LioranBoard Chatbot

Version 2.1 (6 April 2022)

by Melonax

Introduction

LioranBoard Chatbot was made for people who want to use a local chatbot client but do not want to use any of the alternatives, or people who already use LioranBoard 2 and prefer to have everything in one place. It currently features:

- Chat commands (without needing extra buttons)
- Counters
- Giveaways
- Timed messages
- Queues
- Quotes
- Welcome events

Luckily for you, the LioranBoard Chatbot works out of the box. You don't have to install any extensions or edit anything in the deck. Any alteration is done via the stream deck, .csv files, or chat commands.

You may of course adjust the commands in the other buttons however you see fit, but this is at your own risk. Stuff can break. You have been warned.

Feel free to move around or resize the buttons on the deck to your preference.

You can use the chatbot via the LioranBoard 2 Stream Deck, but you can also just use chat commands if you so prefer.

- Melo

PS: After April 10 2022, I will no longer be able to fix bugs or assist with any issues, as I will not have regular access to a PC. I will try to get this chatbot as well-functioning as possible before that time.

NOTE: LioranBoard Chatbot requires LioranBoard version 2.07.6 or higher!

Table of Contents

Click an entry to jump to that page!

Setup

Attaching your bot account to LioranBoard (optional)

Want to use a different username to post messages to chat? Look no further! You will need an extra Twitch account for this, so if you don't have one yet, create it.

1. Log in to your bot account in your browser. TIP: I usually just leave my channel logged in regularly & open an incognito tab (ctrl+shift+N) to log into my bot account.
2. At the bottom of the receiver, click 'Twitch Connections'. A window will pop up.
3. Click Copy URL, and open this URL in the browser where you logged in your bot account. After authorizing, the bot account should show up in LioranBoard.
4. Select 'Join chat under this name' below your bot account
5. Select 'Join Channel' below your stream account
6. Restart LioranBoard

Installing the chatbot

Installing the chatbot itself is simply done by copying the deck into LioranBoard. Follow the following steps to do so:

1. Launch the LioranBoard 2 Receiver (v2.07.6 or higher)
2. Click 'Transmitter' in the left side menu
3. Select 'Install an Extension'
4. Select LioranBoard Chatbot.lb2 and click Open

Done! A deck should have been added and folder called 'chatbot' should have been created in your LioranBoard folder, containing cbsettings.ini (with default settings) and commands.csv (empty apart from header & default rows).

Importing LioranBoard 1 Chatbot (chatbot.ini)

If you used LioranBoard 1 Chatbot before moving over to LioranBoard 2, you can easily import all your commands, quotes, etc. via the following steps:

1. Place your old chatbot.ini file in the newly created chatbot folder.
2. Enable all features in the LB2 chatbot (even if you didn't use them before) and make sure no .csv files are currently opened in Excel.
3. Click or tap the small grey 'IMPORT' button near the top left of the chatbot deck

You will have to set the settings from your LB1 setup button manually via the toggles & gear icons for each feature.

Getting started

Enabling or disabling features

LioranBoard Chatbot currently has 7 optional features: Chat commands, Counters, Giveaway, Queue, Quotes, Timed messages and Welcome events. If you need more information on what those features do, each feature has its own chapter in this document.

All optional features apart from Commands are disabled by default. To enable them, open the chatbot deck on your Stream Deck and use the toggle buttons on the right.

Changing settings

Most features come with some settings you can adjust (if that feature is currently enabled). You can press the gear icon next to a feature's toggle button to view and/or adjust its current settings. This process will happen via pop-ups, which show up on the monitor your receiver is currently active on. Available settings are explained in each feature's chapter.

Adjusting .csv files

For all features except giveaways & queues, a .csv file will be created in the chatbot folder when you first enable said feature. These files holds all custom data.

A .csv file (Comma Separated File) is a file that stores data in a grid-like form. When opening it via Excel, the values will automatically be placed in individual boxes. You may also use Notepad, but you'll have to use your imagination to 'see' the boxes. Either way, adjusting the file shouldn't be too difficult. Just add in new rows for each separate message, command or counter and fill out the boxes. What exactly should be added is explained in each feature's chapter.

After adjusting your file, save it (Excel may nag you another time when closing after saving, you can ignore that), and either restart LioranBoard **or** click the small back 'RELOAD CSV's' button near the top left of the chatbot deck.

NOTE: Make sure the **,DefaulT, row stays at the bottom!**

NOTE: Make sure there are **no empty rows** between the headers & default values!

NOTE: **Close Excel** before using the chatbot! Excel hijacks all writing permissions to the file, so if you leave the file open the chatbot will not be able to adjust it until you close it again.

Default chat commands

LioranBoard Chatbot has several default commands. These commands have been added via the LioranBoard trigger window and will not show up in the !commands list.
Refer to the FAQ-section of this document in case you want to change any of these commands, as changing some of them may break stuff if you do it incorrectly.

Command	Required user level	Output
!commands	-	Lists all commands You may add sub/vip/mod/broadcaster to see the commands for that user level (!commands sub)
!addcommand	Moderator	Adds a command
!delcommand	Moderator	Deletes a command
!disablecommand	Moderator	Temporarily disables a command
!enablecommand	Moderator	Re-enables a command after disabling it
!editcommand	Moderator	Edits a command
!editpermission	Moderator	Edits the required user level for a command
!editcooldown	Moderator	Edits the global cooldown for a command
!editusercooldown	Moderator	Edits the user cooldown for a command
!count+	-	Adds 1 to the current counter
!count-	-	Deducts 1 from the current counter
!count reset	Moderator	Resets the current counter to 0
!give	-	User joins the giveaway
!opengiveaway	Broadcaster	(Re-)Opens giveaway
!closegiveaway	Broadcaster	Closes giveaway
!winner	Broadcaster	Selects random giveaway winner
!join	-	User joins the queue
!next	Broadcaster	Advance to next person in queue
!random	Broadcaster	Choose random person from queue
!clear	Broadcaster	Clears the queue
!quote	-	Outputs a random quote You may add a number to request a specific quote
!lastquote	-	Outputs the newest quote
!addquote	-	Adds a new quote
!delquote	Moderator	Deletes a quote

Custom chat commands

LioranBoard has built in functionality that supports custom chat commands, but this system doesn't require a separate button for each command. There's perks to either system:

	This system	Separate buttons (see page 17/18)
Adding & modifying commands via chat	✓	✗
Automatic list of commands	✓	✗
Add parameters easily	✓	✗
Require certain user levels	✓	✓
Cooldowns & usercooldowns	✓	✓
Sound effects	✓	✓
User-specific commands	✗	✓
Mid-line commands	✗	✓

Modification commands

You may use these commands to alter your custom chat commands. Additional commands are available to change permissions or cooldowns and to disable or enable commands.

!addcommand

Adds a command. Requires the command name & output.

!addcommand [commandname] [command output]
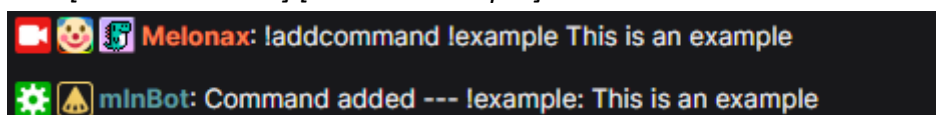
!editcommand

Edits a command. Requires the command name & output.

!editcommand [command name] [command output]

!delcommand

Deletes a command. Requires the command name.

!delcommand [command name]

Parameters

You can add parameters to your chat commands. This means adding a short phrase that will be replaced by something that will be retrieved from the message or elsewhere. You can add as many parameters as you like. The following parameters are currently available (examples on next page):

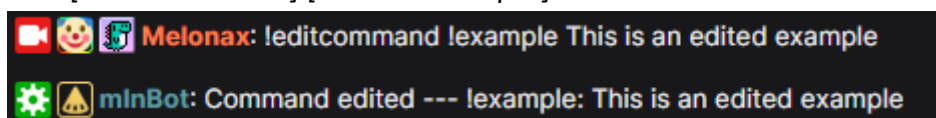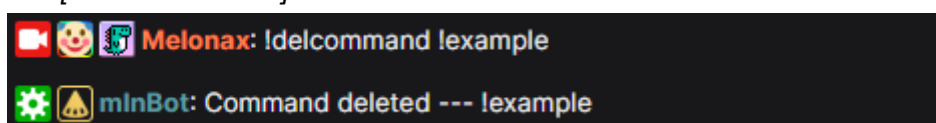| Parameter	Will be replaced by
{channel}	Your channel name, will be lowercase
{user}	The chatter who sent the message
{wildcard}	The word(s) after the command. Will be replaced by {user} if left empty by the chatter
rand{min-max} rand{max}	A random number between min and max (e.g. rand{1-6}) If no min value is listed, 0 is assumed (i.e. rand{100} = rand{0-100})
[[api URL]]	The api result
count{counter}	The resulting number after adding 1 to the mentioned counter. If counter doesn't exist yet, it is created. Counter name is case sensitive. Can be used in conjunction with other parameters.
sound{filename}	Will not show in chat. Plays sound effect of that name. File must be an .ogg file and be added to the chatbot folder. Do not include the extension, i.e. if your file is called yay.ogg, use sound{yay}
volume{percentage}	Will not show in chat. Can only be used in a command that also has a sound effect added. Adjusts the volume of the sound effect. Cannot be higher than 100.

Try out some of the following examples:
!addcommand !hug {user} hugs {wildcard}

!addcommand !death {channel} died! Death count for current game: count{[[https://decapi.me/twitch/game/{channel}]]}

!addcommand !odds The current chance of winning is rand{100}%

!addcommand !uptime Stream uptime: [[https://decapi.me/twitch/uptime/{channel}]]

!addcommand !followage {wildcard} followed on [[https://decapi.me/twitch/followed?channel={channel}&user={wildcard}]]

User levels

The LioranBoard Chatbot allows you to add chat commands at five different user levels. By default, new commands will be available to everyone. If you wish to adjust this, use the !editpermission command.

!editpermission
Edits the required user level for a specific command. Requires the command name & level.
!editpermission [command name] [level]

#	User level
0	everyone
1	subscriber
2	vip
3	moderator
4	broadcaster

NOTE: Chatters with a higher user level will be able to use all lower level commands

Global & user cooldowns

You may add global & user cooldowns for each individual command.

- A global cooldown means nobody can use the command for that many seconds after it has been used.
- A user cooldown means that same person can't use the command again for that many seconds after using it, but other people still can.

By default, new commands will have no cooldowns. If you wish to adjust this, use the !editcooldown or !editusercooldown command.

!editcooldown
Edits the global cooldown for a specific command. Requires the command name & seconds.
!editpermission [command name] [seconds]

!editusercooldown
Edits the user cooldown for a specific command. Requires the command name & seconds.
!editpermission [command name] [seconds]

Enabling/disabling commands

If you want to keep a command without anyone being able to use it, you can disable it temporarily until you need it again. Use the following commands to do so:

!disablecommand
Temporarily disables a command without actually deleting it. Requires the command name.
!disablecommand [command name]

!enablecommand
Re-enables a command after it has been disabled. Requires the command name.
!enablecommand [command name]

Adjusting commands.csv

If you have a list elsewhere and want to copy it over, or just prefer to add commands via a file instead of via chat, you can edit commands.csv directly. Please read up on the general information on adjusting files (page 4) before changing anything.

Boxes cannot be empty. Refer to the following table to know what to enter:

Header	Value
command	Your command name. Cannot contain spaces.
response	What the command should output to chat.
permission	The lowest user level that can use this command. Must be one of the following (lowercase): • everyone • subscriber • vip • moderator • broadcaster
cooldown	The global cooldown in seconds. Use 0 if no cooldown.
usercooldown	The user cooldown in seconds. Use 0 if no cooldown.
enabled	Whether the command is currently enabled. Must be one of the following (lowercase): • true • false

Counters

The counter feature can be used to keep track of one or multiple counters, for example to count deaths in several games.

Setup

By default, two counters called 'counter A' and 'counter B' will be added to your chatbot, but you can rename these and add as many as you like. You can also manually update the number a counter is at, if needed.

Adding or adjusting a counter can be done by editing counters.csv. Please read up on the general information on adjusting files (page 4) before changing anything.

Boxes cannot be empty. Refer to the following table to know what to enter:

Header	Value
name	The name of your counter. May contain spaces.
count	The current number the counter is at.

Settings

The only setting for the counter feature is whichever counter is the currently active counter. Changing this setting via the gear icon is mostly useful if you have a lot of counters. If you only have a few, you can just cycle through the list by tapping the big yellow button.

Usage

The yellow buttons on the deck take care of the counters. The big button shows the active counter and can be used to switch to the next counter in the list. There is one button to increase the counter, one to decrease it and one to reset it back to 0. You can tap them on your stream deck app, or use their respective chat commands:

Command	Required user level	Output
!count+	-	Adds 1 to the current counter
!count-	-	Deducts 1 from the current counter
!count reset	Moderator	Resets the current counter to 0

Additionally, you can use the count{counter} parameter in any chat command (see previous chapter). Using that, you can use multiple counters at the same time, if needed.

If you don't have easy access to the stream deck, you may want to add a hotkey to the counters as well. To do this, open the chatbot deck in your receiver, right click the button you want to add a hotkey to and click 'Edit Triggers'. Click the + icon and select 'Hotkey' at the top. Choose your hotkey and save the triggers and deck.

Giveaway

Using the giveaway feature, you can give away prizes to your community. They can join the giveaway via chat, and you can use the stream deck or chat commands to draw a winner.

Settings

There's two settings you can change for giveaways: the sub multiplier and (dis)allowing re-entry. You can change these settings by clicking the gear next to the giveaway toggle via your stream deck app. Pop-ups will guide you through the process.

The **sub multiplier** is a number that sets the chance a sub has of winning the giveaway, compared to a regular chatter. If you set it to 1.5, a sub has 1.5 times the chance of winning. 1 means there is no difference between subs and anyone else.

Allowing **re-entry** means someone can join the next giveaway(s) even if they already won one during this session. Disallowing re-entry means they cannot join again.

NOTE: Regardless of the re-entry setting, people can only win the **same** giveaway once.

Usage

The purple buttons on the deck take care of the giveaway. The big button shows the entrant number and last winner and can be used to proceed to the next stage of the giveaway. You may also use the medium buttons or following chat commands:

Command	Required user level	Output
!give	-	User joins the giveaway
!opengiveaway	Broadcaster	(Re-)Opens giveaway
!closegiveaway	Broadcaster	Closes giveaway (may be skipped)
!winner	Broadcaster	Selects random giveaway winner

Giveaway process
1. Tap the big button for the **first time** or type **!opengiveaway**
   The giveaway will open. This means people can start typing !give to join. Everyone can only enter once per giveaway.
2. Tap the button a **second time** or type **!closegiveaway**
   The giveaway will close. This means people can no longer enter the giveaway, but no winner has been chosen yet. You can do this to add suspense, but it can be skipped.
3. Tap the button a **third+ time** or type **!winner**
   A winner will be drawn. If someone wins, their name will be removed from the giveaway so they can't win another time. You can pick as many winners as you like, until there are no more entrants left.

Queue

Using the giveaway feature, you can allow your viewers to queue up for something. They can join the queue via chat, and you can use the stream deck or chat commands to proceed to the next person on the list (or choose a random one).

Settings

The only setting for the queue feature is how many people can join the queue. You can view and edit the limit via the gear icon next to the queue toggle. When adjusting, 0 means the queue is unlimited.

Usage

The dark blue buttons on the deck take care of the queue. The big button can be tapped to advance to the next person in the queue. Tapping the random button pulls a random user from the queue, and the clear button will clear the queue. You may also use the following commands:

Command	Required user level	Output
!join	-	User joins the queue
!next	Broadcaster	Advance to next person in queue
!random	Broadcaster	Choose random person from queue
!clear	Broadcaster	Clears the queue

For your convenience, the big button will display the next 5 users in the queue, as well as the total size of the queue.

Quotes

Using the quotes feature, chatters can add and request quotes via chat. A quote will automatically be assigned a number, and will automatically include the current Twitch category and/or date if you so prefer.

Settings

You can adjust whether you want to include the Twitch category and/or date behind a quote, as well as the format in which they show up (e.g. in [square brackets]). Click the gear icon next to the quotes toggle and follow the directions in the pop-ups.

The date can show either numbers or words, using the following parameters:

Parameter	Will be replaced by
day	24th
d	24
month	March
m	3
year	2022
y	2022

Usage

The green buttons on the deck take care of the quotes. You can tap the 'send quote' button to output a random quote to chat, but your chat may get more use out of the following commands:

Command	Required user level	Output
!quote	-	Outputs a random quote You may add a number to request a specific quote
!lastquote	-	Outputs the newest quote
!addquote	-	Adds a new quote
!delquote	Moderator	Deletes a quote

Adjusting quotes.csv

If you have a list elsewhere and want to copy it over, or just prefer to add quotes via a file instead of via chat, you can edit quotes.csv directly. Please read up on the general information on adjusting files (page 4) before changing anything.

Boxes cannot be empty. Refer to the following table to know what to enter:

Header	Value
number	The quote number. Must start with # (e.g. #1)
response	What the quote should output to chat.

Timed messages

The timer feature will automatically send the next message on the list to chat at the interval you specified, but only if more than the minimum chat lines have been posted since the previous message. It will loop back to the start once it reaches the end of the list.

Settings

You may adjust the timer interval and minimum required chat lines via the gear icon next to the timer toggle. Follow the directions in the popups.

The table below can be helpful to figure out how many seconds you need to enter:

Post a message every…	Frequency in seconds
1 minute	60
5 minutes	300
10 minutes	600
15 minutes	900
30 minutes	1800
1 hour	3600

Adjusting timer.csv

To add messages to your timer, you will have to edit timer.csv (should be in your chatbot folder after you've enabled the setting at least once). Please read up on the general information on adjusting files (page 4) before changing anything.

Boxes cannot be empty. Refer to the following table to know what to enter:

Header	Value
number	The timer number (timers will occur in this order). Must start with # (e.g. #1).
response	The timer message.
enabled	Whether this specific timer message is currently enabled. If disabled, the timer will skip this message and continue to the next number. Must be one of the following (lowercase): • true • false

Welcome events

The welcome events will automatically send a message to chat and/or play a sound when a specific user speaks in chat for the first time during that (LioranBoard) session. You can use this to welcome your regular viewers!

Settings

If you want to use a default chat message or default sound effect, you can add/adjust these by clicking the gear icon next to the welcome toggle. Default messages/sounds only play in case there is no personal welcome message for that user in welcome.csv.

Adding personal welcome messages

To add personal messages to your welcome events, you will have to edit welcome.csv (should be in your chatbot folder after you've enabled the setting at least once). Please read up on the general information on adjusting files (page 4) before changing anything.

You can leave boxes empty if you only want to use a message or only want a sound effect. Refer to the following table to know what to enter:

Header	Value
name	The chatters username. Must be lowercase!
response	The welcome message (if you want to send one to chat)
sound	The welcome sound effect (if you want to play one). File must be an .ogg file and be added to the chatbot folder. Do not include the extension, i.e. if your file is called welcomesound.ogg, enter welcomesound
volume	Volume of this specific sound effect. Number between 0 and 100. Will default to 100 if left empty.
enabled	Whether the welcome event for this user is currently enabled. Must be one of the following (lowercase): • true • false

NOTE: If the user already talked in your chat during this session, you will have to restart or reset LioranBoard to test ☺

Separate LB buttons

If you're looking for something a little more advanced, this is where the real power of LioranBoard comes in. Add a new deck and start adding buttons! Just click an empty space and select 'Create Button'. Make sure to always press Save on both the edit window AND the deck view so you're back at the main menu before testing, otherwise your changes haven't been saved.

I've outlined two of the more common chatbot-features below. If you're looking for something else, make sure to check out the LioranBoard documentation at https://lioranboard.ca. Feel free to ask away in the LioranBoard Discord server as well (link on website)!

Mid-line triggers

If you want a chat command that works independently of where a word appears in the sentence (for example, whenever someone says 'purple'), you need to add a trigger with some wildcards on either side (asterisks *). Follow these steps:

1. Add a button and your desired effects
2. Right click the button and select 'Edit triggers'
3. Click the + sign
4. Choose 'Twitch Chat'
5. In the message box, enter `*purple*`
6. Select 'Allow empty Wildcard'
7. Click Save & Save
8. Success!

NOTE: This method ignores the possibility of a term being part of another word. E.g. if you enter *ok*, things like 'p**ok**emon' and '**ok**lahoma' will trigger the button.
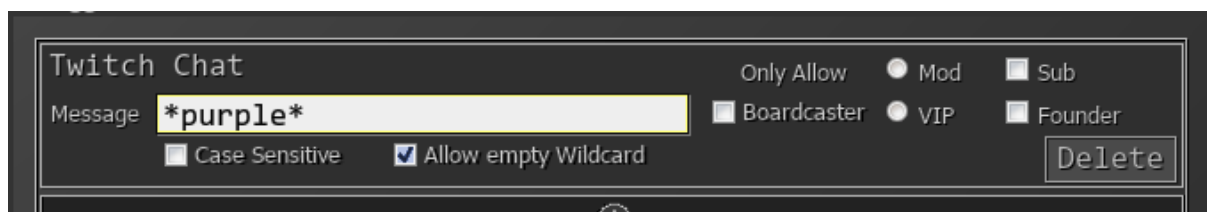
User-specific commands

To make a command user specific, you will have to follow these steps:

1. Add a button and your desired effects/triggers
2. Right click the button and select 'Edit Commands'
3. Click the + sign
4. Add a Trigger Pull Data command
5. In the left box, give the variable a name (e.g. chat_username)
6. For the right box, click the dropdown icon on the right and select either User Name or Display Name. Not all triggers provide both, so check the list.

NOTE: User Name = **all lowercase**, Display Name = **InCluding CapitAls**

7. Add an Exit statement
8. Set it to 'Exit if chat_username != "username"'
9. If you already had some commands on this button, select both commands you just added and drag them to the top of the button.
10. Click Save & Save
11. Success!
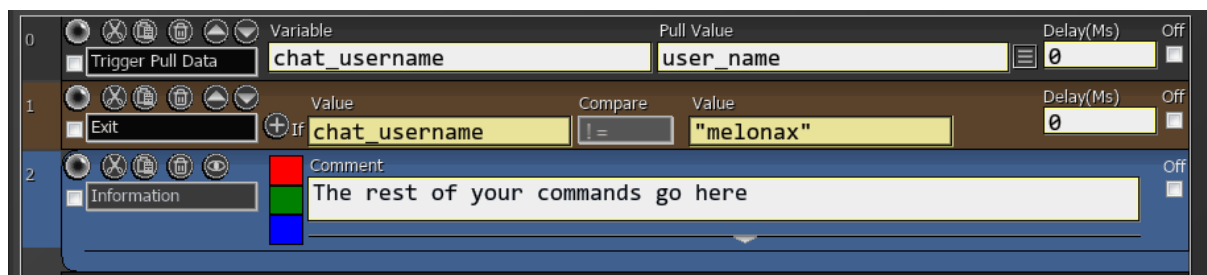
FAQ

Where can I find the chatbot folder?

The chatbot is located inside your LioranBoard folder. It should be added as soon as you add the LioranBoard Chatbot deck (containing cbsettings.ini and commands.csv).

If you added the deck but nothing's there, try restarting LioranBoard and see if it shows up after. If not, something may be up with your writing permissions. Run LioranBoard as admin, make sure the firewall isn't blocking it and make sure LioranBoard isn't in some cloud service or in Program files. If you can't figure it out, ask in the LioranBoard Discord server.

Can I delete buttons I don't need?

No, I would advise against that. Removing buttons can cause the chatbot to stop functioning or even crash LioranBoard. Try if you want to. I'm not taking responsibility.

Can I rearrange/resize buttons on the chatbot deck?

Yes, no problem. Use whatever arrangement you prefer. Make buttons teeny if you don't need them. Make them large if you can't properly read or tap them. Move them around however suits your mood.

Can I copy a button to another deck?

This depends on which button you want to copy. For some buttons, you need to make sure you adjust the button ID to match with what it was on the chatbot deck. This is mainly the case for buttons that change appearance (if you want it to keep doing that). Duplicates are not allowed, so rename the old button to something else first. This applies to the following buttons:

Button(s)	Button ID(s)
All toggle buttons	CBcommands, CBcounters, CBgiveaway, CBqueue, CBquotes, CBtimer, CBwelcome
The large counter button that shows the current counter & count	CBcounter
The large giveaway button	CBgive
The large queue button	CBq
The timer button that shows the time left (?:?? by default)	CBtime
The large welcome button	CBsayhi

How do I adjust a message that's being sent to chat?

Some buttons will automatically send a predefined message to chat. If you want to change that message or get rid of it, find the appropriate button, right click it and select 'Edit Commands'. Most buttons will have multiple (nested) blue comment blocks explaining what the button does. You can expand these by clicking the (closed) eye icon next to the bin.

Find the 'Twitch: Send Chat Message' command you want to change. Some of the buttons are a lot more complicated than others and may be a bit harder to weed through, but all the messages are in there somewhere! The command you're looking for will look somewhat like this:
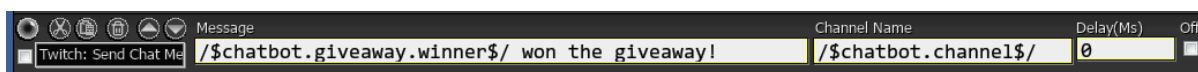
If you wish to use a variable in the message, enclose it in /$ $/ like in the example above. If you don't want to send a message at all you can disable the command using the 'Off' checkmark on the right.

How do I adjust the default chat commands?

The default chat commands and their user levels are coded into the Triggers window for each button. To get to that window, find the appropriate button in the LioranBoard receiver and select 'Edit Triggers'. Buttons that have triggers have a gun icon in the top left.

Editing user levels

You may edit the user levels however you wish, by (un)checking the box for each level. This should not cause any issues.
Be aware that the requirements stack, so if you would select both 'Mod' and 'Sub' on the same trigger, the chatter has to be both modded **AND** subbed to use the command. Add extra triggers if you want to specify more than one user level.

Editing command names

For most buttons (except the ones outlined below), you can change the command name without any issues. Make sure the command name still includes any asterisks (*) wherever they were added by default.

If you see a command has been added **multiple times in different ways**, make sure to stick to that format and only change the name (for all instances)!

Editing !addcommand, !delcommand, !enablecommand and/or !disablecommand

All these commands work via the same button. The type of modification is handled by wildcard 0 (the first *), so the triggers for all of these are **!*command *** and **!*command * ***.

If you just want to change **the word 'command'** (e.g. !add<u>cmd</u>), you can do that without any issues, as long as you stick to the format.

If you wish to change **the format** (e.g. <u>!command add</u>), move the first * to wherever it is appropriate. Make sure the rest of the * are still correct. If you want multiple formats to work, add more triggers.

If you wish to adjust the words that can be used as **the wildcard** (e.g. !<u>new</u>command), you need to edit the commands, not the trigger. Open the commands, find the green block and open it using the closed eye icon next to the bin. Inside, you will find a 'Switch Statement' with four 'Case Statements'. You can use the + sign on each case to add extra terms for that specific modification, or adjust the existing terms. Some examples are given (e.g. !newcommand does the same thing as !addcommand by default). Do not change anything else.

Editing !editcommand, !editpermission, !editcooldown and/or !editusercooldown

All these commands work via the same button. The type of modification is handled by wildcard 0 (the first *), so the trigger for all of these is **!edit* * ***.

If you just want to change **the word 'edit'** (e.g. !<u>update</u>permission), you can do that without any issues, as long as you stick to the format.

If you wish to change **the format** (e.g. <u>!permission edit</u>), move the first * to wherever it is appropriate. Make sure the rest of the * are still correct. If you want multiple formats to work, add more triggers.

If you wish to adjust the words that can be used as **the wildcard** (e.g. !edit<u>level</u>), you need to edit the commands, not the trigger. Open the commands, find the green block and open it using the closed eye icon next to the bin. Inside, you will find a 'Switch Statement' with four 'Case Statements'. You can use the + sign on each case to add extra terms for that specific modification, or adjust the existing terms. Some examples are given (e.g. !editlevel does the same thing as !editpermission by default). Do not change anything else.