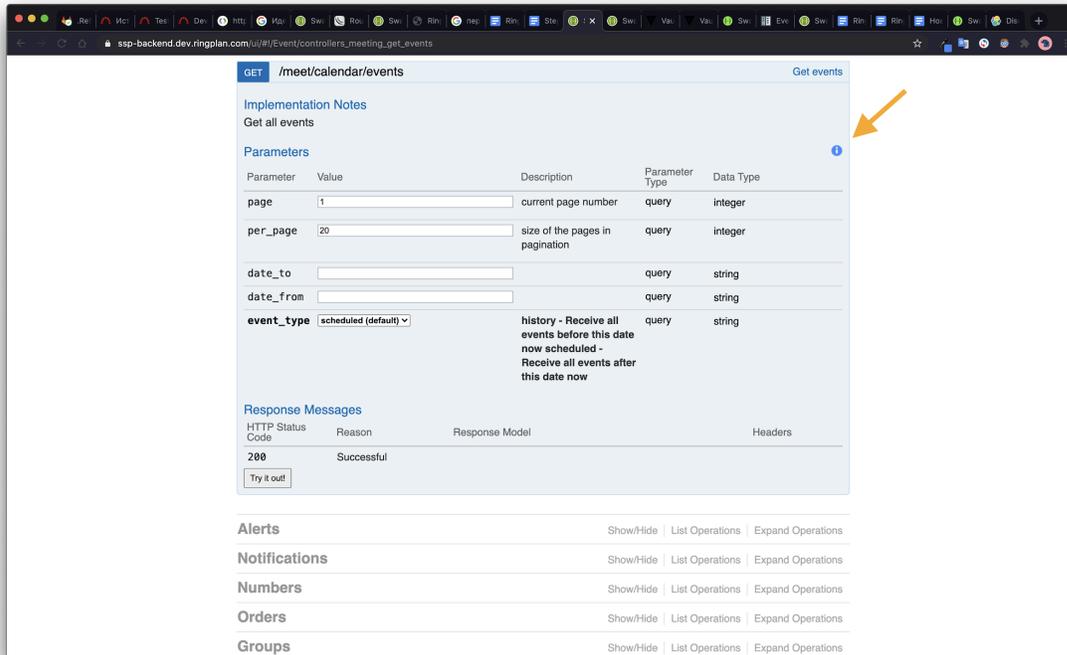


Swagger example

We used this endpoint for get history event

[GET /meet/calendar/events](#)

We need to enter an authentication token in order to be able to send requests.
User_id on SSP-Backend is automatically substituted from the authentication token



The screenshot displays the Swagger UI for the endpoint `GET /meet/calendar/events`. The interface includes the following sections:

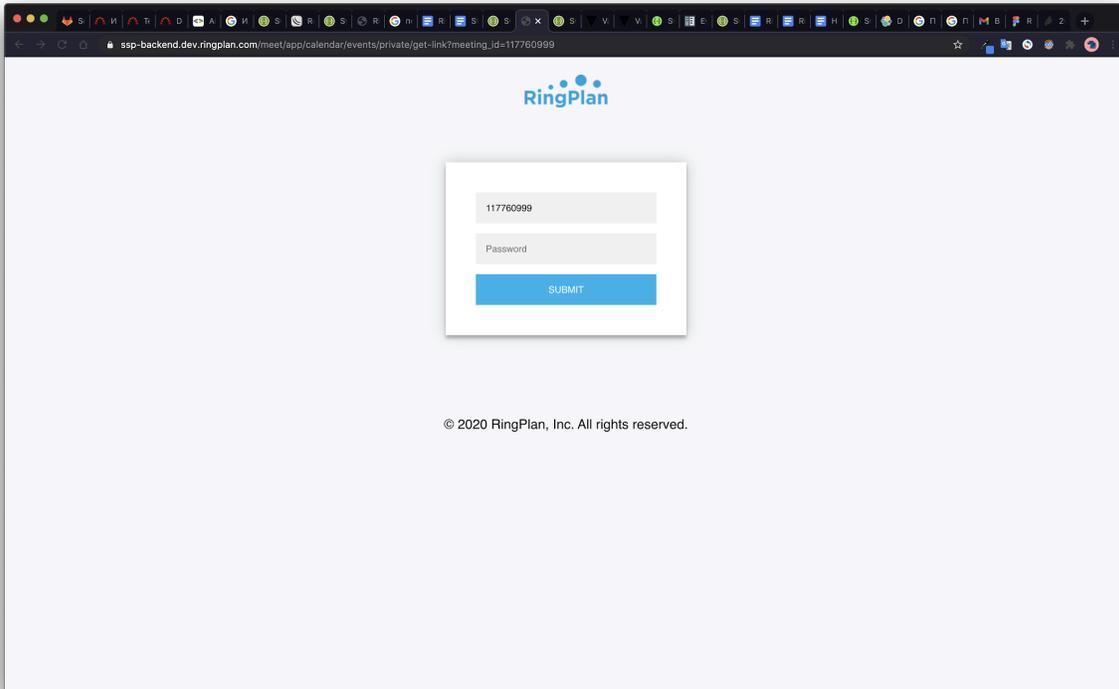
- Implementation Notes:** "Get all events"
- Parameters:** A table with columns for Parameter, Value, Description, Parameter Type, and Data Type.

Parameter	Value	Description	Parameter Type	Data Type
page	1	current page number	query	integer
per_page	20	size of the pages in pagination	query	integer
date_to			query	string
date_from			query	string
event_type	scheduled (default)	history - Receive all events before this date now scheduled - Receive all events after this date now	query	string
- Response Messages:** A table with columns for HTTP Status Code, Reason, Response Model, and Headers.

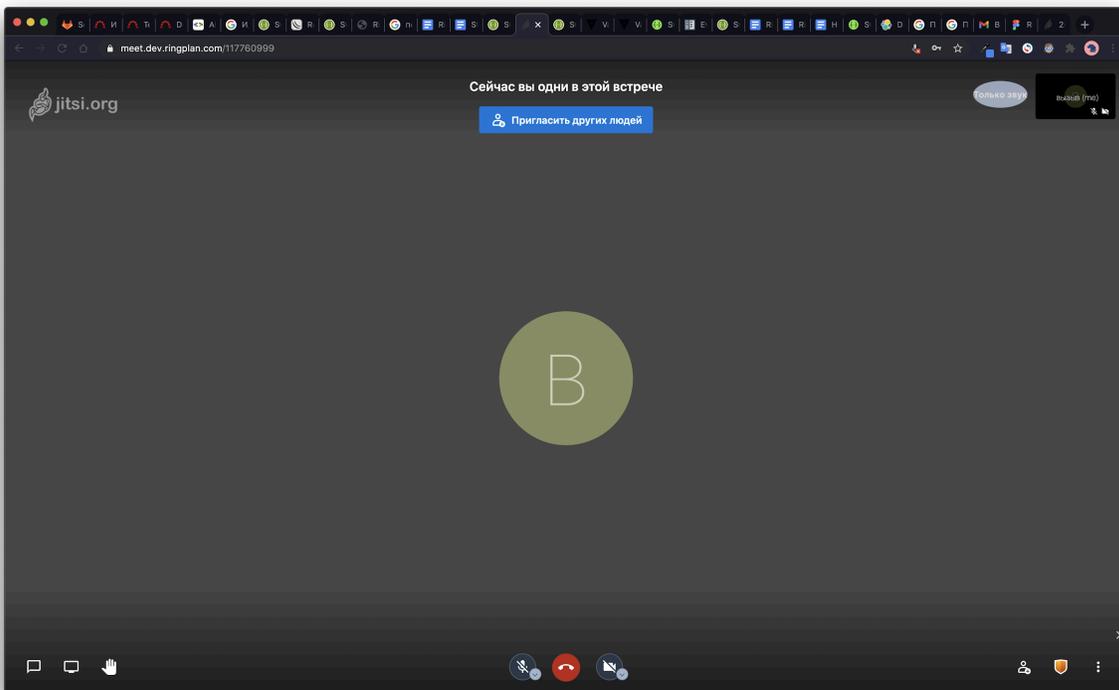
HTTP Status Code	Reason	Response Model	Headers
200	Successful		

At the bottom of the page, there are sections for Alerts, Notifications, Numbers, Orders, and Groups, each with "Show/Hide", "List Operations", and "Expand Operations" links.

We must select the type "history"



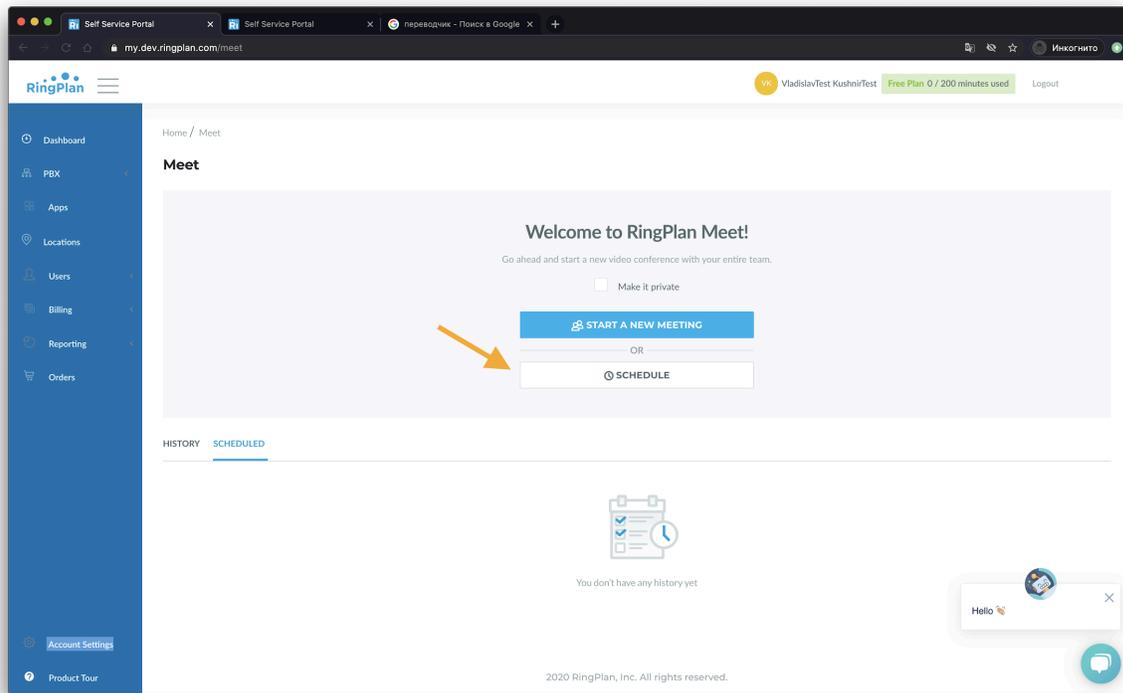
After we enter the correct password, we will be redirected to the jitsi page

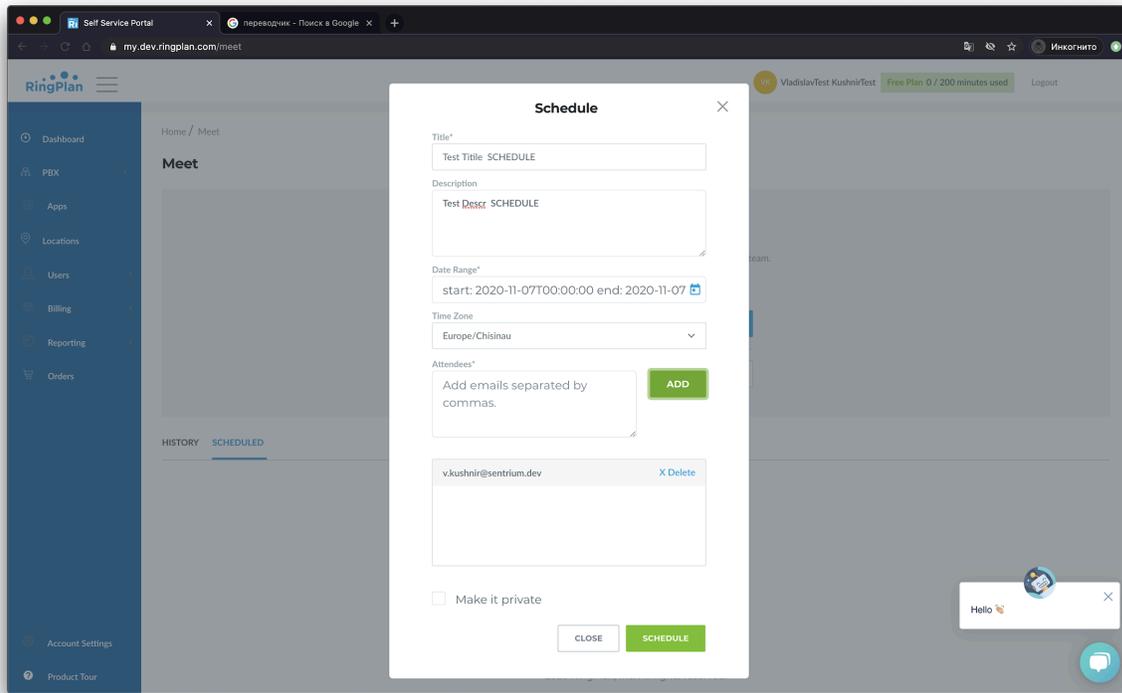


User Authorization in Google App

SSP Backend Example

Click on the button “*SCHEDULE*”





Fill in the data for sending the event. And if we do not have access to your account in the system, we will be transferred to the Google page when we try to create an event.

I have to select the same mail as indicated in the user profile

Home / Account Settings

Account Settings

USER PROFILE COMPANY

User Info

First Name
VladislavTest

Last Name
KushnirTest

Mobile number (sms notifications will be sent to this number)
1(318) 545-6266

Privacy & Time

Email
kushnirvladislav.testing@gmail.com

Password
To change password you will be redirected to Microsoft Azure. **CHANGE**

Timezone
You can search and select their timezone.
Europe/Chisinau

We need to grant access to the application.

After we have provided access, we will be sent to the event creation page

Example data:

```
{
  "attendees": [
    {
      "email": "v.kushnir@sentrion.com"
    }
  ],
  "description": "Test Scheduled Description",
  "end_time": "2020-11-07T12:25:16.153321",
  "start_time": "2020-11-07T12:17:16.153321",
  "summary": "Test Scheduled Summary",
  "time_zone": "Etc/GMT+0",
  "title_meet": "Test Scheduled Title"
}
```

POST /meet/calendar/event/scheduled Create a scheduled event

Implementation Notes
Create a scheduled appointment in google calendar

Response Class (Status 201) i
Successful

Model **Example Value**

```
"hash_password": "38/5034e17855bac03a3cc9e10/b1d28a9b44313d381c3335588525b4e/0b55b",
"link_on_meet": "https://meet.ringplan.com/899717532",
"meeting_id": "899717532",
"password": "Test".
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
data	<pre>{ "attendees": [{ "email": "v.kushnir@sentrion.com" }], "description": "Test Scheduled Description", "end_time": "2020-11-07T12:25:16.153321", "start_time": "2020-11-07T12:17:16.153321", "summary": "Test Scheduled Summary", "time_zone": "Etc/GMT+0", "title_meet": "Test Scheduled Title" }</pre>		body	Model Example Value
				<pre>{ "attendees": [{ "email": "test@gmail.com" }, { "email": "test@gmail2.com" }], }</pre>

Parameter content type:

Response Messages

HTTP Status Code	Reason	Response Model	Headers
404	Bad Request		

Request URL

```
https://ssp-backend.dev.ringplan.com/meet/calendar/event/scheduled
```

Response Body

```
{
  "detail": "This acc doesn`t have credentials for google app",
  "status": 404,
  "title": "Not Found",
  "type": "about:blank"
}
```

Response Code

```
404
```

Response Headers

```
{
  "access-control-allow-origin": "https://ssp-backend.dev.ringplan.com",
  "access-control-expose-headers": "*",
  "content-length": "133",
  "content-security-policy": "default-src https: data: 'unsafe-inline' 'unsafe-eval'",
  "content-type": "application/json",
  "date": "Fri, 06 Nov 2020 09:54:29 GMT",
  "server": "nginx",
  "status": "404",
  "strict-transport-security": "max-age=31536000; includeSubdomains",
  "vary": "Origin",
  "x-content-type-options": "nosniff",
  "x-frame-options": "SAMEORIGIN",
  "x-xss-protection": "1; mode=block"
}
```

Response Body:

```
{
  "detail": "This acc doesn`t have credentials for google app",
  "status": 404,
  "title": "Not Found",
  "type": "about:blank"
}
```

This means that there is no necessary access to your google calendar

To get a link for authorization in the Google application, we must use this endpoint

GET [/meet/app/calendar/get-auth-url](#)

Curl

```
curl -X GET --header 'Accept: application/json' --header 'authorization: eyJ0eXAiOiJKV1QiLCJhbG
```

Request URL

```
https://ssp-backend.dev.ringplan.com/meet/app/calendar/get-auth-url
```

Response Body

```
{
  "auth_url": "https://accounts.google.com/o/oauth2/auth?response_type=code&client_id=308654"
}
```

Response Code

```
200
```

Response Headers

```
{
  "access-control-allow-origin": "*",
  "access-control-expose-headers": "*",
  "content-length": "347",
  "content-security-policy": "default-src https: data: 'unsafe-inline' 'unsafe-eval'",
  "content-type": "application/json",
  "date": "Fri, 06 Nov 2020 09:59:04 GMT",
  "server": "nginx",
  "status": "200",
  "strict-transport-security": "max-age=31536000; includeSubdomains",
  "x-content-type-options": "nosniff",
  "x-frame-options": "SAMEORIGIN",
  "x-xss-protection": "1; mode=block"
}
```

Response Body:

```
{
  "auth_url":
  "https://accounts.google.com/o/oauth2/auth?response_type=code&client_id=3086540204
  17-t5t0iqhdu188fdlhun1qd2aossqrpm71.apps.googleusercontent.com&redirect_uri=https%
  3A%2F%2Fmeetauth.dev.ringplan.com%3A443%2F&scope=https%3A%2F%2Fwww.googleapis.com%
  2Fauth%2Fcalendar&state=kushnirvladislavv.testing%40gmail.com&access_type=offline"
}
```

In response, we will receive a link to which we need to redirect the user.

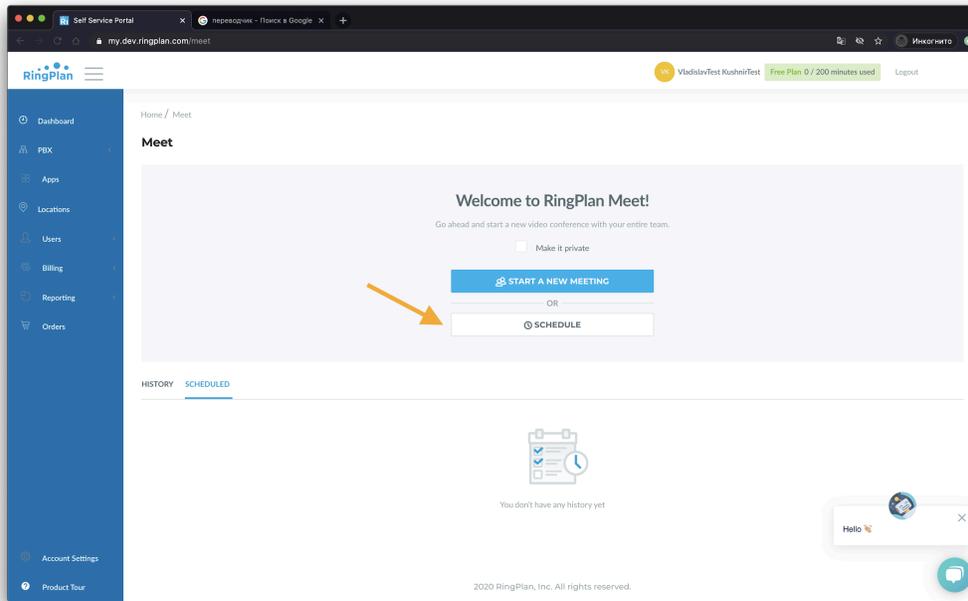
I have to select the same mail as indicated in the user profile

We need to grant access to the application.

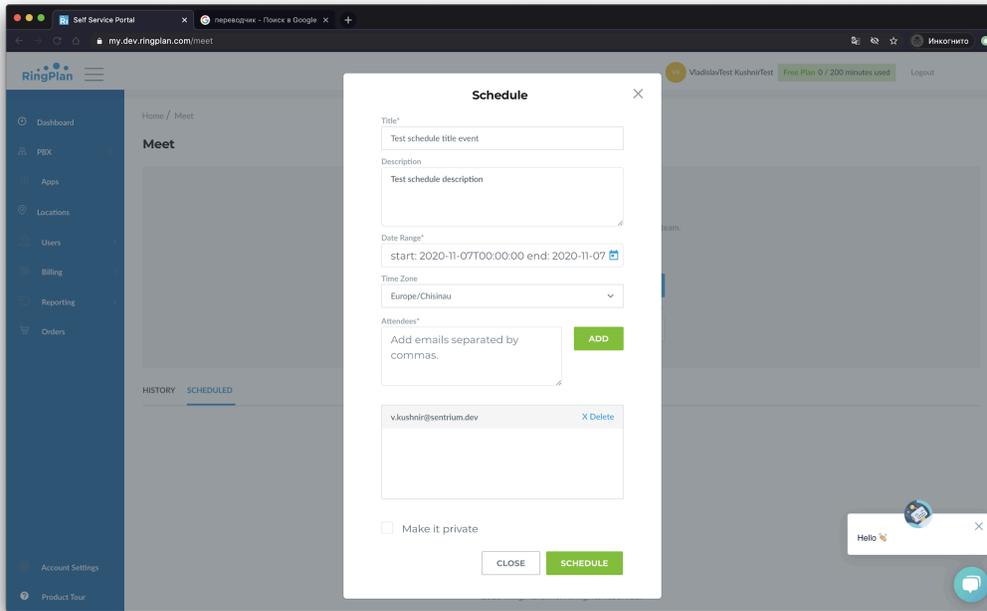
Create Schedule event (not private)

SSP Backend example

Go to **Apps => Ringplan Meet**



Press button **“Scheduled”**
Fill in the required information



This event should appear in the organizer's google calendar and invitations should be sent to the participants

Swagger example

We used this endpoint for get history event

POST </meet/calendar/event/scheduled>

POST </meet/calendar/event/scheduled> Create a scheduled event

Implementation Notes
Create a scheduled appointment in google calendar

Response Class (Status 201) Successful

Model | **Example Value**

```
"hash_password": "3875034e17855bac03a3cc9e107b1d28a9b44313d381c3335588525b4e/0b55b",  
"link_on_meet": "https://meet.ringplan.com/899717532",  
"meeting_id": 899717532,  
"password": "Test".
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type				
data	<pre>{ "attendees": [{ "email": "v.kushnir@sentrrium.com" }], "description": "Test Scheduled Description", "end_time": "2020-11-07T12:25:16.153321", "start_time": "2020-11-07T12:17:16.153321", "summary": "Test Scheduled Summary", "time_zone": "Etc/GMT+0", "title_meet": "Test Scheduled Title" }</pre>		body	<table border="1"><thead><tr><th>Model</th><th>Example Value</th></tr></thead><tbody><tr><td></td><td><pre>{ "attendees": [{ "email": "test@gmail.com" }, { "email": "test@gmail2.com" }], }</pre></td></tr></tbody></table>	Model	Example Value		<pre>{ "attendees": [{ "email": "test@gmail.com" }, { "email": "test@gmail2.com" }], }</pre>
Model	Example Value							
	<pre>{ "attendees": [{ "email": "test@gmail.com" }, { "email": "test@gmail2.com" }], }</pre>							

Parameter content type:

Response Messages

HTTP Status Code	Reason	Response Model	Headers
404	Bad Request		

[Try it out!](#) [Hide Response](#)

Example data:

```
{  
  "attendees": [  
    {  
      "email": "v.kushnir@sentrrium.com"  
    }  
  ],  
}
```


Response Body:

```
{
  "_id": "5fa5229c565ed0340a69f25c",
  "event": {
    "attendees": [
      "v.kushnir@sentrinum.com"
    ],
    "cal":
"BEGIN:VCALENDAR\r\nVERSION:2.0\r\nPRODID:-//Ringplan//Ringplan
meeting//EN\r\nCALSCALE:GREGORIAN\r\nMETHOD:REQUEST\r\nBEGIN:VEVENT\r\nSUMMARY:
SUMMARY\r\nDTSTART;VALUE=DATE-TIME:20201107T121716Z\r\nDTEND;VALUE=DATE-TIME:20
201107T122516Z\r\nDTSTAMP;VALUE=DATE-TIME:20201106T101700Z\r\nUID:3h4h00ubs6uoj
jcqkje9tk749g@google.com\r\nSEQUENCE:0\r\nATTENDEE;CN=v.kushnir@sentrinum.com;CU
TYPE=INDIVIDUAL;PARTSTAT=NEEDS-ACTION\r\n
;ROLE=REQ-PARTICIPANT;RSVP=TRUE;X-NUM-GUESTS=0;MAILTO:v.kushnir@sentrinum.c\r\n
om\r\nATTENDEE;CN=kushnirvladislavv.testing@gmail.com;CUTYPE=INDIVIDUAL;PARTSTA
T\r\n
=NEEDS-ACTION;ROLE=REQ-PARTICIPANT;RSVP=TRUE;X-NUM-GUESTS=0;MAILTO:kushnir\r\n
vladislavv.testing@gmail.com\r\nCREATED;VALUE=DATE-TIME:20201106T101700Z\r\nDES
CRIPTION:\\n<b>Meeting link</b>\\nhttps://meet.dev.ringplan.com/130725076\r\n
\\n\\n<b>Join by phone</b>\\n(+1) 858-999-9915\\n\\n<b>Meeting
ID</b>\\n130-725-\\r\n 076#\\n\\n<b>Description</b>\\nTest Scheduled
Description\r\nLAST-MODIFIED;VALUE=DATE-TIME:20201106T101700Z\r\nLOCATION:\r\nO
RGANIZER;CN=kushnirvladislavv.testing@gmail.com;MAILTO:kushnirvladislavv.\r\n
testing@gmail.com\r\nSTATUS:CONFIRMED\r\nTRANSP:OPAQUE\r\nEND:VEVENT\r\nEND:VCA
LENDAR\r\n",
      "date_end": "2020-11-07T12:25:16+00:00",
      "date_start": "2020-11-07T12:17:16+00:00",
      "description": "Test Scheduled Description",
      "event_type": "scheduled",
      "google_event_id": "3h4h00ubs6uojjcqkje9tk749g",
      "link_on_meet": "https://meet.dev.ringplan.com/130725076",
      "meeting_id": "130725076",
      "summary": "Test Scheduled Summary",
      "text_for_event": "\n<b>Meeting
link</b>\\nhttps://meet.dev.ringplan.com/130725076\\n\\n<b>Join by phone</b>\\n(+1)
858-999-9915\\n\\n<b>Meeting ID</b>\\n130-725-076#\\n\\n<b>Description</b>\\nTest
Scheduled Description",
      "timeZone": "Etc/GMT+0",
      "title_meet": "Test Scheduled Title",
      "token_for_meet": {
        "exp": 1604838316,
        "iat": 1604665036,
        "link_on_meet_with_token":
"https://meet.dev.ringplan.com/130725076?jwt=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1Ni
J9.eyJhdWQiOiJ5OG56VDN1bjRMOG01TWV4bTZLYkRINjVDOFdremlMiLCJpc3MiOiJtZWV0aW5nLXNl
cnZpY2UiLCJzdWIiOiJtZWV0LmRldi5yaW5ncGxhbi5jb20iLCJyb29tIjoibWwNzI1MDE2IiwiaWF0Ijoi
wIjoxNjA0ODM4MzE2LCJqdGkiOiJiMTM0MTM0Zi1lZDUwLTQ3OWYtYmI5YS0zZTQ3MTJmMmMyYWEiLC

```

```

JpYXQiOjE2MDQ2NjUwMzYsIm1vZGVyYXRvciI6dHJlZX0.efQ1VWR3Po_57Ts668WFC14MJ2gQImS5e
51B_I2WojA",
    "room": "130725076",
    "token":
"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJhdWQiOiJ5OG56VDN1bjRMOG01TWV4bTZLYkRIN
jVD0FdrelMiLCJpc3MiOiJtZWV0aW5nLXNlcnZpY2UiLCJzdWIiOiJtZWV0LmRldi5yaW5ncGxhbi5j
b20iLCJyb29tIjoiMTMwNzI1Mdc2IiwiaXhwIjoxNjA0ODM4MzE2LCJqdGkiOiJiMTE4MTM0Zi1lZDU
wLTQ3OWYtYmI5YS0zZTQ5MTJmMmMyYWEiLCJpYXQiOjE2MDQ2NjUwMzYsIm1vZGVyYXRvciI6dHJlZX
0.efQ1VWR3Po_57Ts668WFC14MJ2gQImS5e51B_I2WojA"
  }
},
"is_private": false,
"organizer_email": "kushnirvladislavv.testing@gmail.com",
"organizer_id": "5f88615e8fbe62b6282f9237"
}

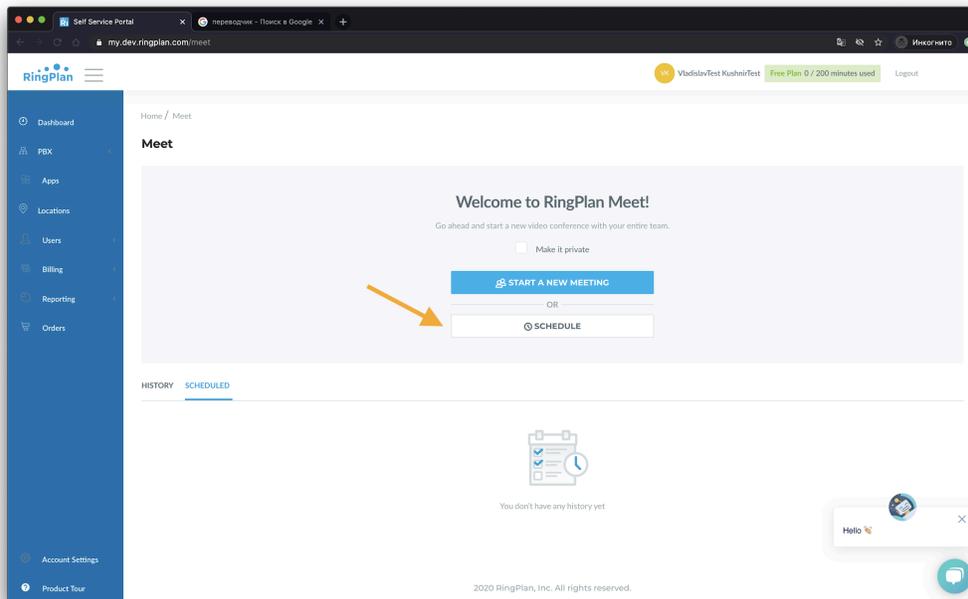
```

This event should appear in the organizer's google calendar and invitations should be sent to the participants

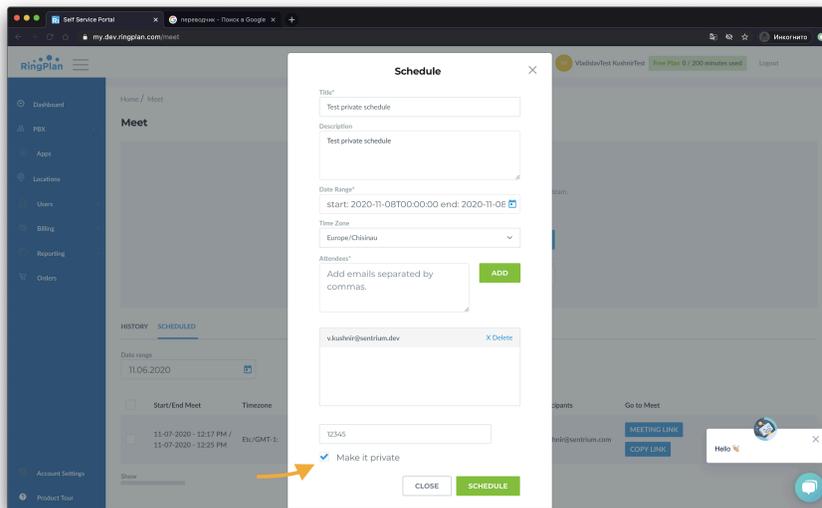
Create Schedule event (private)

SSP Backend example

Go to **Apps => Ringplan Meet**



Press button **“Scheduled”**
Fill in the required information



This event should appear in the organizer's google calendar and invitations should be sent to the participants

Swagger example

We used this endpoint for get history event

POST [/meet/calendar/event/scheduled](#)

POST [/meet/calendar/event/scheduled](#) Create a scheduled event

Implementation Notes
Create a scheduled appointment in google calendar

Response Class (Status 201) Successful

Model **Example Value**

```
{
  "_id": "5ed547b2ce4e5ac3876d7e26",
  "event": {
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
data	<pre>{ "attendees": [{ "email": "v.kushnir@sentrinum.com" }], "description": "Test Scheduled Description", "end_time": "2020-11-17T12:25:16.153321", "start_time": "2020-11-17T12:17:16.153321", "summary": "Test Scheduled Summary", "time_zone": "Etc/GMT+0", "title_meet": "Test Scheduled Title", "password": "1234567" }</pre>		body	Model Example Value

Parameter content type:

Response Messages

HTTP Status Code	Reason	Response Model	Headers
404	Bad Request		

[Try it out!](#) [Hide Response](#)

Response Body:

```
{
  "_id": "5fa524b9565ed0340a69f25d",
  "event": {
    "attendees": [
      "v.kushnir@sentrrium.com"
    ],
    "cal":
      "BEGIN:VCALENDAR\r\nVERSION:2.0\r\nPRODID:-//Ringplan//Ringplan
      meeting//EN\r\nCALSCALE:GREGORIAN\r\nMETHOD:REQUEST\r\nBEGIN:VEVENT\r\nSUM
      MMARY:SUMMARY\r\nDTSTART;VALUE=DATE-TIME:20201117T121716Z\r\nDTEND;VALUE=
      DATE-TIME:20201117T122516Z\r\nDTSTAMP;VALUE=DATE-TIME:20201106T102601Z\r\n
      nUID:q4iud8eohh4fk38eslp263h108@google.com\r\nSEQUENCE:0\r\nATTENDEE;CN=v
      .kushnir@sentrrium.com;CUTYPE=INDIVIDUAL;PARTSTAT=NEEDS-ACTION\r\n
      ;ROLE=REQ-PARTICIPANT;RSVP=TRUE;X-NUM-GUESTS=0:MAILTO:v.kushnir@sentrrium.
      c\r\n
      om\r\nATTENDEE;CN=kushnirvladislavv.testing@gmail.com;CUTYPE=INDIVIDUAL;P
      ARTSTAT\r\n
      =NEEDS-ACTION;ROLE=REQ-PARTICIPANT;RSVP=TRUE;X-NUM-GUESTS=0:MAILTO:kushni
      r\r\n
      vladislavv.testing@gmail.com\r\nCREATED;VALUE=DATE-TIME:20201106T102601Z\r
      \r\nDESCRIPTION:\n\n<b>Meeting
      link</b>\n\nhttps://meet.dev.ringplan.com/249636791\r\n\n\n\n<b>Join by
      phone</b>\n\n(+1) 858-999-9915\n\n\n\n<b>Meeting ID</b>\n\n249-636-\r\n
      791#\n\n\n\n<b>Password for meet</b>\n\n1234567\n\n<b>Description</b>\n\nTest
      Schedu\r\n
      led
      Description\r\n
      LAST-MODIFIED;VALUE=DATE-TIME:20201106T102601Z\r\n
      LOCATION
      :\r\n
      ORGANIZER;CN=kushnirvladislavv.testing@gmail.com:MAILTO:kushnirvladi
      slavv.\r\n
      testing@gmail.com\r\n
      STATUS:CONFIRMED\r\n
      TRANSP:OPAQUE\r\n
      END:VEVENT\r\n
      ND:VCALENDAR\r\n",
    "date_end": "2020-11-17T12:25:16+00:00",
    "date_start": "2020-11-17T12:17:16+00:00",
    "description": "Test Scheduled Description",
    "event_type": "scheduled",
    "google_event_id": "q4iud8eohh4fk38eslp263h108",
    "hash_pass":
      "8bb0cf6eb9b17d0f7d22b456f121257dc1254e1f01665370476383ea776df414",
    "link_on_meet": "https://meet.dev.ringplan.com/249636791",
    "meeting_id": "249636791",
    "mod_text_for_event": "\n\n<b>Meeting
    link</b>\n\nhttps://meet.dev.ringplan.com/249636791\n\n\n\n<b>Join by
    phone</b>\n\n(+1) 858-999-9915\n\n\n\n<b>Meeting
    ID</b>\n\n249-636-791#\n\n\n\n<b>Password for
    meet</b>\n\n$4DoxOWS7SLu\n\n\n\n<b>Description</b>\n\nTest Scheduled Description",
    "mod_token_for_meet": {
      "exp": 1605702316,
      "iat": 1605529036,
    }
  }
}
```

```

    "link_on_meet_with_token":
    "https://meet.dev.ringplan.com/249636791?jwt=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJhdWQiOiJ5OG56VDNlbnJRM0G01TWV4bTZLYkRINjVDOFdreIiMiLCJpc3MiOiJtZWV0aW5nLXNlcnZpY2UiLCJzdWIiOiJtZWV0LmRldi5yaW5ncGxhbi5jb20iLCJyb29tIjoimjQ5NjM2NzcxIiwiaXhwIjoxNjA1NzAyMzE2LCJqdGkiOiJiMTE4MTM0Zi1lZDUwLTQ3OWYtYmI5YS0zZTQ5MTJmMmMyYWEiLCJpYXQiojE2MDU1MjkwMzYsIm1vZGVyYXRvciI6dHJlZX0.R-rEtwN4PGrGGM8xT-E9nLnzCLqT2rDP6lPGkkgMVGU",
    "mod_password": "$4DoxOWS7SLu",
    "room": "249636791",
    "token":
    "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJhdWQiOiJ5OG56VDNlbnJRM0G01TWV4bTZLYkRINjVDOFdreIiMiLCJpc3MiOiJtZWV0aW5nLXNlcnZpY2UiLCJzdWIiOiJtZWV0LmRldi5yaW5ncGxhbi5jb20iLCJyb29tIjoimjQ5NjM2NzcxIiwiaXhwIjoxNjA1NzAyMzE2LCJqdGkiOiJiMTE4MTM0Zi1lZDUwLTQ3OWYtYmI5YS0zZTQ5MTJmMmMyYWEiLCJpYXQiojE2MDU1MjkwMzYsIm1vZGVyYXRvciI6dHJlZX0.R-rEtwN4PGrGGM8xT-E9nLnzCLqT2rDP6lPGkkgMVGU"
  },
  "password": "1234567",
  "summary": "Test Scheduled Summary",
  "text_for_event": "\n<b>Meeting
link</b>\nhttps://meet.dev.ringplan.com/249636791\n\n<b>Join by
phone</b>\n(+1) 858-999-9915\n\n<b>Meeting
ID</b>\n249-636-791#\n\n<b>Password for
meet</b>\n1234567\n\n<b>Description</b>\nTest Scheduled Description",
  "timeZone": "Etc/GMT+0",
  "title_meet": "Test Scheduled Title",
  "token_for_meet": {
    "exp": 1605702316,
    "iat": 1605529036,
    "link_on_meet_with_token":
    "https://meet.dev.ringplan.com/249636791?jwt=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJhdWQiOiJ5OG56VDNlbnJRM0G01TWV4bTZLYkRINjVDOFdreIiMiLCJpc3MiOiJtZWV0aW5nLXNlcnZpY2UiLCJzdWIiOiJtZWV0LmRldi5yaW5ncGxhbi5jb20iLCJyb29tIjoimjQ5NjM2NzcxIiwiaXhwIjoxNjA1NzAyMzE2LCJqdGkiOiJiMTE4MTM0Zi1lZDUwLTQ3OWYtYmI5YS0zZTQ5MTJmMmMyYWEiLCJpYXQiojE2MDU1MjkwMzYsIm1vZGVyYXRvciI6ZmFsc2V9._2s6bVNFq3HxcdRMpd-iYnGTloZceQAOWL7JED8LGxM",
    "room": "249636791",
    "token":
    "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJhdWQiOiJ5OG56VDNlbnJRM0G01TWV4bTZLYkRINjVDOFdreIiMiLCJpc3MiOiJtZWV0aW5nLXNlcnZpY2UiLCJzdWIiOiJtZWV0LmRldi5yaW5ncGxhbi5jb20iLCJyb29tIjoimjQ5NjM2NzcxIiwiaXhwIjoxNjA1NzAyMzE2LCJqdGkiOiJiMTE4MTM0Zi1lZDUwLTQ3OWYtYmI5YS0zZTQ5MTJmMmMyYWEiLCJpYXQiojE2MDU1MjkwMzYsIm1vZGVyYXRvciI6ZmFsc2V9._2s6bVNFq3HxcdRMpd-iYnGTloZceQAOWL7JED8LGxM"
  }
  },
  "is_private": true,
  "organizer_email": "kushnirvladislavv.testing@gmail.com",
  "organizer_id": "5f88615e8f8e62b6282f9237"
}

```

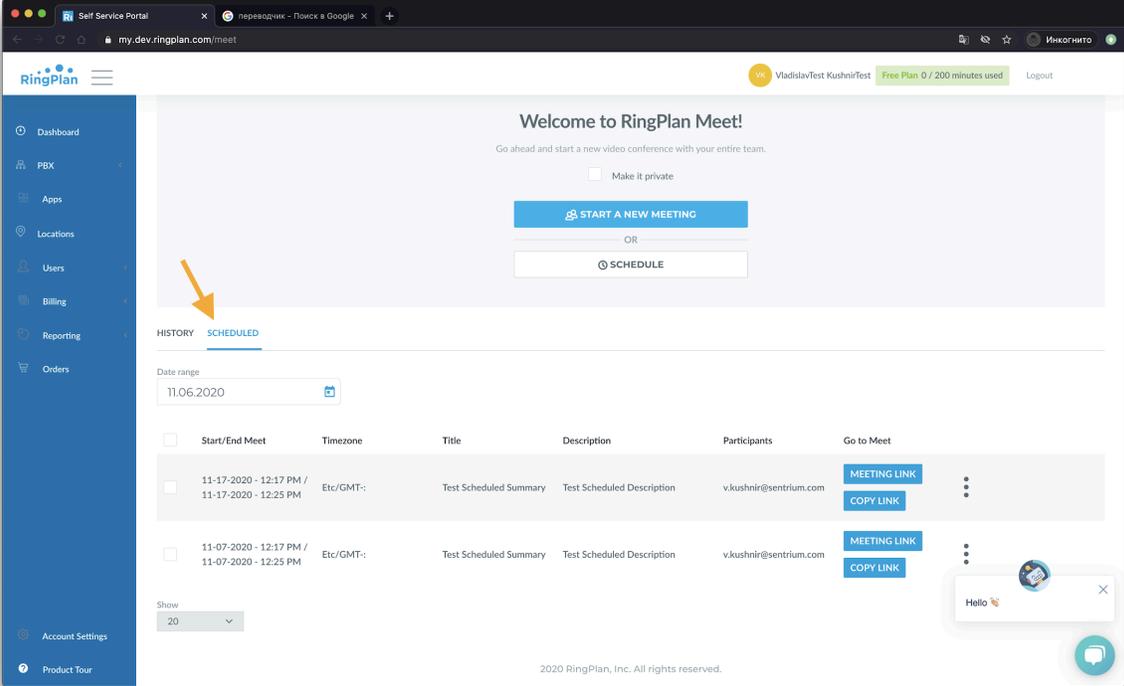
This event should appear in the organizer's google calendar and invitations should be sent to the participants

Get Schedule event (not private)

SSP Backend example

Go to **Apps** => **Ringplan Meet**

Press **“Scheduled”** (default)



The screenshot shows the RingPlan Meet web interface. The top navigation bar includes the RingPlan logo, a user profile for 'VladislavTest Kushnir/Test' with a 'Free Plan 0 / 200 minutes used' indicator, and a 'Logout' button. The main content area is titled 'Welcome to RingPlan Meet!' and contains a 'START A NEW MEETING' button and a 'SCHEDULE' button. An orange arrow points to the 'SCHEDULE' button. Below this, the 'SCHEDULED' tab is active, showing a table of scheduled meetings. The table has columns for 'Start/End Meet', 'Timezone', 'Title', 'Description', 'Participants', and 'Go to Meet'. Two meetings are listed, both with the title 'Test Scheduled Summary' and description 'Test Scheduled Description'. The 'Go to Meet' column contains 'MEETING LINK' and 'COPY LINK' buttons. A 'Date range' filter is set to '11.06.2020'. The footer of the page reads '© 2020 RingPlan, Inc. All rights reserved.'

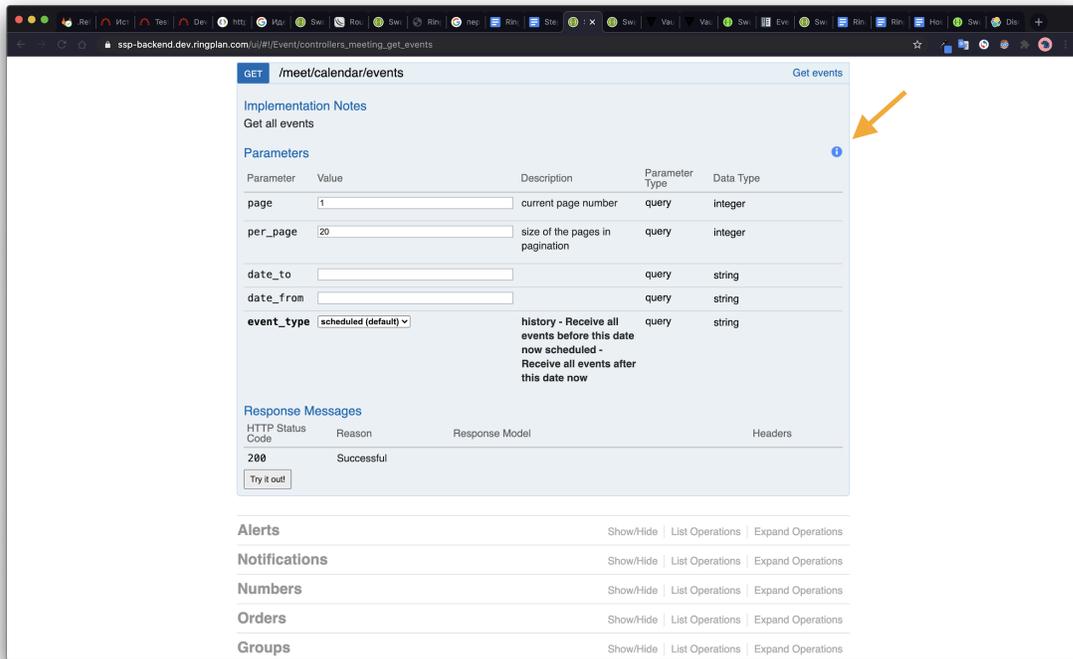
Start/End Meet	Timezone	Title	Description	Participants	Go to Meet
11-17-2020 - 12:17 PM / 11-17-2020 - 12:25 PM	Etc/GMT:-	Test Scheduled Summary	Test Scheduled Description	v.kushnir@sentrrium.com	MEETING LINK COPY LINK
11-07-2020 - 12:17 PM / 11-07-2020 - 12:25 PM	Etc/GMT:-	Test Scheduled Summary	Test Scheduled Description	v.kushnir@sentrrium.com	MEETING LINK COPY LINK

Swagger example

We used this endpoint for get scheduled event

GET [/meet/calendar/events](#)

We need to enter an authentication token in order to be able to send requests.
User_id on SSP-Backend is automatically substituted from the authentication token



The screenshot displays the Swagger UI for the endpoint `GET /meet/calendar/events`. The interface includes the following sections:

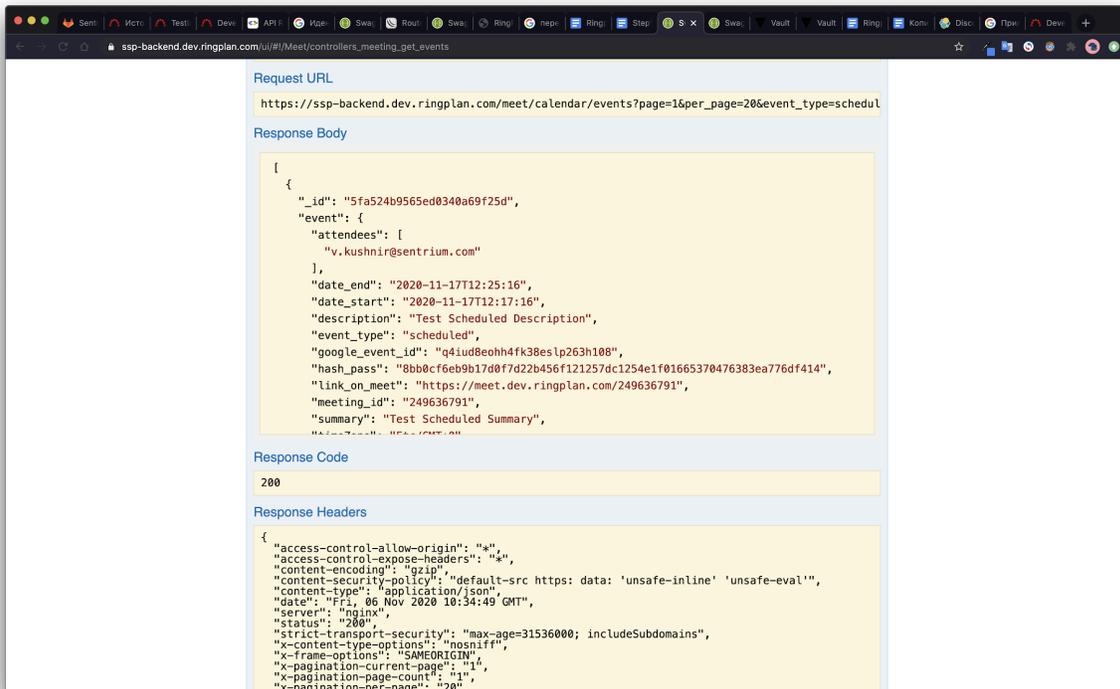
- Implementation Notes:** "Get all events"
- Parameters:** A table with columns for Parameter, Value, Description, Parameter Type, and Data Type.

Parameter	Value	Description	Parameter Type	Data Type
page	1	current page number	query	integer
per_page	20	size of the pages in pagination	query	integer
date_to			query	string
date_from			query	string
event_type	scheduled (default)	history - Receive all events before this date now scheduled - Receive all events after this date now	query	string
- Response Messages:** A table with columns for HTTP Status Code, Reason, Response Model, and Headers.

HTTP Status Code	Reason	Response Model	Headers
200	Successful		

An orange arrow points to a blue information icon in the top right corner of the main content area.

We must select the type **“scheduled”**



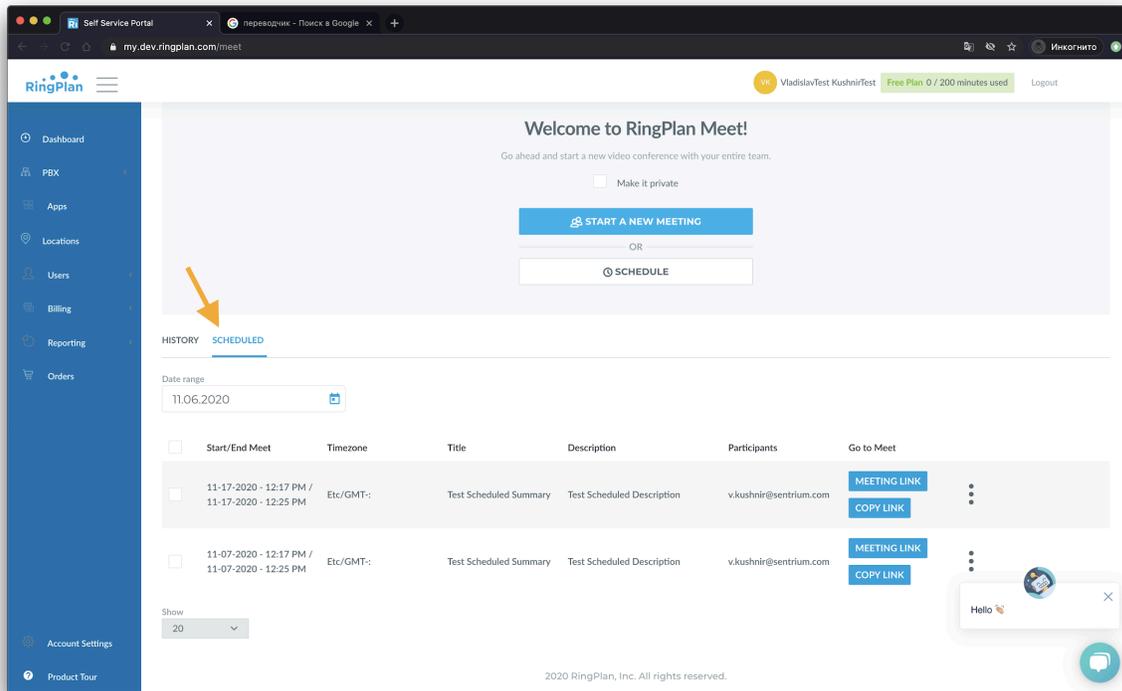
Response Body:

```
[
  {
    "_id": "5fa524b9565ed0340a69f25d",
    "event": {
      "attendees": [
        "v.kushnir@sentrinum.com"
      ],
      "date_end": "2020-11-17T12:25:16",
      "date_start": "2020-11-17T12:17:16",
      "description": "Test Scheduled Description",
      "event_type": "scheduled",
      "google_event_id": "q4iud8eohh4fk38eslp263h108",
      "hash_pass":
"8bb0cf6eb9b17d0f7d22b456f121257dc1254e1f01665370476383ea776df414",
      "link_on_meet": "https://meet.dev.ringplan.com/249636791",
      "meeting_id": "249636791",
      "summary": "Test Scheduled Summary",
      "timeZone": "Etc/GMT+0",
      "title_meet": "Test Scheduled Title"
    },
    "is_private": true,
    "organizer_email": "kushnirvladislavv.testing@gmail.com",
    "organizer_id": "5f88615e8fbe62b6282f9237"
  }
]
```

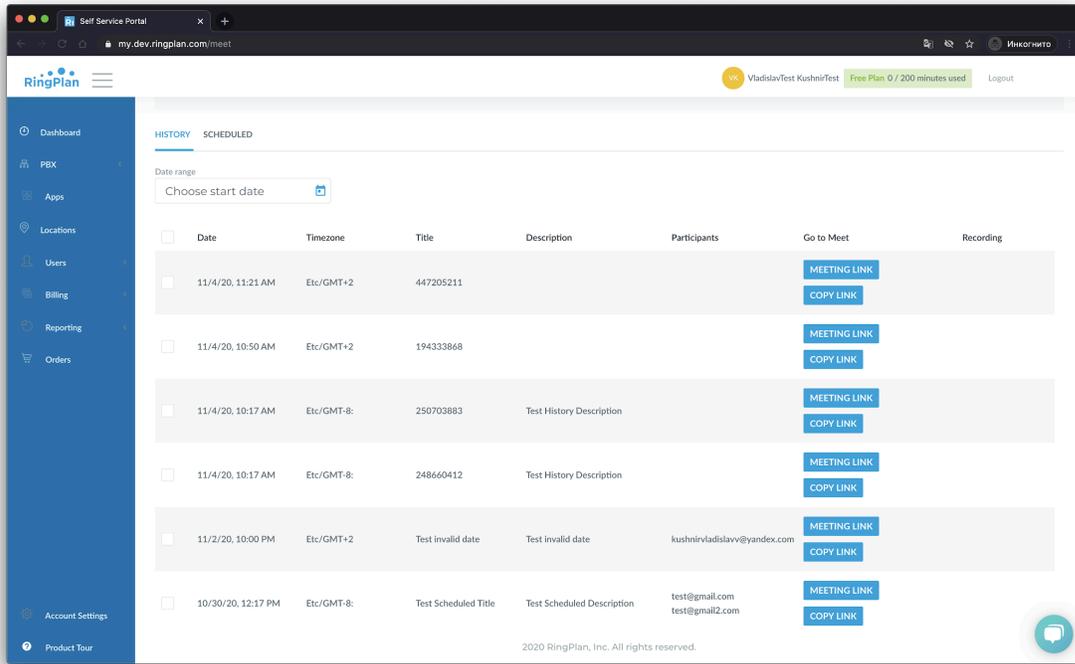
Get Schedule event (private)

SSP Backend example

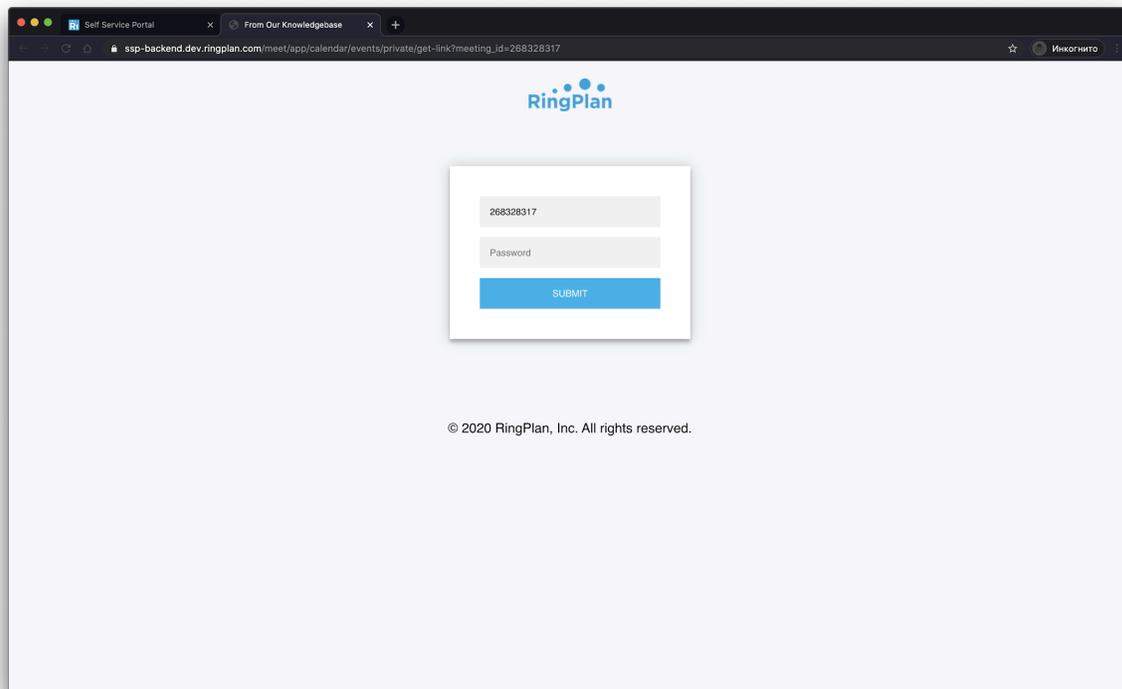
Go to **Apps => Ringplan Meet**



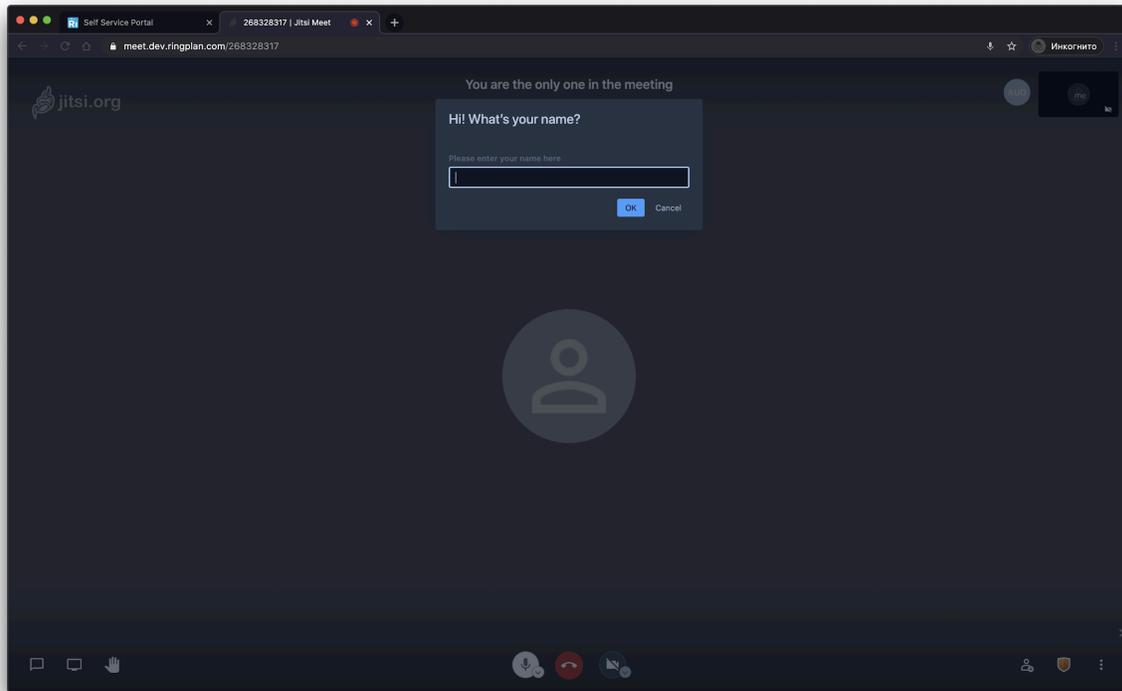
Press **“Scheduled”**
And see all scheduled event



If we click the **"Meeting Link"** button and this meeting is private, We will be taken to the password entry page.



After we enter the correct password, we will be redirected to the jitsi page



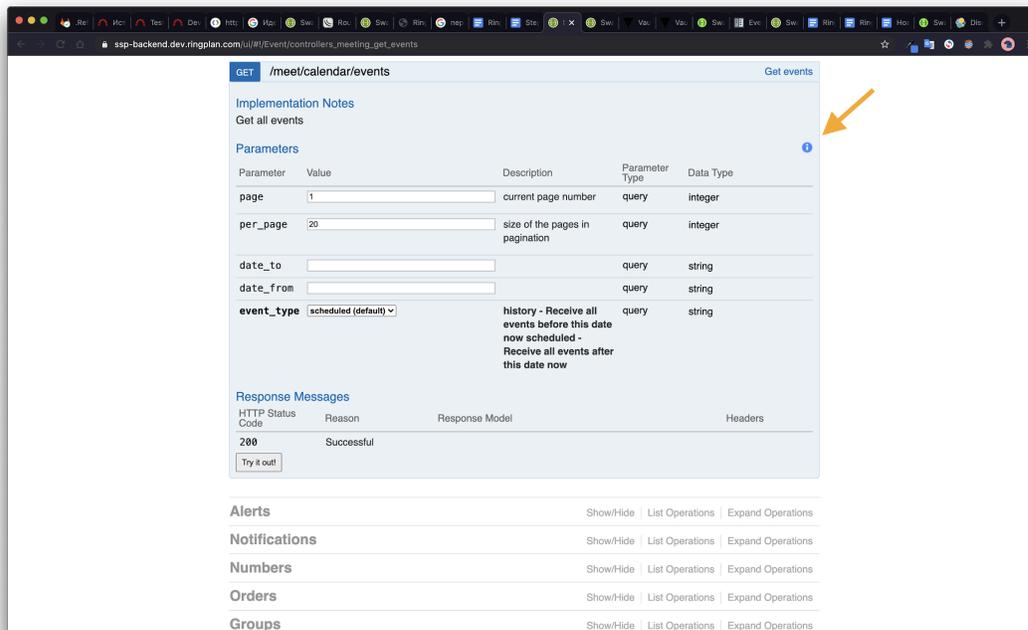
Swagger example

We used this endpoint for get scheduled event

GET [/meet/calendar/events](#)

We need to enter an authentication token in order to be able to send requests.

User_id on SSP-Backend is automatically substituted from the authentication token



The screenshot displays the Swagger UI for the endpoint `GET /meet/calendar/events`. The interface includes the following sections:

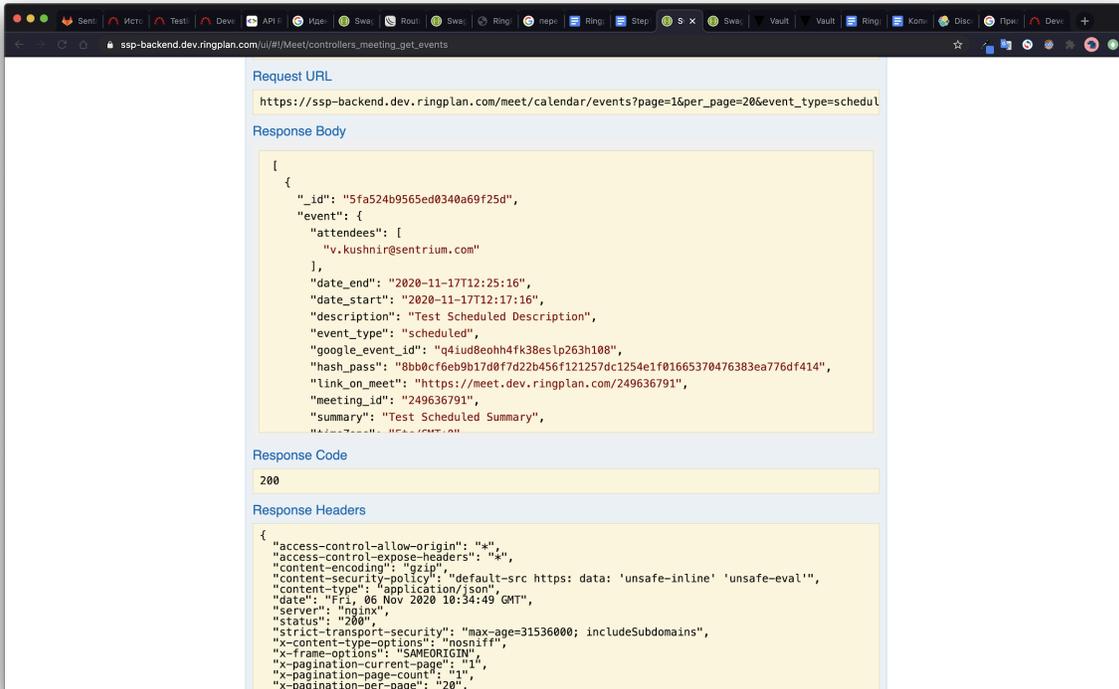
- Implementation Notes:** "Get all events"
- Parameters:** A table listing query parameters:

Parameter	Value	Description	Parameter Type	Data Type
page	1	current page number	query	integer
per_page	20	size of the pages in pagination	query	integer
date_to			query	string
date_from			query	string
event_type	scheduled (default)	history - Receive all events before this date now scheduled - Receive all events after this date now	query	string
- Response Messages:** A table showing the response status:

HTTP Status Code	Reason	Response Model	Headers
200	Successful		

A yellow arrow points to the information icon (i) in the top right corner of the endpoint details.

We must select the type **“scheduled”**



Response Body:

```
[
  {
    "_id": "5fa524b9565ed0340a69f25d",
    "event": {
      "attendees": [
        "v.kushnir@sentrinum.com"
      ],
      "date_end": "2020-11-17T12:25:16",
      "date_start": "2020-11-17T12:17:16",
      "description": "Test Scheduled Description",
      "event_type": "scheduled",
      "google_event_id": "q4iud8eohh4fk38eslp263h108",
      "hash_pass":
"8bb0cf6eb9b17d0f7d22b456f121257dc1254e1f01665370476383ea776df414",
      "link_on_meet": "https://meet.dev.ringplan.com/249636791",
      "meeting_id": "249636791",
      "summary": "Test Scheduled Summary",
      "timeZone": "Etc/GMT+0",
      "title_meet": "Test Scheduled Title"
    },
    "is_private": true,
    "organizer_email": "kushnirvladislavv.testing@gmail.com",
    "organizer_id": "5f88615e8fbe62b6282f9237"
  }
]
```

And after sending the request, we will receive a response with all history events

If the **"is_private"** field is **True** then this is a **private meeting**.

Questions?

Contact support@ringplan.com