

最終更新日：2025-05-26 (v3.3.1)

Twitch EventSub Response Bot (twitch-eventsub-response-py)

Twitch で配信中にレイドを受けたときに、それに応答して自動で「 /shoutout レイド元のユーザー名 」Twitch 公式チャットコマンドの実行や、チャット欄に指定したメッセージを表示してくれる、ボットアプリです。

百聞は一見に如かず、 **本ボットの動作例** をご覧ください。

v3.0.0から チャットメッセージの **棒読みちゃん** への受け渡し機能 も搭載しています。 **Twitch以外のサイトに同時配信していなければ**、 **わんコメ**・ **マルチコメントビューア**・ **Tubeyomi**などを併用せずに本ボットと **棒読みちゃん** だけで、チャットに送信されたメッセージを読み上げできます。設定方法は下記の **棒読みちゃん連携機能の設定** を参照ください。

v2.0から **チャットメッセージの翻訳機能** も搭載しています。 **チャット翻訳ちゃん** を併用せずに本ボットだけで、チャットに送信されたメッセージを翻訳できます。設定方法は下記の **チャット翻訳機能の設定** を参照ください。

※ これらの機能は、個別にオン・オフができます。

目次

- [背景説明](#)
- [本ボットのメイン機能](#)
- [動作環境](#)
- [ダウンロードとインストールの方法](#)
- [事前設定](#)
 - [ボットとして運用するユーザーにモデレーター権限を付与](#)
 - [ボットとして運用するユーザーのユーザーアクセストークン文字列の取得](#)
 - [トークン取得ウェブサービスが **トークン文字列を悪用しないと信じる** 場合](#)
 - [トークン取得ウェブサービスがトークン文字列を悪用しないと信じない場合](#)
 - [config.json5 ファイルへの設定の記述](#)
 - [必須の設定](#)
 - [イベントに自動で応答する機能の設定](#)
 - [チャット翻訳機能の設定](#)
 - [棒読みちゃん連携機能の設定](#)
- [実行](#)
 - [起動](#)
 - [動作中であるかの確認](#)
 - [停止](#)
 - [再起動](#)
 - [異常中断](#)
- [アンインストール方法](#)
- [今後の展開](#)
- [バージョン履歴](#)

- 参考資料

背景説明

Twitch配信のチャット欄に指定したメッセージを自動で表示してくれるボットサービスには [Nightbot](#) などがあり、例えば「！」で始まる『**ユーザーチャットコマンド（以下：ユーザーコマンド）**』を定義し、ある程度条件を指定して実行させることができます。 [Streamlabs](#) ないし [StreamElements](#) といったサービスと組み合わせれば、他配信者からのレイドや視聴者によるフォローなどのイベントが発生すると自動で応答してユーザーコマンドを実行させることもできます。

しかし、少なくとも [Nightbot](#) に関しては、「/」で始まる『**Twitch公式チャットコマンド（以下：公式コマンド）**』のうち、以下のものしか実行させることができないようです。

公式コマンド	実行内容
/me メッセージ	「メッセージ」をイタリックで表示させる（日本語は非対応）
/announce メッセージ	「お知らせ（改行）メッセージ」を表示させる

詳しくは未調査ですが、[Streamlabs](#) や [StreamElements](#) も公式コマンドを実行させることができないと推測しています。

本ボットのメイン機能

このような背景を踏まえて、**イベントに自動で応答し、かつ、上記以外の公式コマンドを実行させることのできるボット**を開発しました。本稿更新時点でサポートしている機能は以下です。

応答タイミング	コマンド	実行内容
レイドを受け たとき	/shoutout レイド元のユーザー名	レイド元のユーザーのチャンネルを応援し、フォローボタン付きでチャット内で紹介する
レイドを受け たとき	（任意のメッセージ）	「（任意のメッセージ）」を表示させる（これを利用して、 ユーザーコマンドも実行可能 ）

そのほか、以下の機能も備えています。

- チャットメッセージの翻訳機能（v.2.0以降）
- チャットメッセージの [棒読みちゃん](#) への受け渡し機能（v3.0.0以降）

なお、各機能は、メイン機能も含めて、個別にオン・オフが設定できます。

動作環境

- .exeファイル版：たぶん、本稿更新時点でサポートされている Windows（64bit版）で動作
- スクリプト版：たぶん、Python 3.10 以降のPythonインタプリタで動作

ダウンロードとインストールの方法

- .exeファイル版：右上にある [Releases](#) → 最新版の [twitch-eventsub-response-py-vX.Y.Z.zip](#) ファイルをダウンロードして展開

- `X.Y.Z` の部分は数字
- スクリプト版：右のReleasesからソースコードをダウンロードするなり本リポジトリをクローンするなりし、必要な外部パッケージをインストールしたうえで、Pythonインタプリタを使って実行
 - 必要な外部パッケージは `./Venvs/requirements.txt` に記載
 - ただし `DeepL Translate` は、ソースコードをダウンロードし、`deepl` となっている全ての箇所を `deepltranslate` に変更したうえでインストールすることが必要
 - `DeepL Python Library` とパッケージ名が競合するのを回避するため

.exeファイル版は、ダウンロードするのに使用するブラウザーによってはウイルスの疑いありと判定され、ダウンロードが妨げられる可能性があります。その場合は、（もちろん本ボットはウイルスではないので）疑いを解除してダウンロードできるようにしてください。

事前設定

本ボットを起動する前にやるべきことは最大で3つあります。

ボットとして運用するユーザーにモデレーター権限を付与

ボットとして運用するユーザーを決めてください。

- 配信で使っているユーザーをボットとしても運用する場合：権限を付与する必要はなし
 - すでにモデレーター以上の権限を持っているため
- ボットとして運用するユーザーを別に用意する場合：そのユーザーにモデレーターの権限を与えること
 - **セキュリティの観点から、こちらをお勧め**

すでに `チャット翻訳ちゃん` などでユーザーをボットとして使用している場合は、同じユーザーを本ボットに使用しても、お互い正常に動作するようです。

ボットとして運用するユーザーのユーザーアクセストークン文字列の取得

本ボットが正常に動作するには、`チャット翻訳ちゃん` などと同様に、「ユーザーアクセストークン」文字列というものをTwitchから取得して使用しなければなりません。**トークン文字列は、ユーザーによって、そして、本ボットを含むTwitch関係の外部アプリやサービスが何を行う権限を要求するかによって、異なるものになります**。本ボットが要求する権限は以下のとおりです。

公式コマンド・メッセージ	実行に必要な権限名	権限の意味
<code>/shoutout</code>	<code>moderator:manage:shoutouts</code>	<code>/shoutout</code> 公式コマンドを実行できる
<code>/color</code>	<code>user:manage:chat_color</code>	チャット欄で表示されるボットユーザー名の色を設定できる
(任意のメッセージ) <code>/me</code>	<code>chat:edit</code>	チャット欄に投稿できる
(全般)	<code>chat:read</code>	チャット欄に接続できる

さて、トークン文字列を取得するのに外部サービスを利用すると、そのサービスはトークン文字列を知り得てしまうので、要求して承認された権限を悪用できてしまいます。なので、ここから先は **セキュリティ意識に応じてトークン文字列の取得方法を選んでください**。

トークン取得ウェブサービスがトークン文字列を悪用しないと信じる 場合

「Twitch Chat OAuth Password Generator (Twitch Chat OAuth Token Generator)」は、[チャット翻訳ちゃん](#)の公式ページにて、トークン文字列を取得する方法として紹介されているウェブサービスです。ただし、本ボットは [チャット翻訳ちゃん](#) とは違う権限を要求するので、同じ方法ではなく、以下の手順でトークン文字列を取得してください。

まず、ブラウザを開いて、**ボットとして運用するユーザーでTwitchにログインした状態にしてください**。

- ボットとして運用するユーザーを別に用意する場合は、配信で使っているユーザーでいったんログアウトしてボットとして運用するユーザーでログインし直るか、ブラウザのシークレットモードなどと呼ばれる機能を使ってボットとして運用するユーザーでログインしてください
 - Chrome : 「シークレット ウィンドウ」
 - Edge : 「InPrivate ウィンドウ」
 - Firefox : 「新しいプライベートウィンドウ」

そして、以下のURLをコピーし、ブラウザにペーストして、URLにアクセスしてください。

```
https://id.twitch.tv/oauth2/authorize?
response_type=token&client_id=q6batx0epp608isickayubi39itsckt&redirect_uri=https://
/twitchapps.com/tmi/&scope=moderator:manage:shoutouts+user:manage:chat_color+chat:
edit+chat:read
```

「Twitch Chat OAuth Token Generatorアカウントにアクセスしようとしています」というページが表示されるので、「許可」を選んでください。すると、画面が遷移し、「**oauth:9y0urb0tuser0authacceesst0ken9**」などという文字列が表示されます。このうち **oauth: より右の(おそらく30桁前後となる)文字列** (この例の場合、**9y0urb0tuser0authacceesst0ken9**) がトークン文字列になります。

トークン取得ウェブサービスがトークン文字列を悪用しないと信じない場合

自前でトークン文字列を取得してください。

- 例えば [Twitch APIに必要なOAuth認証のアクセストークンを取得しよう](#) に書かれている方法
 - 「トークンを取得してみる」
 - 「1. The OAuth implicit code flow」

どちらの場合でも、トークン文字列を取得できたら、ブラウザは閉じてかまいませんが、その前にトークン文字列を一時的にどこかにコピペなどして忘れないようにしてください。

config.json5 ファイルへの設定の記述

まず、**config.json5** ファイルを、テキストエディタ (メモ帳など) で開いてください。**config.json5** は、ダウンロード時点では以下の内容になっています。

```
// Twitch EventSub Response Bot - Config (v3.0.0--)
{
  // ■ メッセージ送信先となるチャンネルに関する設定たち
  "messageChannel": {
    // チャンネル配信者のユーザー名(チャンネルURLの末尾)
    // (* 全て英小文字でも、英大文字と英小文字が混在していても、どちらでも可)
    "broadcasterUserName": "yourchannelname",
  },
  //
  // ■ メッセージ送信を行うボットに関する設定たち
  "bot": {
    // ボットとして運用するユーザーのOAuthアクセストークン
    // (* 使う機能が要求する権限をボットとなるユーザーが持っていること)
    // (* 使う機能が要求する権限をトークンが持っていること)
    "oAuthAccessToken": "9y0urb0tuser0authaccesst0ken9",
    //
    // チャンネルで表示されるボットの名前の色
    // (* トークンが "user:manage:chat_color" 権限を持っていること)
    // "blue", "blue_violet", "cadet_blue", "chocolate", "coral",
    // "dodger_blue", "firebrick", "golden_rod", "green", "hot_pink",
    // "orange_red", "red", "sea_green", "spring_green", "yellow_green",
    // "#RRGGBB" (* Turboユーザーのみ設定可), "doNotChange" (* 色を変えない),
    "nameColor": "blue",
  },
  //
  // ■ イベントたちに対する応答たちに関する設定
  "responses": {
    // レイド
    "/raid": [
      // [
      //   送信前の待機時間(秒), 公式コマンド・メッセージ (* ユーザーコマンドを含む),
      //   (* 必要あれば追加情報1, 追加情報2, ...)
      // ] の組たち
      // (* 上から順に1つずつ実行)
      //
      // コマンドやメッセージの中で置換される文字列たち
      // {{raidBroadcasterUserName}} -> レイド元のユーザー名(チャンネルURLの末尾)
      //
      // メッセージ (* ユーザーコマンドを含む) の例
      [ 5, "!raided {{raidBroadcasterUserName}}", ],
      //
      // 公式コマンドの例
      // /shoutout
      //   (* ボットとなるユーザーが モデレーター 以上であること)
      //   (* トークンが "moderator:manage:shoutouts" 権限を持っていること)
      [10, "/shoutout", ],
      //
      // (* 公式コマンド・メッセージを実行しない場合は、
      //   該当する [ ] の行を削除するか、行の頭に // を挿入(コメントアウト))
      // [10, "Sample message", ],
      //
      // (* ほかのコマンドは、要望があれば追加対応するかも)
    ],
  },
}
```

```

//
// (* ほかのイベントは、要望があれば追加対応するかも)
},
//
// ■ メッセージたちに対する翻訳に関する設定
"translation": {
  // 使用する翻訳サービスたちと優先使用順位
  // (* 各メッセージについて、翻訳できるまで、上に設定したサービスから順に使用)
  "servicesWithKeyOrURL": [
    // DeepL翻訳で、認証キーを使用しない場合 (* 不具合がなければ変更不要)
    // エラーが出て翻訳されない場合が多いため、コメントアウト中
    // ["deeplTranslate", "https://www2.deepl.com/jsonrpc", ],
    //
    // Google翻訳で、Google アカウント にひも付いた設定を必要としない場合
    // "translate.google.????/"
    // (* いずれかの国のURLを設定するが、不具合がなければ変更不要)
    ["googleTrans", "translate.google.co.jp", ],
    //
    // (* サービスを使用しない場合は、
    //   該当する [ ] の行を削除するか、行の頭に // を挿入(コメントアウト))
    //
    // DeepL翻訳で、認証キーを使用する場合
    // ["deeplKey", "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx:fx", ],
    //
    // Google翻訳で、Google Apps Script (GAS) を使用する場合
    // "https://script.google.com/macros/s/????/exec" (* GASのURL)
    // ["googleGAS", "https://script.google.com/macros/s/????/exec", ],
  ],
  //
  // 翻訳元言語の判定に使用するサービス (* 不具合がなければ変更不要)
  "fromLanguageDetection": "translate.google.co.jp",
  //
  // 翻訳しないメッセージたち
  "messagesToIgnore": {
    // 送信ユーザー名たち
    // (* 英大文字と英小文字が混在していても可)
    // (* ボットとして運用するユーザーについては、ここに記載がない場合、
    //   手動で投稿したメッセージたちは翻訳を実行)
    "senderUserNames": ["nightbot", "", ],
    //
    // 翻訳元言語たち
    // (* ノルウェー語、中国語は、
    //   DeepL翻訳とGoogle翻訳とで略称が異なるため、
    //   これらの言語を設定する場合は両方の略称を併記するのがよい)
    // DeepL翻訳(認証キー使用, 不使用), Google翻訳すべてで利用可能な言語たち
    // "BG" (Bulgarian), "CS" (Czech), "DA" (Danish),
    (中略)
    // "SV" (Swedish),
    // DeepL翻訳(認証キー使用), Google翻訳で利用可能な言語たち
    // "ID" (Indonesian), "KO" (Korean), "TR" (Turkish),
    // "UK" (Ukrainian),
    // DeepL翻訳(認証キー使用, 不使用)で利用可能な言語たち
    // "ZH" (Chinese)
    // (* Google翻訳では "zh-cn" または "zh-tw" ),

```



```

// DeepL翻訳(認証キー使用)でのみ利用可能な言語たち
//      "NB" (Norwegian),
//      (* Google翻訳では "no" )
// Google翻訳でのみ利用可能な言語たち
//      "af" (afrikaans), "sq" (albanian), "am" (amharic),
(中略)
//      "yi" (yiddish), "yo" (yoruba), "zu" (zulu),
//      (* ほかにもあるが、本ボットでは未対応)
"fromLanguages": [ "", ],
//
// ユーザーコマンドの接頭辞たち (* "<ter>_" は記載がなくても翻訳を不実行)
"userCommandPrefixes": [ "!", "", ],
//
// メッセージ内に含まれる文字列たち
"stringsInMessage": [ "http", "", ],
},
//
// 翻訳先言語(たち)
// (* 英語, ノルウェー語, ポルトガル語, 中国語は、
//   DeepL翻訳(認証キー使用, 不使用)とGoogle翻訳とで略称が異なるため、
//   これらの言語を設定する場合は両方の略称を併記するのがよい)
// DeepL翻訳(認証キー使用, 不使用), Google翻訳すべてで利用可能な言語たち
//      "BG" (Bulgarian), "CS" (Czech), "DA" (Danish), "DE" (German),
(中略)
//      "SK" (Slovak), "SL" (Slovenian), "SV" (Swedish),
// DeepL翻訳(認証キー使用), Google翻訳で利用可能な言語たち
//      "ID" (Indonesian), "KO" (Korean), "TR" (Turkish),
//      "UK" (Ukrainian),
// DeepL翻訳(認証キー不使用), Google翻訳で利用可能な言語たち
//      "EN" (English),
//      (* DeepL翻訳(認証キー使用)では "EN-GB" または "EN-US" )
//      "PT" (Portuguese),
//      (* DeepL翻訳(認証キー使用)では "PT-BR" または "PT-PT" )
// DeepL翻訳(認証キー使用, 不使用)で利用可能な言語たち
//      "ZH" (Chinese)
//      (* Google翻訳では "zh-cn" または "zh-tw" ),
// DeepL翻訳(認証キー使用)でのみ利用可能な言語たち
//      "EN-GB" (English (British)), "EN-US" (English (American)),
//      (* DeepL翻訳(認証キー不使用), Google翻訳では "en" )
(中略)
//      "PT-BR" (Portuguese (Brazilian)), "PT-PT" (Portuguese (European)),
//      (* DeepL翻訳(認証キー不使用), Google翻訳では "pt")
// Google翻訳でのみ利用可能な言語たち
//      "af" (afrikaans), "sq" (albanian), "am" (amharic),
(中略)
//      "xh" (xhosa), "yi" (yiddish), "yo" (yoruba), "zu" (zulu),
//      (* ほかにもあるが、本ボットでは未対応)
"toLanguages": {
// 既定の翻訳先言語(たち)
"defaults": [ "JA", "", ],
//
// 翻訳元言語が既定の翻訳先言語であった場合の、代わりの翻訳先言語(たち)
"onesIfFromLanguageIsInDefaults": [ "EN-US", "EN", "", ],
},

```

```

//
// 翻訳先メッセージの構成
//
// 翻訳先メッセージの中で置換される文字列たち
//  {{senderUserName}} -> 送信ユーザー名(チャンネルURLの末尾)
//  {{senderDisplayName}} -> 送信ユーザーの表示名
//  {{toMessage}} -> 翻訳された送信ユーザーのメッセージ
//  {{fromLanguage}} -> 翻訳元言語
//  {{toLanguage}} -> 翻訳先言語
"messagesFormat": "{{senderUserName}}: {{toMessage}} ({{fromLanguage}} >
{{toLanguage}})",
},
//
// ■ メッセージたちの 棒読みちゃん への受け渡しに関する設定
"bouyomiChan": {
    // メッセージたちを受け渡すか否か
    // true (受け渡す), false (受け渡さない)
    "sendsMessages": false,
    //
    // 本ボット起動時に、自動で 棒読みちゃん を起動させる場合の、
    // 棒読みちゃん の実行ファイルの絶対パス
    // (* 例:
"C:\\Users\\youru\\Documents\\SoftwareWithoutInstaller\\BouyomiChan_0_1_11_0_Beta2
1\\BouyomiChan.exe" )
    //      (* 「 \ 」(フォルダー区切りの記号)は「 \\ 」(同じ記号2つ)に変更すること)
    //      (* "" とした場合や、間違ったパスを設定した場合は、自動で起動されない)
    //      (* 自動で起動させない場合は、本ボット起動前に 棒読みちゃん を手動で起動させておくこ
と)
    //      (* 自動で起動させた場合は、本ボットの停止と共に 棒読みちゃん も自動で停止)
    "autoRunKillPath":
"C:\\Users\\youru\\Documents\\SoftwareWithoutInstaller\\BouyomiChan_0_1_11_0_Beta2
1\\BouyomiChan.exe",
    //
    // 使用するローカル(本ボットを動かすPC)HTTPサーバ (localhost) のポート番号
    // (* 棒読みちゃん での設定値 (49152~65535) に合わせること)
    "portNo": 60080,
    //
    // 受け渡すメッセージたちに対する制限
    "limitsWhenPassing": {
        // 送信ユーザーのユーザー名ないし表示名の、末尾の算用数字部分を省略するか否か
        // true (省略する), false (省略しない)
        "ignoresSenderNameSuffixNum": true,
        //
        // 送信ユーザーのユーザー名ないし表示名の、先頭からの文字数の上限 (* 0~25)
        "numSenderNameCharacters": 25,
        //
        // 先頭からのエモート(スタンプ)数の上限 (* 0~125)
        "numEmotes": 3,
    },
    //
    // 受け渡さないメッセージたち
    "messagesToIgnore": {
        // 送信ユーザー名たち
        // (* 英大文字と英小文字が混在していても可)

```



```
// (* ボットとして運用するユーザーについては、ここに記載がない場合、
//     手動で投稿したメッセージたちは受け渡しを実行)
"senderUserNames": ["nightbot", "", ],
//
// ユーザーコマンドの接頭辞たち (* "<ter>_" は記載がなくても受け渡しを不実行)
"userCommandPrefixes": ["!", "", ],
//
// メッセージ内に含まれる文字列たち
"stringsInMessage": ["", ],
},
//
// 受け渡すメッセージの構成
//
// 受け渡すメッセージの中で置換される文字列たち
// {{senderUserName}} -> 送信ユーザー名(チャンネルURLの末尾)
// {{senderDisplayName}} -> 送信ユーザーの表示名
// {{senderMessage}} -> 送信ユーザーのメッセージ
"messagesFormat": "{{senderDisplayName}} san: {{senderMessage}}",
},
}
```

必須の設定

// ■ メッセージ送信先となるチャンネルに関する設定たち および // ■ メッセージ送信を行うボットに関する設定たち 以降にある以下の箇所について、必要な変更をして、上書き保存してください。

箇所	変更すべき部分	何に変更するか
"broadcasterUserName": "yourchannelname"	yourchannelname	配信を行うユーザー名 (チャンネルURLの末尾)
"oAuthAccessToken": "9y0urb0tuser0authaccesst0ken9"	9y0urb0tuser0authaccesst0ken9	上記の ボットとして運用するユーザーのユーザーアクセストークン文字列の取得 で得たトークン文字列
"nameColor": "blue"	blue	上の行の「名前の色」で候補として挙げられている、色を表す文字列たちから1つ

イベントに自動で応答する機能の設定

// ■ イベントたちに対する応答たちに関する設定 以降にある以下の箇所を、やりたいことに応じて変更して、上書き保存してください。

箇所	変更すべき部分	何に変更するか
----	---------	---------

箇所	変更すべき部分	何に変更するか
<code>[5, "!raided {{raidBroadcasterUserName}}", 5],</code>		順序が1つ前のコマンド・メッセージが実行されてから、表示したいメッセージが実行されるまで待機する時間（秒）（最初に実行されるコマンド・メッセージの場合は、レイドを受けてからの時間）
	<code>!raided {{raidBroadcasterUserName}}</code>	表示したいメッセージ（これを利用して、 ユーザーコマンドも実行可能 ）
	<code>[5, "!raided {{raidBroadcasterUserName}}",],</code>	このメッセージを表示したくない場合は、行ごと削除するか、行の頭に <code>//</code> を挿入（コメントアウト）
<code>[10, "/shoutout",],</code>	10	順序が1つ前のコマンド・メッセージが実行されてから、 <code>/shoutout</code> 公式コマンドが実行されるまで待機する時間（秒）
	<code>[10, "/shoutout",],</code>	<code>/shoutout</code> 公式コマンドを実行したくない場合は、行ごと削除するか、行の頭に <code>//</code> を挿入（コメントアウト）

例として `[5, "!raided {{raidBroadcasterUserName}}",],` および `[10, "/shoutout",],` の行を全く変更しない場合、レイドを受けたときに本ボットは以下の動作をします。

- まず、5秒待機したのち、チャット欄に「`!raided` レイド元のユーザー名(チャンネル名)」というメッセージを表示
 - もし [Twitchでレイドされたときに自動でお礼と宣伝をする方法](#) のとおりに [Nightbot](#) に `!raided` ユーザーコマンド表示したいメッセージを設定していた場合、チャット欄に「【表示名】さんレイドありがとうございます！【表示名】さん(【ゲーム名】をプレイ中)のチャンネルはコチラ→【URL】」と表示
 - 本ボットと [Nightbot](#) を設定すれば、[Streamlabs](#) ないし [StreamElements](#) といったサービス側の設定は不要
- 次に、10秒待機したのち、`/shoutout` レイド元のユーザー名 公式コマンドを実行

コマンド・メッセージの実行順序を変えたい場合は、`[]` で囲まれた行の上下を入れ替えてください。

`config.json5` の文字コードは、ダウンロード時点では `UTF-8 (BOMなし)` ですが、上書き保存した際にほかの文字コードに変わってしまっても、問題なく動作するように作ったつもりです。

チャット翻訳機能の設定

// ■ メッセージたちに対する翻訳に関する設定 以降にある箇所を、やりたいことに応じて変更して、上書き保存してください。なお、初期設定のままでも、上記の [必須の設定](#) を行っていれば、チャット翻訳機能は動作します。

- 初期設定以外のチャット翻訳サービスを使用したい場合：
 - DeepL翻訳で、認証キーを使用する場合：[DeepL翻訳の無料版APIキーの登録発行手順！世界のAI翻訳サービスをAPI利用](#)などを参考にして、キーを取得して入力したのち、行の頭の // を削除（アンコメント）
 - Google翻訳で、Google Apps Script (GAS) を使用する場合：[Google翻訳APIを無料で作る方法](#)などを参考にして、翻訳スクリプトを作成・デプロイし、ウェブアプリのURLを取得して入力したのち、行の頭の // を削除（アンコメント）
 - 翻訳サービスたちの優先使用順に応じて、[] で囲まれた行の上下を入れ替え
- チャット翻訳をしたくない場合：["deeplTranslate", "https://www2.deepl.com/jsonrpc",] および ["googleTrans", "translate.google.co.jp",] を行ごと削除するか、行の頭に // を挿入（コメントアウト）
 - ["deeplTranslate", "https://www2.deepl.com/jsonrpc",] は、429 Client Error: Too Many Requests for url: https://www2.deepl.com/jsonrpc というエラーが出て翻訳されない場合が多いため、初期設定の時点で行の頭に // を挿入済み

なお、各メッセージの前後に以下のような指定を付加することで、メッセージ単位で翻訳のされ方を細かく制御できます。

- `trnslt ^ (メッセージ)` : `config.json5` の設定により翻訳しないルールに該当するメッセージであっても強制的に翻訳
- `(翻訳サービス名) ~ (メッセージ)` : 翻訳サービスを指定
 - 例 : `deepltranslate ~ とりま` → `anyhow (JA > EN)` vs. `googletrans ~ とりま` → `Torima (ja > en)`
- `(翻訳元言語) > (メッセージ)` : メッセージを何語と認識するかを指定
 - 例 : `湯` → `hot water (JA > EN)` vs. `zh > 湯` → `スープ (ZH > JA)`
- `(メッセージ) > (翻訳先言語)` : メッセージを何語に翻訳するかを指定
 - 例 : `こんばんは` → `good evening (JA > EN)` vs. `こんばんは > fr` → `Bonne soirée (JA > FR)`

上記のオプションは、記述の順番を守れば、`trnslt ^ (翻訳サービス名) ~ (翻訳元言語) > (メッセージ) > (翻訳先言語)` などと、組み合わせて使用もできます。

棒読みちゃん連携機能の設定

// ■ メッセージたちの [棒読みちゃん](#) への受け渡しに関する設定 以降にある箇所を、やりたいことに応じて変更して、上書き保存してください。なお、初期設定のままでは、上記の [必須の設定](#) を行っていても、[棒読みちゃん](#) によるメッセージたちの読み上げはなされません。読み上げてもらうには、少なくとも以下の設定を行ってください。

- [棒読みちゃん](#) の変更と確認
 - バージョンを `Ver0.1.11.0 Beta21` に更新
 - 例えば [【読み上げ】棒読みちゃん連携](#) を参考に、基本設定の `01)ローカルHTTPサーバ機能を使う` の設定値を `True` に変更し、`02)ポート番号` の設定値を記憶

- エラーメッセージが表示されるようになった場合は、**02)ポート番号** の初期設定値である **60080** を **49152** ～ **65535** の間の別の整数値に変更するか、以下を参考に問題を解消
 - **棒読みちゃんの起動失敗時にパソコンの再起動をせず対応する手順**
 - **棒読みちゃんβ21でエラー「HTTPサーバを開始できませんでした(Port:50080)」が出ます。**
- 本ボットの **config.json5** の設定値の変更
 - **"sendsMessages"** の設定値を **false** → **true** に変更
 - **"portNo"** の設定値を、上記の **02)ポート番号** の設定値に変更

実行

さあ、本ボットを使いましょう！

なお、本ボットの起動や動作中であるかの確認が失敗する場合で、原因が上記の **ボットとして運用するユーザーのユーザーアクセストークン文字列の取得** で得たトークン文字列であると推測される場合は、ボットとして運用するユーザーでTwitchにログインし、「設定（アカウント設定）」の「**リンク**」の「その他のリンク」から、トークン文字列取得に使用したサービスをいったん「リンク解除」し、もう一度同じ方法でトークン文字列を取得すると、解決するかもしれません。

- 「Twitch Chat OAuth Password Generator（Twitch Chat OAuth Token Generator）ウェブサービスが、トークン文字列を悪用しないと信じる場合」を選択したのであれば、リンク解除するサービス名は「Twitch Chat OAuth Token Generator」

起動

- .exeファイル版：**twitch-eventsub-response-py.exe** を実行
- スクリプト版：Pythonで **./Code/main.py** を実行
 - ウィンドウを立ち上げたくない場合は、**main.py** の変数 **uses_tkinter_window** の値を **True** → **False** に変更すること

.exeファイル版は、Windowsやセキュリティソフトによりウイルスの疑いありと判定され、初回の起動が妨げられる可能性があります。その場合は、（もちろん本ボットはウイルスではないので）疑いを解除して起動できるようにしてください。

- .exeファイルはスクリプト版に **PylInstaller** を適用して生成しているが、**PylInstaller** を使用して生成した.exeファイルにはよく起こる現象

本ボットはネット通信を行うアプリであるため、初回起動時にファイアウォールソフトが通信をブロックしようとする可能性があります。その場合は、**通信を許可してください**。

正常に起動すると、配信のチャット欄に「YourBotUserName bot for <ter>_ has joined.」と表示されます。

また、本ボットのコンソール模擬ウィンドウ（自作の黒い画面）に以下のようなメッセージが表示されます。

```
----- Twitch EventSub Response Bot (v3.3.1) -----
[Preprocess]
JSON5 file path = C:\Users\youru\Desktop\twitch-eventsub-response-py-
v3.3.1\config.json5
parsing this file ... done.
```

```
[Activation of Bot]
  Initializing bot ...
    Message channel user name = yourchannelname
    Bot token length = 30
  done.

[Run of Bot]
  Joining channel ...
    Channel name = yourchannelname
  done.

  Making bot ready ...
    Bot user ID = 888888888
    Bot user name = yourbotusername
    Bot commands
      <ter>_kill
      <ter>_restart
      <ter>_test
    Bot cogs
      TERRaidCog
        [5, '!raided {{raidBroadcasterUserName}}']
        [10, '/shoutout']
      TERBouyomiCog
      TERTransCog
      Services
        GOOGLETRANS
          Getting instance ... done.
          Getting language detection function ... done.
        Setting bot name color = blue ... done.
    done.
```

- **Running**
 "C:\Users\youru\Documents\SoftwareWithoutInstaller\BouyomiChan_0_1_11_0_Beta21\BouyomiChan.exe" ... done. は、本ボット起動時に自動で **棒読みちゃん** も起動させる設定にしている場合に表示
- **Services** は、対応する翻訳サービスを使用する設定にしている場合に表示

動作中であるかの確認

配信のチャット欄に「<ter>_test」と入力すると、「YourBotUserName bot for <ter>_ is alive.」と表示されます。

また、本ボットのコンソール模擬ウィンドウ（自作の黒い画面）に以下のようなメッセージが表示されます。

```
Testing bot (v3.3.1) ...
  Channel name = yourchannelname
  Bot user ID = 888888888
  Bot user name = yourbotusername
```

```

Bot commands
  <ter>_kill
  <ter>_restart
  <ter>_test
Bot cogs
  TERRaidCog
  TERBouyomiCog
  TERTransCog
done.

```

この2つが表示されれば、本ボットは動作中です。

停止

- 直接停止させる方法
 - .exeファイル版：本ボットのコンソール模擬ウィンドウ（自作の黒い画面）を閉鎖
 - スクリプト版：実行中の `./Code/main.py` スクリプトを停止
 - 場合によっては、本ボットのコンソール模擬ウィンドウ（自作の黒い画面）を閉鎖する必要もあり
- 配信のチャット欄から停止させる方法：チャット欄に「`<ter>_kill`」または「`<ter>_kill (0から255の間の整数値(ただし、3以外))`」と入力
 - チャンネルの配信者またはボットとして使用するユーザーのみが実行可能
 - チャット欄に「`YourBotUserName bot for <ter>_ has stopped.`」と表示
 - `(0から255の間の整数値(ただし、3以外))` を入力した場合、本アプリのリターンコードに設定
 - 入力しなかった場合、リターンコードは `0`
 - `3` を入力した場合、本ボットが再起動（下記）

例えば、配信のチャット欄に `<ter>_kill 222` と入力した場合は、本ボットのコンソール模擬ウィンドウ（自作の黒い画面）に以下のようなメッセージが表示されます。

```

Killing bot ...
  Return code = 222
done.

[Postprocess]
  Return code = 222

```

- `Killing BouyomiChan ... done.` は、本ボット起動時に自動で [棒読みちゃん](#) も起動させている場合に表示

再起動

配信のチャット欄に「`<ter>_restart`」または「`<ter>_kill 3`」と入力すると、配信のチャット欄に「`YourBotUserName bot for <ter>_ has stopped.`」と表示されたあと、本ボットが再起動します。

- チャンネルの配信者またはボットとして使用するユーザーのみが実行可能

また、本ボットのコンソール模擬ウィンドウ（自作の黒い画面）に以下のようなメッセージが表示されます。

```
Killing bot ...  
  Return code = 3 (Restart)  
done.  
  
[Postprocess]  
  Return code = 3 (Restart)  
  Killing BouyomiChan ... done.
```

Restart after 4 s.

- `KillingBouyomiChan ... done.` は、本ボット起動時に自動で `棒読みちゃん` も起動させている場合に表示

異常中断

何らかの異常により本ボットの動作が中断し、動作が再開されない場合は、以下の手順を実施したあと、本ボットを再度 `起動` してください。

1. 上記の `停止` に書かれている方法で本ボットを停止
2. 使用しているならば `棒読みちゃん` も停止
3. `config.json5` の設定値の確認
4. 使用PCがインターネットにつながっているかの確認

ただし、上記の1.の方法で本ボットを停止させても、一部の機能が停止されずに（バックグラウンドで）動き続けている場合があります。その場合は、お手数ですが、以下の対応を行ってください。

- タスク マネージャー（アクティビティモニタ）などから、該当するものを強制終了
- .exeファイル版は、.exeファイルと同じフォルダー内に `_MEIxxxxxx`（`xxxxxx` の部分は数字）フォルダーが残されていることがあるので、その場合は削除
 - このフォルダーは、**本ボットが起動していない状態であればいつでも削除可能**
 - 削除できなければ、本ボットの一部機能が未停止
- どうしても一部の機能が動き続けている場合は、使用PCの再起動

アンインストール方法

`README.pdf` が格納されているフォルダーを削除してください。

今後の展開

いつリリースできるかはお約束できませんが、以下のような機能を追加したいと考えています。

- マイク音声を認識して文字起こしや翻訳を行い、OBS上に表示する機能

- [ゆかりねっとコネクターNEOのOBS連携](#) に相当
- 要望に応じて、自動で反応するイベントの種類や反応の内容の拡充
 - 本ボットのアプリ名には「[EventSub](#)」とついているが、EventSub購読が必要な機能は搭載の見込みなし
 - 各ユーザーないし誰かがHTTPS対応サーバーを用意しなければならないため

バージョン履歴

2025-05-26 (v3.3.1)

※ もし v3.3.0 で正常動作しているならばバージョンアップは不要

- 機能に影響のない変更
 - `config.json5` の初期設定値の変更
 - `//` ■ メッセージたちに対する翻訳に関する設定 以降にある `["deeplTranslate", "https://www2.deepl.com/jsonrpc",]`, の行の頭に `//` を挿入 (コメントアウト)
 - `429 Client Error: Too Many Requests for url: https://www2.deepl.com/jsonrpc` というエラーが出て翻訳されない場合が多いため
 - `//` ■ メッセージたちの 棒読みちゃん への受け渡しに関する設定 以降にある `"portNo": 50080`, を `"portNo": 60080`, に変更
 - 開発環境の Windows 11 のバージョンを 23H2 から 24H2 にアップしたところ、`50080` 番ポートが別の何かに使用されるようになったため
 - 使用しているPythonパッケージのアップデート
 - `googletrans (4.0.0rc1)` とそれが依存しているパッケージを除く
 - 今後の展開に関連したPythonパッケージの先行導入
 - プログラムコードのリファクタリング

2024-01-11 (v3.3.0)

- `trnslt ^ (メッセージ)` で `config.json5` の設定により翻訳しないルールに該当するメッセージであっても強制的に翻訳できるオプションを追加
- 翻訳サービスの指定方法を (翻訳サービス名) = (メッセージ) → (翻訳サービス名) ~ (メッセージ) に変更
- コンソール模擬ウィンドウ (自作の黒い画面) に出力されるメッセージを変更

2023-04-17 (v3.2.2)

- 棒読みちゃんの連携をしていない状態で、終了時にエラーがでるケースを修正

2023-04-17 (v3.2.1)

- チャットメッセージ発生時の処理順を 棒読みちゃん → 翻訳 に (したつもり)
- コンソール模擬ウィンドウ (自作の黒い画面) の起動待機間隔を 1秒 → 1/64秒 単位に
 - (Windowsの標準のタイマーの分解能は 1/64秒 間隔らしい)

2023-04-16 (v3.2.0)

- コンソール模擬ウィンドウ (自作の黒い画面) の導入
- .exeファイル版に `run.bat` を同梱しないように
 - 今後は `twitch-eventsub-response-py.exe` を直接実行

- 正常起動すると配信のチャット欄に表示されるメッセージを「 YourBotUserName bot for <ter>_ has joined and is ready. 」 → 「 YourBotUserName bot for <ter>_ has joined. 」に変更
- 停止時に 棒読みちゃん も自動で停止させるケースの拡充

2023-04-15 (v3.1.0)

- 本ボットの停止時に 棒読みちゃん も自動で停止させるケースの拡充

2023-04-14 (v3.0.0)

※ config.json5 の再設定が必要

- チャットメッセージを 棒読みちゃん に受け渡す機能を追加
- config.json5 の書式変更
 - bouyomiChan キーと値の追加
 - 翻訳先メッセージへの追加文字列の設定を見直し、 messagesFormat キーと値の追加
 - その他軽微な変更
- /me (メッセージ) 公式コマンド実行時に ACTION という文字列がメッセージに付加されて翻訳されるのを回避
- (翻訳サービス名) = (メッセージ) で翻訳サービス名を指定できるオプションを追加

2023-03-28 : v2.0

※ config.json5 の再設定が必要

- チャット翻訳機能を追加
- config.json5 の書式変更
 - translation キーと値の追加

2023-03-10 : v1.0

- .exeファイル版に run.bat を同梱し、こちらのほうを実行することを推奨するように変更
- チャット欄で表示されるボットユーザー名の色を設定できなくなっていたのを修正
 - Twitch APIの仕様変更が原因と推測
- <ter>_restart および <ter>_kill 3 コマンドの追加
 - .exeファイル版で run.bat を実行して本ボットを起動していた場合、本ボットが再起動する機能
 - チャンネルの配信者またはボットとして使用するユーザーのみが実行可能
- /shoutout 公式コマンドの実行に失敗した場合のリトライの回数制限を撤廃

2023-03-08 : v0.5

- .exeファイル版で _MEIxxxxxx (xxxxxx の部分は数字) 一時フォルダーが.exeファイルがあるフォルダーに生成されるように変更

2023-02-26 : v0.4

※ config.json5 の再設定が必要

- <ter>_kill または <ter>_kill (0から255の整数値) コマンドの追加
 - 本ボットを停止させる機能
 - チャンネルの配信者またはボットとして使用するユーザーのみが使用可能

- (0から255の整数値)を入力した場合、本アプリのリターンコードに設定
 - 入力しなかった場合、リターンコードは 0
- /shoutout 公式コマンドの実行に失敗した場合に、2分5秒後にリトライする機能の追加
 - リトライは1度だけ実行
- config.json5 の書式変更
 - userName キーを broadcasterUserName キーに変更

2023-02-24 : v0.3

※ config.json5 の再設定が必要

- config.json5 の書式変更
 - commands と messages のキーと実行順の指定を廃止し、上から順に実行されるように簡素化
 - raid を /raid に、shoutout を /shoutout にそれぞれ変更

2023-02-22 : v0.2

- ボットとして運用するユーザーとして、配信で使っているユーザー以外も指定可能に
- 1分間の間に複数のレイドを受けた場合、ボットがエラー終了しないように

2023-02-21 : v0.1

参考資料

- ボット全般
 - [Twitch Developer Documentation](#)
 - TwitchIO ([documentation](#)、[GitHub](#))
 - [TwitchIOの実装例](#)
 - [TwitchIOでTwitchのBotを作る](#)
 - [Twitch APIに必要なOAuth認証のアクセストークンを取得しよう](#)
 - 「トークンを取得してみる」
 - 「1. The OAuth implicit code flow」
- イベントに対する自動応答機能
 - [Twitchでレイドされたときに自動でお礼と宣伝をする方法](#)
- チャットメッセージの翻訳機能
 - [チャット翻訳ちゃん](#)
 - [DeepL翻訳の無料版APIキーの登録発行手順！世界一のAI翻訳サービスをAPI利用](#)
 - [Google翻訳APIを無料で作る方法](#)
- チャットメッセージの棒読みちゃんへの受け渡し機能
 - [棒読みちゃん](#)
 - [【読み上げ】棒読みちゃん連携](#)
 - [棒読みちゃんの起動失敗時にパソコンの再起動をせず対応する手順](#)
 - [棒読みちゃんβ21でエラー「HTTPサーバを開始できませんでした\(Port:50080\)」が出ます。](#)