**ChatRace Mobile App**

We already have a web version of our Inbox, and we highly recommend to the code related to the inbox web. A lot of JavaScript code can be reused if you are using React native.

It is required to check the code of **chatarea.php** to know how to render messages on the screen.

It is required to check the code on **inbox.js** to know how the WebSocket works when there is a new message.

Google Drive (code): <https://drive.google.com/drive/folders/13hJtyux2iGW47ltp4KAFwCYJYb9WD1VS?usp=sharing>

You will need to check our render messages

**API URL: {{subdomain}}/php/user**

**API documentation**: <https://documenter.getpostman.com/view/27874332/2s93zCZg9Q>

**Important notes:**

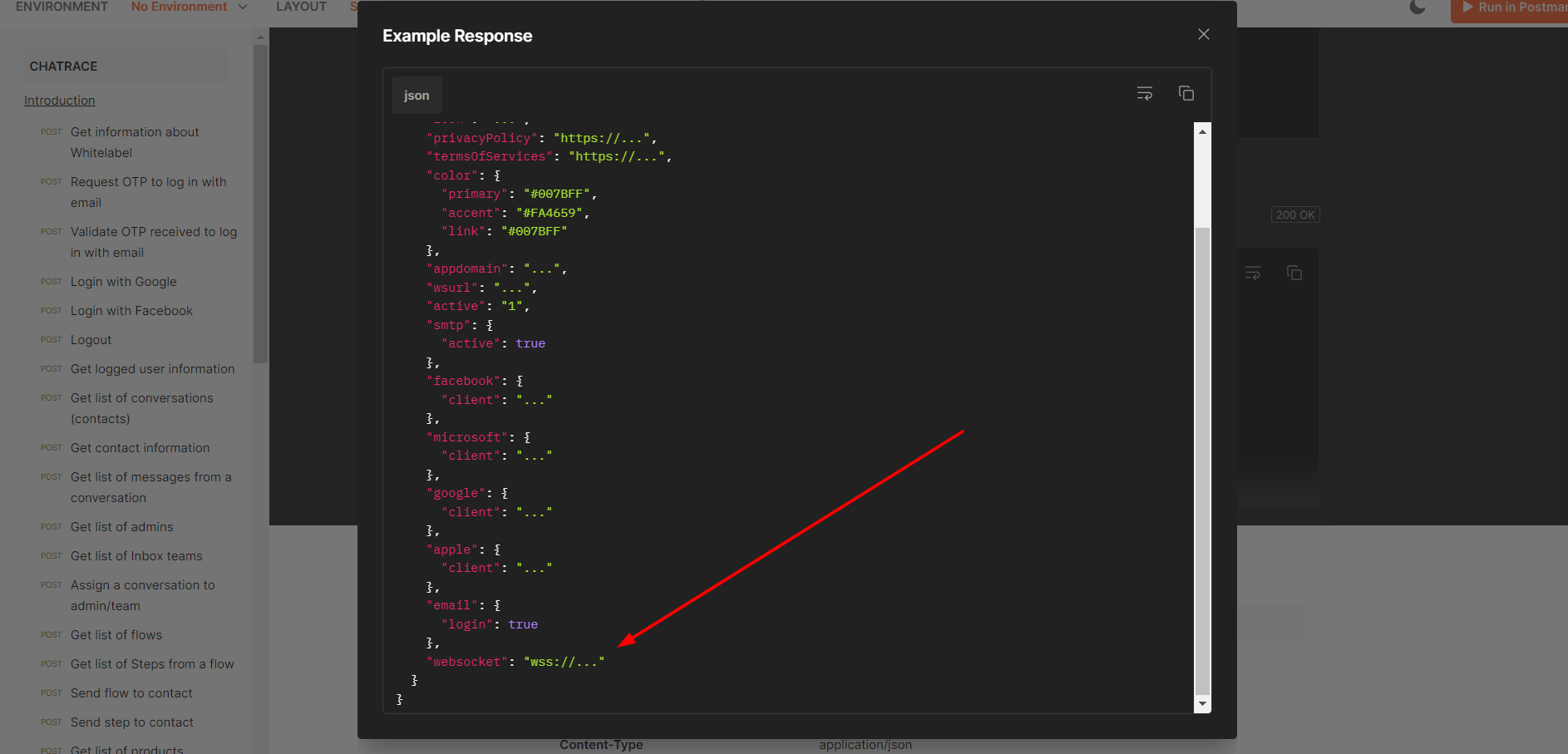
page\_id = account\_id (Business account ID)

ms\_id = contact\_id (User that sends a message to the bot)

fb\_id = admin\_id (Team member or an admin of a business account)  
  
A business account can have multiple contacts and team members. A team member can manage multiple business accounts.

**Websockt**

The WebSocket URL can be found on the Whitelabel information. The API to get the Whitelabel information is <https://documenter.getpostman.com/view/27874332/2s93zCZg9Q#6ba7310f-ae23-49a3-9738-32f83c1e3211>

****

After you connect to the WebSocket, you need to immediately **authenticate**.  
{

"action": "authenticate",

"data":{

"platform":"web|ios|android",

"account\_id": {{account\_id}},

"user\_id": {{logged\_user\_id}},

"token":"{{logged\_user\_token}}"

}

}

**Send a text message using WebSocket (send JSON object as a string)**

{

"action": 0,

"data":{

"platform":"web|ios|android",  
 “dir”:0,

"account\_id": {{account\_id}},

"contact\_id": {{contact\_id}},

"user\_id": {{logged\_user\_id}},

"token":"{{logged\_user\_token}}",

"fromInbox":true,

"channel": {{channel\_ID}},

"from": {{logged\_user\_id}},

"hash": {{contact\_hash}},

"timestamp":"{{unix\_timestamp\_milliseconds}}",

"message":[

{

"type": "text",

"text": "Your Message",

"dir": 0,

"channel": {{channel\_ID}},

"from": {{logged\_user\_id}},

"replyingTo": null | “message\_id"

}

]

}

}

**Send a image, audio, video,voice, and file message using WebSocket**

{

"action": 0,

"data":{

"platform":"web|ios|android",  
 “dir”:0,

"account\_id": {{account\_id}},

"contact\_id": {{contact\_id}},

"user\_id": {{logged\_user\_id}},

"token":"{{logged\_user\_token}}",

"fromInbox":true,

"channel": {{channel\_ID}},

"from": {{logged\_user\_id}},

"hash": {{contact\_hash}},

"timestamp":"{{unix\_timestamp\_milliseconds}}",

"message":[

{

"type":"file",

"dir": 0,

"channel": {{channel\_ID}},

"from": {{logged\_user\_id}},

"replyingTo": null | “message\_id",

"attachment":{

"type": "image|video|audio|file|voice",

"payload":{

"url":"{{uploaded\_file\_url}}"

}

},

"file":{

"url": "{{uploaded\_file\_url}}",

"file\_name":"{{file\_name}}",

"length":{{file\_length}},

"type":image|video|audio|file|voice"

}

}

]

}

}

**Push notifications**

Setup push notifications on IOS  
<https://medium.com/@ashoniaa/react-native-expo-push-notifications-with-fcm-a-step-by-step-guide-fa5cfc0372fd>

Notifications

{

"type": "message",

"account\_id": “...”,

"contact\_id": “...”,

"wt": “...”,

"subtype": "text",

"text": "..."

}