Open and close Xano channels | WeWeb Documentation

|  |  |
| --- | --- |
| URL source | <https://docs.weweb.io/websockets/xano-realtime/open-close-channel.html> |
| Date de scraping | 2025-04-08T15:56:25.568522+00:00 |

## Open and close a Xano channel ​

## Pre-requisites ​

To leverage Xano realtime actions in WeWeb, you first need to enable and connect Xano realtime.  
If you haven't already, please do so here .

## Why open a channel matters ​

If you want users of your app to be able to listen and react to events in a realtime channel, you first need to make them open the channel.

## Open channel ​

The Xano Realtime | Open channel action allows you to establish a connection to a Xano channel.  
In the example below, you can see we created a workflow that opens a channel called room1 :

### Images

<https://docs.weweb.io/assets/xano-realtime-open-channel.Bksm8Ekk.png>

## Configuration options ​

When you setup an Open channel action, you have the option to enable or disable three Xano realtime features:  
Listen to Presence : This feature keeps track of who enters and leaves the channel, helping you see who is currently active or when someone has left.  
Get History on Join : When you join a chatroom, this option lets you view previous messages from other users. To leverage this option, you first need to enable Message History in the settings of your channel in Xano:  
  
Queue Offline Actions : When a user performs an action, such as sending a message while disconnected, these actions are saved and automatically sent when they reconnect, ensuring nothing is missed.

### Images

<https://docs.weweb.io/assets/xano-realtime-message-history-setting.CH66ArFq.png>

## Three things to keep in mind ​

1. You can name the channel anything you like.  
Just remember that, later on, when you start sending events to that channel and listening to events from that channel, you'll need to reference the same channel name.  
2. You need to decide when this workflow is triggered.  
Examples include, but are not limited to:  
On click , to subscribe to a chat channel when a user clicks on a support icon. On page load , to subscribe to a mousemove channel that updates cursor positions in a live collaboration tool. On app load , to subscribe to an update channel that notifies users of new product updates.  
3. The Open channel action does one thing and one thing only : it opens a two way road between the client (the user's browser tab) and the Xano realtime channel.  
Sending and receiving events are a seperate topic.

## Next steps ​

Once we have established a connection between WeWeb and a Xano channel, we can:  
Send a message to that channel Listen and react to realtime events from that channel .

## Close channel ​

The Close channel action ends an active realtime channel connection, halting all communications and data exchanges through that channel.  
Taking the example of a support chat, you could trigger the Close channel action when the user clicks on a button to close the chat window:

### Images

<https://docs.weweb.io/assets/xano-realtime-close-disconnected.DuIPKJLo.png>