

Stream Deck のススメ

Stream Deck とは

様々な機能を割り当てることが出来るマルチボタンデバイス。いわゆる左手デバイスです。

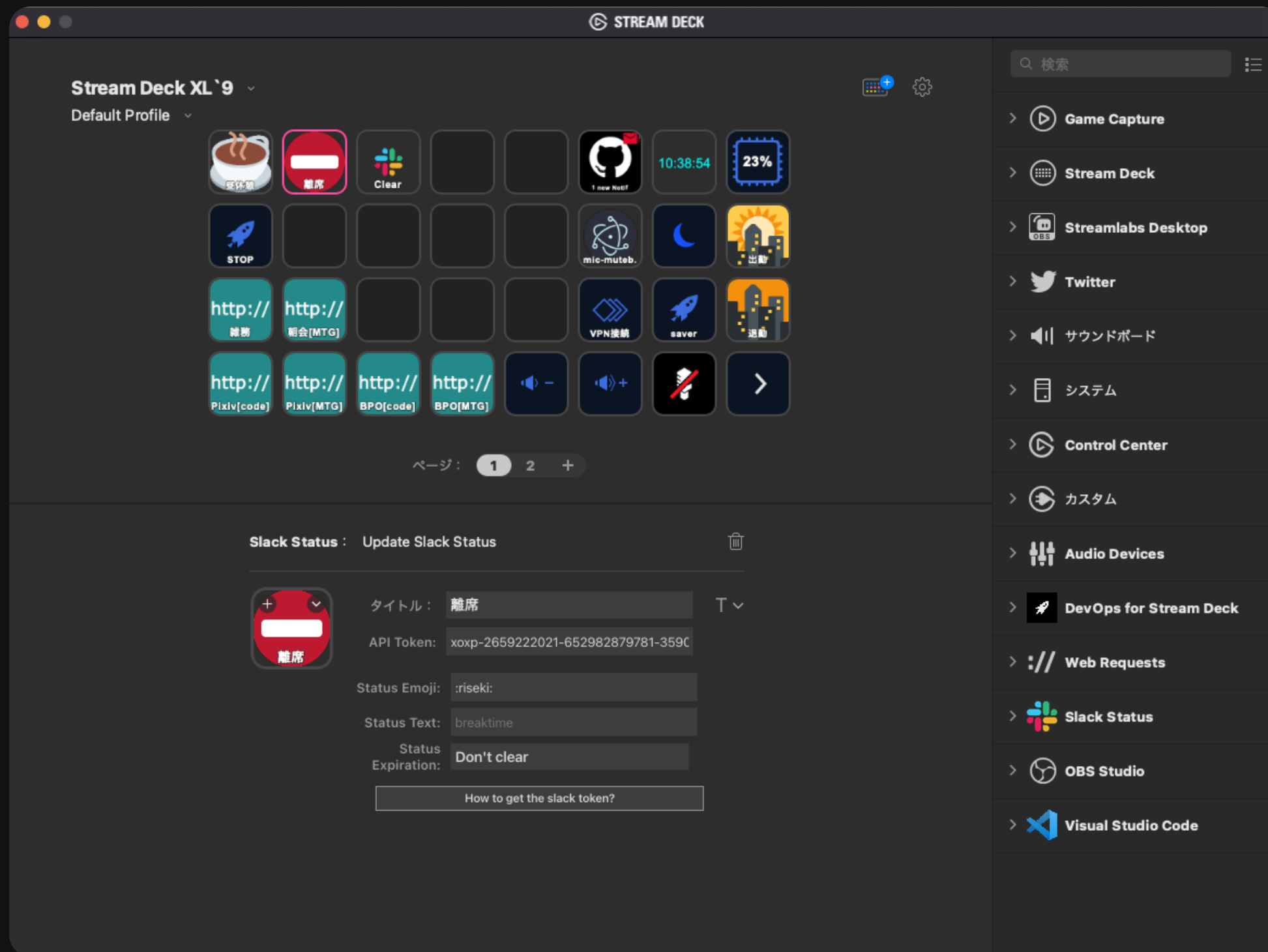
元々ライブ配信用に作成されたデバイスですが仕事に使う人も増えています。



出来ること

- フォルダを開く
- シェルコマンドの実行
- ショートカットキーの実行
- 定型文の登録
- 様々なアプリと連携
- ライブ配信(音楽の切り替え、シーンの切り替え、効果音を入れるetc…)

設定画面



使用例

出勤・退勤の自動化

- 自分で作成した自動打刻CLIツールを実行しています。
- 同時にSlackの退勤スタンプを設定しています。

マイクのミュート

- マイクのミュートのオンオフを切り替えることができます。
- MTG中いちいちブラウザ上でミュートするのは面倒くさいし見ただけで今の状態がわかるので便利です。

Toggl Track

- Togglというタスク管理ツールの更新を行います。
- ブラウザで変更するのは面倒なのでよく忘れてしまいましたが、これならすぐ切り替えられます。

自作プラグインの紹介

Slackのステータスを変更するプラグインです。

<https://github.com/ellreka/streamdeck-slack-status>

Slack Status : Update Slack Status

+

▼

離席

タイトル :

離席

T ▼

API Token:

xoxp-2659222021-652982879781-359C

Status Emoji:

:riseki:

Status Text:

breaktime

Status Expiration:

Don't clear

Don't clear

30 minutes

1 hour

4 hour

Today

This week

Custom

プラグインの作り方

ディレクトリ構成

```
├── com.example.plugin-name.sdPlugin // ビルド済みファイル
│   ├── images
│   │   ├── icon.png
│   │   └── icon@2x.png
│   ├── manifest.json
│   ├── code.html
│   ├── code.js
│   ├── pi.html
│   ├── pi.js
│   └── sdpi.css
├── src
│   ├── code.ts // プラグインのメイン処理を書くファイル
│   └── pi.ts // Property Inspector。プラグインの設定ページ
└── package.json
```


Property Inspector

プラグインの設定ページ。

色々なUIが用意されていてそれらを組み合わせて作成します。

メイン処理とのやり取りはWebSocketで行います。

Main

PIから受け取ったアクションを実行します。

ここでAPIを叩いたりします。

デバッグ方法

defaultsコマンドでデバッグモードをオンにします。

```
defaults write com.elgato.StreamDeck html_remote_debugging_enabled -bool YES
```

その後 <http://localhost:23654/> にアクセスするとDevToolsやログを見ることが出来ます。

esbuildを使ってTypeScriptをコンパイルする

`esbuild` では複数ファイルをそれぞれ別々にコンパイルすることが出来ないなので `concurrently` を使って並列処理しています。

→ これ書いてて気づいたけど `esbuild src/{pi,main}.ts --outdir=net.ellreka.slack-status.sdPlugin --target=esnext` でいける気がする。

```
"scripts": {  
  "build:main": "esbuild ./src/main.ts --bundle --outfile=./net.ellreka.slack-status.sdPlugin/main.js --target=esnext",  
  "build:pi": "esbuild ./src/pi.ts --bundle --outfile=./net.ellreka.slack-status.sdPlugin/pi.js --target=esnext",  
  "dev": "yarn concurrently \"yarn build:main --watch\" \"yarn build:pi --watch\"",  
  "build": "yarn build:main && yarn build:pi",  
}
```

パッケージをビルドする

専用のビルドツールを使ってパッケージをビルドします。

```
DistributionTool -b -i net.ellreka.slack-status.sdPlugin -o ~/Desktop/
```

成功すれば `net.ellreka.slack-status.streamDeckPlugin` というファイルが生成されます。

私はGithub Actionsで自動化しています。注意点としてMacOSかWindowsでしか動作しないのでOSを設定する必要があります。

```
jobs:  
  build:  
    runs-on: macos-latest
```

<https://github.com/ellreka/streamdeck-slack-status/blob/main/.github/workflows/build.yml>

プラグインを公開する

開発者ページなどが無いので、公開する場合は
elgato社にメールで連絡する必要があります。

必要事項を入力してメールして、OKをもらう
と公開してくれます。

私の場合「APIキーの取得方法を追加したほう
が良いんじゃない」みたいなアドバイスをくれ
ました。

e

ellreka <ellreka5364@gmail.com>
To streamdeck.elgato ▾

6月28日(火) 14:08 ☆ ↶ ⋮

- **Company/Developer:** ellreka
- **User:** ellreka
- **Description:** Update the slack status.
- **Release Notes:** First release
- **Support-Link:** <https://github.com/ellreka/streamdeck-slack-status>
- **Operating System:** mac/win
- **Category:** BUSINESS TOOLS

Package download link:
<https://github.com/ellreka/streamdeck-slack-status/raw/main/Release/net.ellreka.slack-status.streamDeckPlugin>

S

Streamdeck Elgato
To 自分 ▾

6月29日(水) 6:24 ☆ ↶ ⋮

🌐 英語 ▾ > 日本語 ▾ [メッセージを翻訳](#)

[次の言語で無効にする: 英語](#) ×

Hi ellreka,

This looks super cool! I was looking into maybe building a Slack plugin myself, so I'm glad to see it, but I'm having some trouble getting it working. It would be nice to see a "How To" guide in the support link, since the users (and myself) will have to know how to get a hold of the correct token.

Best,
Zack

3つ目のデバイスという選択肢

Loupedeck Live



StreamDeck Pedal

