



OPENCHS Rapid Pro Integrations

These endpoints are used to redirect a client conversation from the CHATBOT to the Helplines. The CHATBOT will notify the individual helpline based on location in cases of high risk, as determined by the CHATBOT. This will notify the Agents, who will then start a chat session with the client.

These are all test links that should only be used in testing environments. Individual country Endpoints will be made available during production.

To make the test easier, we've created a new server with the following links.

To connect to the test server, follow the steps below.

<https://35.79.109.15/helpline/>

username = test

password = p@ssw0rd

NB: Because the test environment is using a self-signed SSL certificate, accept SSL for the following two URLs as well.

<https://35.79.109.15:8089/ws>

<https://35.79.109.15:8384/ami/>

Once logged in, click "Join queue" – this will join the queue for both calls and chats.

Customer Requirement - Documentation

After interacting with the Chatbot, the Client wants to share some information with the Helplines.

```
provide API documentation for Case Creation.
```

{**BITZ** will provide an API endpoint for each site in production}

Weni will consume the API and confirm case creation

<https://documenter.getpostman.com/view/21578213/UzBpLRXa>

Weni CHATBOT starts a conversation:

- Weni invokes Rapidpro
- Rapidpro processes the defined workflow

BITZ will provide an Endpoint/Webhook based on WENI'S Data Definitions.

I.e

The conversation contains a new conversation_id (session_id) that uniquely identifies the conversation etc

#User Activity

The BITZ Helpline System 'hunts' for an available counselor

The counselor receives a chat notification on the Helpline system



If the counselor does not attend to the message within the required timeout, the notification is closed, and 'hunting' restarted

WENI configures RapidPRO workflows and defines and shares the data structure with the Bitz team, This will Trigger conversation with the Helplines.

The RapidPro workflow calls Webhook, which points to the BITZ API Gateway.
Rapidpro should retry in case of network Failures and/or timeout.

BITZ API Gateway should authenticate the message and ensure it is from the correct source and correct format

The Helpline System will notify the API Gateway that the message is delivered
API Gateway will notify Rapidpro of message delivery via an endpoint (to be provided by Weni in the desired format)

Rapidpro will notify Weni of message delivery
Conversation Acknowledgment Provides an endpoint for acknowledgment of delivered messages from Weni

Provides an endpoint for acknowledgment of delivered messages from Bitz

https://rapidpro.ilhasoft.mobi/api/v2/flow_starts

The acknowledgment endpoints ought to have: a conversation ID, from either the helpline system or Weni, and the message status

...

Test BOT has been provided as

http://t.me/mhpss_mvp_bot

Counselor reply:

- The Helpline System sends the message to the API Gateway
- The Gateway forwards the message to Rapidpro

WENI Provides an API endpoint To receive the message as configured on RapidPRO

#Notes

The message contains the conversation_id and pseudo-name of the current counselor handling the conversion.

Rapidpro forward the message to Weni

Weni should send back an acknowledgement that the message has been delivered.

client reply

- This is similar to 'start a conversion' but the conversation_id should already exist
- Each message should have a conversation_id, message_id, and client's pseudo-name (since the chats are anonymous)

Conversation End:

- The conversation can be terminated from either the Helpline System or Weni System
- Each message should have a conversation_id, and message_id, and the client's pseudo-WENI should clarify what we do when the actual conversation ends.

Error Handling:



HTTP status code

- HTTP status codes should be agreed upon and documented

POST Authentication-Token Request

```
https://openchs.bitz-itc.com/open/api/token/
```

The API uses token authentication. A username and password are provided to respective API consumers who then shall use them to acquire time-bound authorization tokens as follows: We shall use username: child and password: P@sswd for this documentation, however, the details do not exist and will not work on the provided links

```
POST token/ -d {"username":"child","password":"P@sswd"} -H {"content-type:application/json"}
```

HEADERS

Content-Type

application/json

BODY raw

```
{
  "username": "child@openchs.com",
  "password": "Op3nCh1ld"
}
```

Example Request

Authentication-Token Request

```
curl --location --request POST 'https://openchs.bitz-itc.com/open/api/token/' \
--header 'Content-Type: application/json' \
--data-raw '{
  "username": "child@openchs.com",
  "password": "Op3nCh1ld"
}'
```

GET List the Chats

```
https://openchs.bitz-itc.com/open/api/chat/
```



HEADERS

Content-Type

application/json

Authorization

Token e3beef7dbf5ee90d8d7dd335bb1eb2eb6e34d577

Example Request

List the Chats

```
curl --location --request GET 'https://openchs.bitz-itc.com/open/api/chat/' \
--header 'Content-Type: application/json' \
--header 'Authorization: Token e3beef7dbf5ee90d8d7dd335bb1eb2eb6e34d577'
```

POST Send a chat to OPENCHS

https://openchs.bitz-itc.com/open/api/chat/

This endpoint sends a message from the CHATBOT to the helping system.

HEADERS

Content-Type

application/json

Authorization

Token e3beef7dbf5ee90d8d7dd335bb1eb2eb6e34d577

BODY raw

```
{
  "chat_sender": 23343411,
  "chat_receiver": 233432311,
  "chat_message": "This is the message I am sending FROM PSTM",
  "chat_session": "e32434511",
  "chat_source": "WENI"
}
```

```
curl --location --request POST 'https://openchs.bitz-itc.com/open/api/chat/' \
--header 'Content-Type: application/json' \
--header 'Authorization: Token e3beef7dbf5ee90d8d7dd335bb1eb2eb6e34d577' \
--data-raw '{
  "chat_sender": 23343411,
  "chat_receiver": 233432311,
  "chat_message": "This is the message I am sending FROM PSTM",
  "chat_session": "e32434511",
  "chat_source": "PSTM"
}
```

View More

POST Chat Session Close

```
https://openchs.bitz-itc.com/open/api/chat/529150097d63/close/
```

HEADERS

Authorization

Token e3beef7dbf5ee90d8d7dd335bb1eb2eb6e34d577

BODY raw

```
{
  "chat_source": "WENI"
}
```

Example Request

Chat Session Close

```
curl --location --request POST 'https://openchs.bitz-itc.com/open/api/chat/529150097d63/close/' \
--header 'Authorization: Token e3beef7dbf5ee90d8d7dd335bb1eb2eb6e34d577' \
--data-raw '{
  "chat_source": "WENI"
}
```