

LINE Bot をつくってみよう

API を試して学んでしっかりわかる

mochikoAsTech 著

2023-05-20 版 mochikoAsTech 発行

はじめに

2023 年 5 月 mochikoAsTech

この本を手に取ってくださったあなた、こんにちは、あるいははじめまして。「LINE Bot をつくってみよう～API を試して学んでしっかりわかる～」の筆者、mochikoAsTech です。

想定する読者層

本書は、こんな人に向けて書かれています。

- ・「API を叩く」をやってみたい人
- ・「Bot を作る」をやってみたい人
- ・これからプログラミングを学ぼうと思っている人
- ・LINE API を使ってみたい人
- ・LINE で Bot を作ってみたい人

マッチしない読者層

本書は、こんな人が読むと恐らく「not for me だった…（私向けじゃなかった）」となります。

- ・コミュニケーションアプリ「LINE（ライン）」を使ったことがなく、基本操作から知りたい人
- ・LINE 公式アカウントで友だちをうまく増やす方法を知りたい人
- ・LINE Official Account Manager の使い方を知りたい人
- ・LINE で簡単に月 20 万稼げる！みたいなことがしたい人

本書の特徴

本著では実際に LINE Bot を作ったり、メッセージを配信したりします。先に手を動かして試して、続いて裏付けになる事柄を学ぶ、というステップを交互に踏むため、理解がしやすく、プログラミング初心者でも飽きずに読み進められる内容です。

また実際にありがちなトラブルをとり上げて、

- こんなエラーが起きたら原因はどう調べたらいいのか？
- 問題をどう解決したらいいのか？
- どうしたら事前に避けられるのか？

を解説するとともに、実際にコマンドを叩いて反復学習するためのドリルもついています。

紙の書籍をお買い上げいただいた方は、フルカラーの PDF 版を無料でダウンロードできますので、ぜひご活用ください。詳しくは「あとがき」にある「PDF 版のダウンロード」を参照してください。

用意するもの

- LINE の入ったスマートフォン

本書のゴール

本書を読み終わると、このような状態になっています。

- LINE Bot の仕組みを理解している
- LINE Bot を作るときに何を準備すればいいか分かっている
- プロバイダーやチャネルに何を登録すべきか分かっている
- 読む前より LINE API が好きになっている

免責事項

本書に記載された社名、製品名およびサービス名は、各社の登録商標または商標です。

本書は LINE API 好きな筆者が個人として執筆したものであり、本書に記載されている内容はいずれも筆者の所属する組織の公式見解ではありません。

本書は一般に開示されている情報を元に書かれており、筆者が所属する組織で知り得た情報は含まれていません。

また本書はできるだけ正確を期すように努めましたが、筆者が内容を保証するものではありません。よって本書の記載内容に基づいて読者が行なった行為、及び読者が被った損害について筆者は何ら責任を負うものではありません。

不正確あるいは誤認と思われる箇所がありましたら、必要に応じて適宜改訂を行いますので GitHub の Issue や Pull request で筆者までお知らせいただけますと幸いです。

<https://github.com/mochikoAsTech/startLINEBot>

目次

はじめに	3
想定する読者層	3
マッチしない読者層	3
本書の特徴	4
用意するもの	4
本書のゴール	4
免責事項	4
第1章 LINE 公式アカウントをつくってみよう	11
1.1 用語の定義	11
1.2 LINE 公式アカウントとは	12
1.2.1 LINE 公式アカウントにかかる費用	13
1.3 LINE 公式アカウントを作つてみる	14
1.3.1 LINE で LINE 公式アカウントを作つてみる	14
1.3.2 スマートフォンに LINE 公式アカウント管理アプリをインストー ルする	26
【コラム】個人の LINE アカウントを LINE 公式アカウントに切り替え られる？	28
1.3.3 LINE 公式アカウントと普通の LINE アカウントとの違い	29
1.3.4 LINE 公式アカウントと友だちになる	30
1.3.5 LINE 公式アカウントから友だちにメッセージを送つてみる	30
1.3.6 メッセージの配信対象を属性で絞り込む	31
第2章 Messaging API で LINE Bot をつくってみよう	33
2.1 そもそも API とは	33
2.2 Messaging API とは	34

2.2.1	開発には「個人の LINE アカウント」が必須か？	34
2.2.2	LINE 公式アカウントと Bot の違い	34
2.2.3	チャットボットとは	34
2.2.4	プレミアム ID とベーシック ID	35
2.2.5	未認証アカウントと認証済アカウント	35
	【コラム】開発中の LINE 公式アカウントをリリース日まで非公開にしておける？	36
	【コラム】開発中の LINE 公式アカウントに知らない人が友だち追加されてしまった。ブロックできる？	36
	【コラム】開発チームだけにメッセージのテスト配信がしたい	37
2.2.6	プロバイダーとは	38
2.2.7	チャネルとは	38
2.2.8	アカウントの管理方法	38
2.3	LINE 公式アカウントの運用方法	38
2.4	LINE キャンパスで資格を取ってみよう	38
2.5	ツール編	38
2.5.1	LINE Official Account Manager で運用する	38
2.5.2	LINE 公式アカウント管理アプリで運用する	38
2.5.3	Messaging API で運用する	38
2.5.4	非公式の管理画面で運用する	38
2.6	返信方法編	38
2.6.1	方法 1. 固定の自動応答を設定しておいて個別の応対は一切しない	39
2.6.2	方法 2. 人間が手打ちのチャットで返信する	39
2.6.3	方法 3. 内容に応じた自動応答メッセージで応対する	39
2.6.4	方法 4. Messaging API で返信する	40
	【コラム】LINE 公式アカウントの「既読」はいつ付くのか？	40
2.6.5	Webhook とは	41
2.7	用意するもの	42
2.7.1	ボットサーバー	42
2.7.2	エラーの統計情報	43
2.7.3	ボットサーバーが Webhook を受け取れなかったときの再送機能	43
	【コラム】Webhook へのレスポンスが先？ 応答メッセージが先？	44
	【コラム】これは本当に LINE プラットフォームから来た Webhook？	44
2.8	メッセージを送る	45
2.8.1	友だち追加されたときのあいさつメッセージ	45

2.8.2 友だち全員にメッセージを一斉配信する	45
【コラム】URL を送る前に OGP の見た目を事前に確認したり、キャッシュを消したりしたい	45
2.8.3 特定の属性を指定してメッセージを送る	47
2.8.4 ユーザー ID を指定して特定の人に送る	47
2.8.5 メッセージ送信元のアイコンと表示名を変更する	47
2.8.6 無料通数を超えたたら自動で追加課金されるのか	47
2.9 メッセージを Webhook で受け取る	47
2.9.1 画像などのメディアファイルを指定して取得する	47
2.10 メッセージ以外のイベントを Webhook で受け取る	47
2.10.1 友だちになったとき	47
2.10.2 ブロックされたとき	47
2.10.3 ユーザーが何かをタップしたとき	47
2.11 リッチメニュー	47
2.12 リッチメニュー プレイグラウンドでリッチメニューを体験してみる	49
2.13 リッチメニューの画像を作る	50
2.13.1 LINE Official Account Manager と Messaging API でのリッチメニューの制約の違い	50
2.13.2 リッチメニューを設定する	50
2.14 リッチメニューの設定の種類	51
2.15 3つの設定方法と表示の優先順位	51
2.15.1 OA Manager で設定したリッチメニュー	51
2.15.2 Messaging API で設定したリッチメニュー	51
2.15.3 ユーザー単位のリッチメニュー	51
2.16 どんな見た目のメッセージが送れるのか	51
2.16.1 テキストメッセージ	51
2.16.2 スタンプメッセージ	51
2.16.3 画像メッセージ	51
2.16.4 Flex Message	51
2.16.5 Flex Message Simulator は便利	51
2.17 やりとりする場所	52
2.17.1 ユーザーとのトーク	52
2.17.2 グループトーク	52
2.17.3 複数人トーク	52
2.18 Messaging API 以外のプロダクトとの組み合わせ	52

目次

2.18.1 LINE Social Plugins	52
2.18.2 LINE VOOM	52
2.18.3 LINE ログイン	52
2.18.4 LIFF や LINE ミニアプリ	52
2.18.5 プロバイダーページ	52
2.19 開発を始める	52
2.20 Messaging API SDK を使おう	52
2.20.1 Java	52
2.20.2 PHP	52
2.20.3 Go	52
2.20.4 Perl	52
2.20.5 Ruby	52
2.20.6 Python	52
2.20.7 Node.js	53
2.21 コミュニティに参加しよう	53
2.21.1 LINE Developers community	53
あとがき	55
PDF 版のダウンロード	56
Special Thanks:	56
レビュー	56
参考文献	56
著者紹介	57

第1章

LINE 公式アカウントをつくってみよう

1.1 用語の定義

気づいていないかもしれません、実は私たちは「LINE（ライン）」という言葉を、無意識に色々な意味で使っています。

「LINEってやってる？」「うん。でも家族としか使ってない」というときの LINE は、コミュニケーションアプリとしての LINE アプリのことを指しています。

「昨日送った LINE 見た？」「ごめん。寝てた」というときの LINE は、LINE アプリのトーク上でやりとりするメッセージのことを指しています。

「LINEってヤフーと経営統合したんでしょ」「そうなんだ」というときの LINE は、LINE 株式会社のことを指しています。

「お支払いは何になさいますか？」「あ、LINEで！」というときの LINE は、スマホで送金や決済ができる LINE Pay というサービスを指しています。

こんなふうに、同じ「LINE」という言葉でも文脈によって色々な意味になるため、本書での用語定義を最初にお伝えしておきます。

- ・本書では、LINE アプリのことを「LINE」と呼びます。ただ「LINE」とだけ書いてあったら、これは LINE アプリのことだな、と思ってください。
- ・LINE アプリのトーク上でやりとりするメッセージのことは、「メッセージ」と呼びます。
- ・会社としての LINE は、「LINE 株式会社」と呼びます。
- ・それ以外の各サービスは「LINE Pay」のような正式名称で呼びます。

1.2 LINE 公式アカウントとは

服屋さんのレジで「LINEで友だちになると5%オフになるクーポンがもらえるんですが、よかつたらどうですか?」と声をかけられたり、カフェでテーブルの上に「友だち追加するとワンドリンク無料!」と書かれたポップが置いてあるのを見たことはありませんか? (図1.1)



▲図1.1 街中で見かけるLINE公式アカウント

普段、あなたはLINEで友だちとメッセージや写真をやりとりしていると思います。実在する人間の友だちとやりとりするのと同じように、LINEでは企業や店舗ともやりとりできます。この企業や店舗のアカウントが「LINE公式アカウント」と呼ばれるものです。

LINE公式アカウントには、企業や店舗だけでなく、芸能人のものも存在します。たとえば地震のときに誰よりも早く「大丈夫?」とメッセージをくれて、彼氏感がすごい、と話題になったLINE公式アカウントもありました。あまりのプライベートっぽさに、まるで普通にLINEで友だちになったような感覚になりますが、落ち着いてください。あれはLINE公式アカウントです。^{*1}

^{*1} これは好きになっちゃう…。地震直後に送られた佐藤健のLINEが話題に | BuzzFeed - バズフィードジャパン <https://www.buzzfeed.com/jp/harunayamazaki/satotakeru-line-0214>

以前の「LINE@（ラインアット）」という名称の方が見覚えがある、という人も居るかもしれません。LINE@は2019年4月18日にLINE公式アカウントへ統合された^{*2}のですが、いまだにLINE@の名前で友だちを募集している店頭ポスターを見るのも少なくありません。

このLINE公式アカウントは、企業や店舗などの法人、あるいは芸能人のような「特別に許可を得た人」だけでなく、誰でも、つまりあなたでも無料ですぐに作ることができます。

1.2.1 LINE 公式アカウントにかかる費用

無料ですぐに作れると言われても、本当にお金がかからないのか気になりますよね。LINE公式アカウントにかかる必要について説明しておきましょう。

LINE公式アカウントには、3つのプランがあります。それぞれのプランで使える機能に差はなく、無料メッセージの通数や、無料メッセージを使い切ったときに、さらに追加でメッセージが送れるか、という点でのみ違いがあります。

- ・コミュニケーションプラン（0円。無料メッセージは200通/月）
- ・ライトプラン（月額5,000円。無料メッセージは5,000通/月）
- ・スタンダードプラン（月額15,000円。無料メッセージは30,000通/月。さらに追加メッセージは1通3円）

2023年5月31日までは、フリープラン、ライトプラン、スタンダードプランの3種類でしたが、2023年6月1日^{*3}からフリープランの名称がコミュニケーションプランに代わり、それぞれのプランで送れる無料メッセージの通数も変わりました。^{*4}

たとえば友だちが100人いるLINE公式アカウントから、すべての友だちに一斉送信する形でメッセージを2回送ったら、それだけで200通になりますね。こんなふうにたくさんの友だちにメッセージを継続して送ろうとすると、有料のプランを使う必要がありますが、逆に友だちが自分ひとりだけなら200回送れます。なので本書でちょっと試してみる程度なら無料のフリープランで問題ありません。無料メッセージの通数を使い切れば、

^{*2} LINE@サービス統合および移行について <https://d.line-scdn.net/stf/line-lp/migrationinfomation-LINEAT190718.pdf>

^{*3} 本書が出る技術書典14の開催期間（2023年5月20日から2023年6月4日）の途中でちょうどプランが変わるので、ここでは分かりやすく、「今日は2023年6月1日以降だ！」という認知で進みます。2023年5月時点で本書を読んでいるみなさん、いいですか、いまは2023年6月1日以降です。もうプランは変わったのです。

^{*4} 【重要：再掲】LINE公式アカウントの月額プラン改定に伴うプラン料金の日割り廃止のお知らせ | LINE for Business <https://www.linebiz.com/jp/news/20230125/>

次の月初までメッセージが送れなくなるだけで、自分で有料のライトプランやスタンダードプランに申し込まない限りは、メッセージの使いすぎで勝手に課金が始まることはありますので安心してください。

1.3 LINE 公式アカウントを作つてみる

色々説明しましたが、実際に手を動かしてやってみるのが理解への一番の近道です。LINE 公式アカウントを作つてみましょう。

LINE 公式アカウントを作る方法には、以下の 4 つがあります。どれで作つても同じ「LINE 公式アカウント」ができます。

1. LINE で作る
2. LINE 公式アカウント管理アプリで作る
3. LINE Official Account Manager で作る
4. LINE Developers コンソールで作る

方法は色々ありますが、注目してほしいのは 1 番目！ そう、なんと専用のアプリや管理画面を使わなくても、お手持ちのスマートフォンに入っている LINE で作れるんです！ 早速、LINE 公式アカウントを作つてみましょう。

1.3.1 LINE で LINE 公式アカウントを作つてみる

まずは LINE を開いて、下に並んでいるホーム、トーク、VOOM、ニュース、ウォレットという 5 つのメニューのうち、中央にある [VOOM] をタップしてください。（図 1.2）

1.3 LINE 公式アカウントを作成する



▲図 1.2 下部の中央にある [VOOM] をタップ

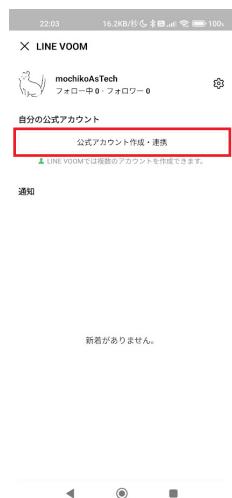
VOOM とやらがなんなのかということはいまは考えずに進みましょう。きっとショート動画のプラットフォーム的な何かです。VOOM を開いたら、右上にある人型のアイコンをタップします。(図 1.3)

第1章 LINE 公式アカウントをつくってみよう



▲図 1.3 右上にある人型のアイコンをタップ

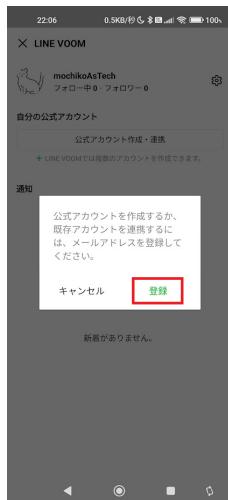
そして「自分の公式アカウント」の下にある「公式アカウント作成・連携」というボタンをタップします。(図 1.4)



▲図 1.4 「公式アカウント作成・連携」をタップ

このとき、もし「公式アカウントを作成するか、既存アカウントを連携するには、メー

「メールアドレスを登録してください」と表示されたら、あなたはまだLINEアカウントでメールアドレスを登録していないようです。その場合は、そのまま【登録】をタップします。(図1.5)



▲図1.5 メールアドレスが未登録だった場合は【登録】をタップ

LINEアカウントに登録するメールアドレスを入力して、【次へ】をタップしてください。(図1.6)

第1章 LINE 公式アカウントをつくってみよう



▲図 1.6 メールアドレスを入力して【次へ】をタップ

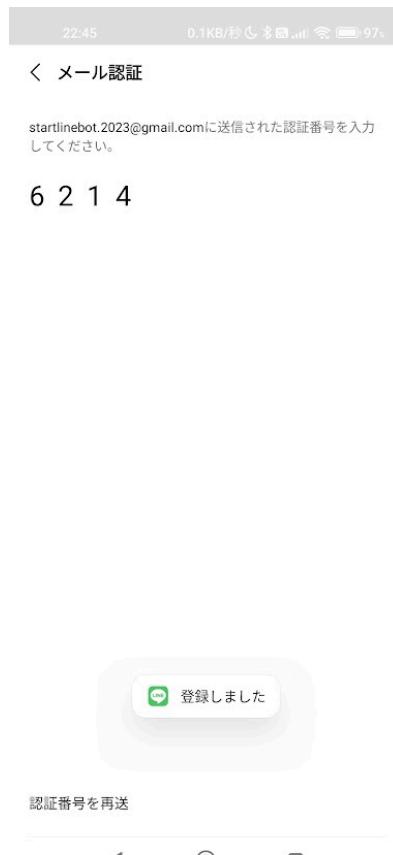
すると入力したメールアドレス宛てに「[LINE] メールアドレス登録確認メール」という件名のメールが届きます。(図 1.7)



▲図 1.7 メールアドレス登録確認メールが届く

メールの本文に書いてある 4 枠の認証番号を、LINE の【メール認証】の画面で入力し

てください。(図 1.8)



▲図 1.8 4 行の認証番号を LINE で入力

LINE で [メールアドレスが登録されました。] というメッセージが届いたら、メールアドレスの登録は完了です。(図 1.9)*5

*5 登録したメールアドレスを後で変更したくなったら、下に並んでいる [ホーム] から右上の歯車アイコンをタップして [設定] を開き、[アカウント] の [メールアドレス] で変更できます。

第1章 LINE 公式アカウントをつくってみよう



▲図 1.9 メールアドレスを登録するとメッセージが届く

それでは改めて、VOOM の右上にある人型のアイコンから、[公式アカウント作成・連携] というボタンをタップしてください。すると、LINE 公式アカウントを紹介する画面が表示されますので、続けて [公式アカウントを作成] をタップします。(図 1.10)



LINE公式アカウントを作成

公式アカウントでコミュニケーションの機会が広がります。



▲図 1.10 [公式アカウントを作成] をタップ

まずは LINE 公式アカウントのプロフィール画像とアカウント名を設定します。アカウント名と言われてもピンとこないかもしれません、もしあなたが技術書典に出展しているサークルさんなら、サークルの名前とアイコンを設定して、サークルの宣伝をする LINE 公式アカウントにしてもいいかもしれません。あるいは自分のハンドルネームにしてもいいでしょう。このアカウント名はユニークではないため、既存の LINE 公式アカウントとかぶる名称であっても付けられます。なおプロフィール画像もアカウント名も後から変更が可能ですので、悩みすぎずに取りあえずの画像や名前を設定しても構いません。(図 1.11)

第1章 LINE 公式アカウントをつくってみよう



▲図 1.11 プロフィール画像とアカウント名を設定しよう

カテゴリーの大業種は「個人」を選ぶと、小業種に「個人（芸術・創造・執筆）」があります。たとえば自分の本の告知用という位置づけで作るなら、この「個人（芸術・創造・執筆）」という小業種にしますが、他にも「個人（IT・コンピュータ）」や「個人（学生）」、「個人（その他）」などもあるので、個人的に開発を試す用途など、LINE公式アカウントの使い道に応じて適していると思われる業種を選択してください。「会社・事業者の所在国・地域」は、個人の場合は店舗や居住地を選択すればよいので、もしあなたが日本にお住まいなら「日本」を選びます。（図 1.12）

1.3 LINE 公式アカウントを作成する

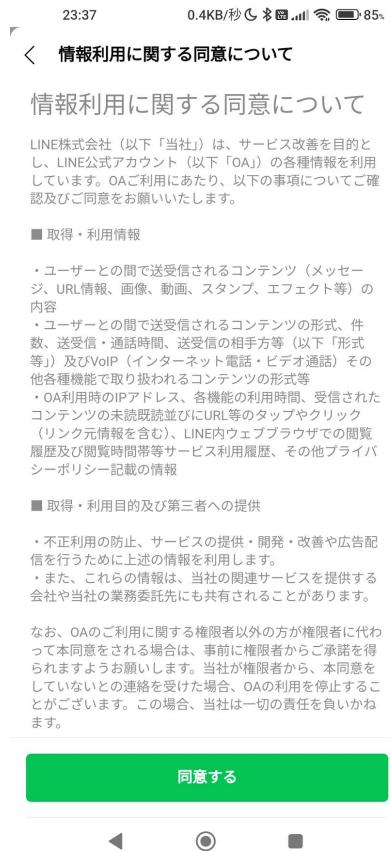


▲図 1.12 使い道に応じて適した大業種と小業種を選ぼう

下部に表示されている「LINE 公式アカウント利用規約」と「LINE ビジネス ID 利用規約」はしっかりと読んで^{*6}、記載内容に自分が同意できることを確認してから「アカウントを作成」をタップします。

ここからはいくつか同意の必要なページが出てきます。まず「情報利用に関する同意について」です。こちらもよく読んで、自分が同意できる内容であることを確認してから「同意する」をタップします。（図 1.13）

^{*6} 全部入力してから利用規約を開いて読み、Android の戻るボタンで戻ったら、入力した項目が全部消えていた。つらい。



▲図 1.13 よく読んでから【同意する】をタップ

おめでとうございます！ これで自分の LINE 公式アカウントができました！（図 1.14）

1.3 LINE 公式アカウントを作ってみる



▲図 1.14 LINE 公式アカウントができた！

「コンテンツを投稿して LINE VOOM をはじめよう！」とあるので、どうやらこれで、いま作った LINE 公式アカウントから LINE VOOM に動画が投稿できるようです。

でも今回やりたいことは LINE VOOM での動画投稿ではなく、友だち追加してくれた相手に対するメッセージの送信です。LINE 公式アカウントから友だちにトークでメッセージが送りたいときはどうすればいいのでしょうか？

1.3.2 スマートフォンに LINE 公式アカウント管理アプリをインストールする

作ったばかりの LINE 公式アカウントから友だちにメッセージを送るため、既に LINE を使っているスマートフォンで、LINE 公式アカウント管理アプリをインストールします。

- LINE 公式アカウント管理アプリ
 - Android (Google Play)
 - * <https://play.google.com/store/apps/details?id=com.linecorp.lineoa>
(図 1.15)
 - iPhone (App Store)
 - * <https://apps.apple.com/jp/app/id1450599059> (図 1.16)



▲図 1.15 Android (Google Play) 向けの管理アプリ



▲図 1.16 iPhone (App Store) 向けの管理アプリ

インストールした LINE 公式アカウント管理アプリを開いて、[LINE アプリで登録・ログイン] をタップします。 (図 1.17)

1.3 LINE 公式アカウントを作成する

23:45 207KB/秒 83%

LINEで企業やお店の
宣伝をはじめよう！



LINEでつながったお客さまに企業や
お店の情報をメッセージで一斉配信

● ● ●

LINEアプリで登録・ログイン

メールアドレスで登録・ログイン

ログイン方法でお困りの方へ ②

◀ ◎ ▶

▲図 1.17 [LINE アプリで登録・ログイン] をタップ

先ほど LINE 公式アカウントを作成するときに確認を求められた、[LINE ビジネス ID 利用規約]への同意が再び求められます。問題なければ [ログイン・登録] をタップします。

すると LINE 公式アカウントの管理アプリに対して、LINE で自分が登録しているプロフィール情報を提供して良いか、という認証画面が表示されます。問題なければ [許可する] をタップします。もしここで許可した情報提供を、後で取り消したくなった場合は、LINE の [設定] から [アカウント] を開いて、[連動アプリ] で確認したり、取り消したりできます。

出てきたー！ さっき作った LINE 公式アカウントが、管理アプリで表示されました！ 早速、[メッセージを配信する] をタップして、メッセージを送ってみましょう。

[追加] をタップして、先ずは [テキスト] を選び、適当なテキストを入力します。さらに [追加] をタップして、今度は [スタンプ] を選び、[スタンプを選択] から適当なスタンプを選びます。これでどんなメッセージが送られるのか、右上の [プレビュー] から確認してみましょう。

[トークルーム] では、こんな感じの見た目になるようです。[トーキリスト] をタップすると、トークの一覧画面での見え方も確認できます。

[次へ] をタップして、そのまま [すべての友だち] に送ってみましょう。さてここで [すべての友だち] に送ったら、LINE の友だち全員にこの変なメッセージが届いてしまわないか、ちょっと不安になりますね。

でも大丈夫です。あなたは LINE から、LINE 公式アカウントを作りましたが、LINE の友だちと、LINE 公式アカウントの友だちは別物なので、LINE の友だち全員にこのメッセージが届いてしまうことはありません。安心して [配信] をタップしましょう。

では自分の LINE に戻って、メッセージが届いたか確認してみます。みんなに届かなくても、少なくとも LINE 公式アカウントを作った創造主である自分には届いているかな、と思いましたが、残念ながら何も届いていません。なぜならばあなたはまだ、自分で作った LINE 公式アカウントと友だちになっていないからです。かなしい。[メッセージ配信] から [配信済み] を開いて、いま送ったメッセージを確認しても [配信人数] はゼロのままで。なぜならば友だちがいないから！

【コラム】個人の LINE アカウントを LINE 公式アカウントに切り替えられる？

たとえばあなたがおしゃれなタピオカミルクティー屋さんの店員で、TikTok などでたくさんのフォロワーを抱えるインフルエンサーでもあり、LINE の友だちも 3,000 人くらいいるとします。^{*7}

折角友だちがいっぱいいるので、TikTok で認証済みアカウントになって認証バッジをもらうような感じで、個人の LINE アカウントを運営に認証してもらって LINE 公式アカウントに切り替えられないのかな？ と思ったのですが、そんなことは可能なのでしょうか？

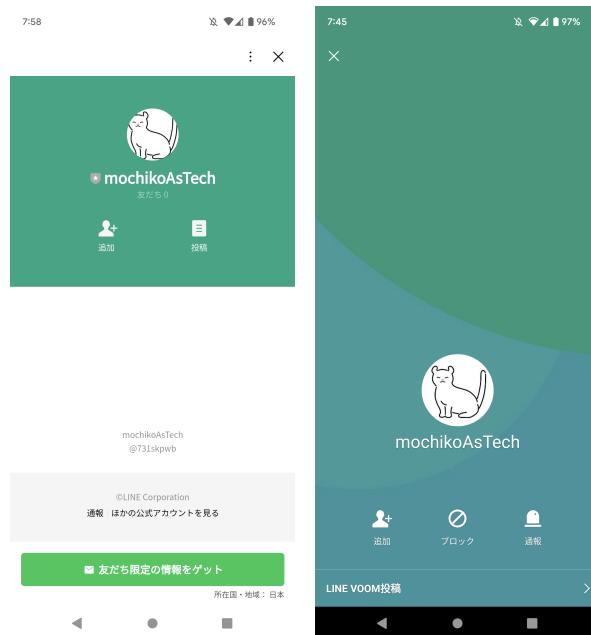
結論から言うと、個人の LINE アカウントを LINE 公式アカウントに昇格させることはできません。折角個人の LINE アカウントでたくさんの友だちを集めて、それをそのまま LINE 公式アカウントに引き継ぐことはできないので、仕事や宣伝を目的として LINE で友だちを増やしたいなら、最初から LINE 公式アカ

ウントではじめるのがオススメです。

1.3.3 LINE 公式アカウントと普通の LINE アカウントとの違い

LINE 公式アカウントと普通の LINE アカウントは、どうやって見分けがつくのでしょうか？ LINE 公式アカウントを運用している本人にはもちろん分かりますが、どちらも QR コードを読み込んで友だち追加する形なので、友だちになる側はどこで「あ、これ LINE 公式アカウントだ」と気づけるのでしょうか？

友だち追加の画面を比較してみましょう。左が LINE 公式アカウントで、右が普通の LINE アカウントです。（図 1.18）



▲図 1.18 LINE 公式アカウント

普通の LINE アカウントには、追加、ブロック、通報のボタンが並んでいますが、LINE 公式アカウントには追加、投稿、が並んでいます。また LINE 公式アカウントの方は、

*7 普段、Twitter で暮らしているので、おしゃれなインフルエンサーの解像度が低い。

ベーシック ID が表示されていたり、「ほかの公式アカウントを見る」とあったり、「友だち限定の情報をゲット」と出ているので、ちょっと変な感じがしますね。

1.3.4 LINE 公式アカウントと友だちになる

LINE 公式アカウントからメッセージを送ったら、ちゃんと「友だち」に届くのかを確認するため、まずは自分が LINE 公式アカウントと友だちになってみましょう。

LINE 公式アカウントの管理アプリで「友だちを増やす」をタップして、[URL を作成] を開きます。[URL をコピー] をタップして、友だち追加用の URL をコピーしましょう。

LINE に戻って、Keep メモという自分専用トークルームみたいなところに友だち追加 URL を貼ります。元気よく「友だち募集中」と表示されました。タップして友だちになってしましょう。

[友だち 0] [フォロワー 1] と出ました。フォロワーというのは LINE VOOM のフォロワーのこと、さっき LINE 公式アカウントを作ったときに、自分で自分をフォローしていたので、この 1 は自分のことです。知らない誰かではありません。

さっそく [追加] のアイコンをタップして、LINE 公式アカウントと友だちになってみます。友だちになると、アイコンが [追加] から [トーク] に変わるので、[トーク] をタップしてトークルームを開いてみましょう。

設定した覚えもないのに「はじめまして！ 友だち追加ありがとうございます」という元気なメッセージが届きました。誰だお前、私が作った LINE 公式アカウントだけど誰だ。後述しますが、これは「あいさつメッセージ」という機能で、友だち追加されたときに、自動で任意のメッセージを送ることができます。

この友だち追加用の URL をうっかり Twitter などに書いてしまうと、誰でもあなたの LINE 公式アカウントと友だちになってしまいます。自分しかメッセージを受け取っていないつもりで色々テストしていたら、実は自分以外の人もひっそり見ていた、ということにならないよう、URL の取り扱いには注意しましょう。

1.3.5 LINE 公式アカウントから友だちにメッセージを送ってみる

再び LINE 公式アカウント管理アプリに戻ると、「ターゲットリーチ」が先ほどの 0 から 1 に変わっています！ ターゲットリーチとは、要は「送信対象となる友だち」の母数のことのようです。友だちできた…ひとりできた…自分だけ…。鏡に向かって「私たち今日からお友だちだね…！」と言っている気分になってきましたが、例え相手が自分自身だろうが友だちは友だちです！ メッセージを送ってみましょう。

先ほどと同じように [メッセージを配信する] をタップします。

[追加] をタップして、先ずは [テキスト] を選び、適当なテキストを入力します。さらに [追加] をタップして、今度は [スタンプ] を選び、[スタンプを選択] から適当なスタンプを選びます。これでどんなメッセージが送られるのか、右上の [プレビュー] から確認してみましょう。

[トークルーム] では、こんな感じの見た目になるようです。さっきより圧の強いメッセージが準備できました。

[次へ] をタップして、そのまま [すべての友だち] に送つてみましょう。さっきは誰にも届きませんでしたが、いまはたつたひとりの友だちに届くはずです。私に届け！ 私のメッセージ…！ 祈りを込めて [配信] をタップします。

ピロン！ LINE の通知が来ました！

圧の強いメッセージがきました。やつたね！ これで「LINE 公式アカウントから友だちにメッセージを送る」というアチーブメントを達成しました。

1.3.6 メッセージの配信対象を属性で絞り込む

第2章

Messaging API で LINE Bot をつくってみよう

2.1 そもそも API とは

さて、Messaging API とは何か、という話をする前に、そもそも「API」とは何か、という話をさせてください。

そもそも Messaging API の「API」は、「REST API」^{*1}のことです。

通常、私たちはブラウザで URL を入力したり、リンクをクリックしたりすることで「このウェブページを見せてくれ！」とリクエストを投げ、リクエストを受けたウェブサーバが「はい、どうぞ」とウェブページをレスポンスで返してくれます。REST API は、このウェブページと同じように、ウェブサーバー上で提供されます。私たちが curl コマンドや、Postman や、プログラムを通じて REST API に対してリクエストを投げると、ウェブサーバ上でうごく REST API が「API の実行に成功したよ！」とか「君が求めていた情報はこれだよ！」というようにレスポンスを返してくれるのです。

本来、API はもっと広義の意味をもつ言葉ですが、本書においてはただ「API」と呼んだら、それは REST API のことを指していると思ってください。

^{*1} REST は REpresentational State Transfer の略。REST アーキテクチャスタイルという、ルールの ようなものに従って作られた API が REST API や RESTful AP と呼ばれます。

2.2 Messaging API とは

2.2.1 開発には「個人の LINE アカウント」が必須か？

Messaging API で開発をするとき、様々な設定をするための管理画面が「LINE Developers コンソール」です。この LINE Developers コンソールにログインするには、LINE アカウントまたはビジネスアカウントのどちらかが必要です。

すでにスマートフォンで LINE を使っていれば、そのまま LINE アカウントでログインできます。個人開発であればそれで問題ありませんが、仕事の場合は「私物のスマートフォンに入れている個人の LINE アカウントを業務で使うのはちょっと…」というケースももちろんあると思います。その場合は業務用のメールアドレスでビジネスアカウントを作りましょう。ただしビジネスアカウントでログインした場合は、Messaging API チャネルが作れないなど、できることに一部制限^{*2}があります。

その場合は、プロジェクトの中で誰かひとりが会社端末に LINE を入れ、その LINE アカウントで LINE Developers コンソールにログインして、Messaging API チャネルを作ります。後はそのプロバイダーやチャネルに、他のメンバーのビジネスアカウントを Admin 権限で追加してやれば、開発を進める上で問題はないでしょう。

2.2.2 LINE 公式アカウントと Bot の違い

2.2.3 チャットボットとは

チャットボットとは、リアルタイムにメッセージをやりとりする「チャット」と、人間のように動いたり働いたりする機械の「ロボット」を組み合わせた言葉です。チャットボットの裏側には人間がいるわけではなく、プログラムがメッセージの内容に応じた返信をしてくれたり、自動でメッセージを送ってきたりしています。

このチャットボットは、近年では身近な存在になったことで、単に「ボット」と呼ばれることが多いかもしれません。Twitter で地震が起きるとすぐに震度を知らせてくれる地震速報ボットや、著名人の名言を定期的につぶやくボット、特定の用語に反応してリプライしてくるボットなどを、あなたも一度は見たことがあるのではないでしょうか。会社の Slack に、GitHub の通知をしてくれるボットやメッセージを自動翻訳してくれるボットがいる、というケースもあるかも知れません。

本書では、この自動応答するチャットボットのプログラムのことを「ボット」と呼び

^{*2} LINE Developers コンソールへのログイン | LINE Developers <https://developers.line.biz/ja/docs/line-developers-console/login-account/>

ます。

2.2.4 プレミアム ID とベーシック ID

LINE 公式アカウントを作ると、最初に「ベーシック ID」というものが自動で割り振られます。@マークと、ランダムな 3 桁の数字と 5 桁の英字で構成された「@184ylqsv」みたいなものがベーシック ID です。

LINE の検索窓でこのベーシック ID を検索すると、ちゃんと LINE 公式アカウントにたどり着けるのですが、ランダムな文字列をユーザーに「184 の…わいえる…きゅー…えすぶい」のように入力せざるを得ないことが多いのです。もうちょっと名前っぽい、覚えやすい ID がほしくなりますよね。

そこでなんと、お金さえ出せば希望の「プレミアム ID」が取得できます！ 但し、プレミアム ID は一意なので、他の人が既に取得しているプレミアム ID と重複するものは取得できません。

- <https://www.linebiz.com/jp/service/line-official-account/plan/>

プレミアム ID を取得しても、ベーシック ID や、過去に作った友だち追加の QR コードは引き続き使用できますので安心してください。

2.2.5 未認証アカウントと認証済アカウント

LINE 公式アカウントには認証ステータスというものがあり、未認証アカウントと認証済アカウントの 2 種類に分けられます。

- <https://www.linebiz.com/jp/service/line-official-account/verified-account/>

認証済アカウントになると、青色のバッジがついて「公式っぽさ」が増すと共に、LINE 内でアカウント名や概要が検索対象^{*3}になり、検索結果に表示されるようになります。^{*4}

逆に言えば、未認証アカウントであれば、友だち追加の QR コードや、ベーシック ID を自ら流出させない限り、知らない人が勝手に友だち追加される可能性は低いです。

^{*3} 未認証アカウントの場合、アカウント名や概要は検索に引っかかりませんが、ベーシック ID やプレミアム ID で検索した場合はしっかりと検索結果に表示されます。

^{*4} 厳密に言えば、認証済アカウントになると検索結果に表示させるのか、非表示にしておくのかを選べるようになります。たとえば「お金を払って謎解きゲームのチケットを買った人だけ、LINE 公式アカウントと友だちになってゲームに参加できる」というようなことを実現したい場合は、検索結果には表示されないよう「非表示」を選んでおきましょう。 <https://www.linebiz.com/jp/manual/OfficialAccountManager/tutorial-step5/>

【コラム】開発中の LINE 公式アカウントをリリース日まで非公開にしておける？

開発中の LINE 公式アカウントが人目に触れてしまうことのないよう、リリース日まではアカウントそのものを非公開にしておきたい！と思ったとします。そんなことは可能なのでしょうか？

残念ながら LINE 公式アカウントには「非公開」と「公開」、あるいは「開発中」と「リリース済み」というような状態管理がありません。作られた瞬間から、ベーシック ID やプレミアム ID さえ分かれば、リリース前だろうがなんだろうが誰でも友だち追加できてしまいます。

リリース前や開発用の LINE 公式アカウントは、うっかりベーシック ID やプレミアム ID、そして友だち追加の URL などを外部に露出させないよう注意しましょう。

また認証済アカウントにした場合は、「検索結果での表示」をデフォルトの「非表示」のままにしておきましょう。

【コラム】開発中の LINE 公式アカウントに知らない人が友だち追加されてしまった。ブロックできる？

個人的な開発用途として LINE 公式アカウントを作っていたのに、うっかり友だち追加の URL を Twitter に書いてしまい、知らない人に友だち追加されてしまった。まだ開発途中なので、あれこれ見られたくない……LINE 公式アカウント側から、友だちをブロックすることはできるのでしょうか？

結論から言うと、LINE 公式アカウント側から友だちをブロックしたり、削除したりすることはできません。

【コラム】開発チームだけにメッセージのテスト配信がしたい

たとえば本番用の LINE 公式アカウントと、テスト環境用の LINE 公式アカウントを別々に用意していて、開発チームの A さん、B さん、C さんがテスト環境用の LINE 公式アカウントと友だちになっていたような場合、C さんが退職してしまうと、A さん B さん側で C さんをテスト環境用の LINE 公式アカウントの友だちからキックするような方法がありません。退職前に C さんに頼んで、テスト環境用の LINE 公式アカウントをブロックしてもらうしかないので。

さらに厳密に言うと、一度対面であって目の前でブロックの操作をしてもらつたとしても、ブロックは解除できるので、悪意のある C さんがブロックを解除してしまうことは止められません。

なので一度友だち追加されたら、開発者側からは友だち状態はコントロールできないという前提で、色々仕様を組んでおく必要があります。

たとえば、テスト配信はテスト環境用の LINE 公式アカウントから送るとしても、ブロードキャストメッセージで全員に送るのでなく、開発チームの A さん B さん C さんのユーザー ID を指定したマルチキャストメッセージを送る、という方法がオススメです。これなら C さんの退職後、開発チームに残った A さん B さんが送信対象から C さんのユーザー ID を消せば、C さんには公開前のテストメッセージが届かなくなります。

2.2.6 プロバイダーとは

2.2.7 チャネルとは

2.2.8 アカウントの管理方法

複数人で開発するとき

2.3 LINE 公式アカウントの運用方法

2.4 LINE キャンパスで資格を取ってみよう

2.5 ツール編

2.5.1 LINE Official Account Manager で運用する

2.5.2 LINE 公式アカウント管理アプリで運用する

2.5.3 Messaging API で運用する

2.5.4 非公式の管理画面で運用する

2.6 返信方法編

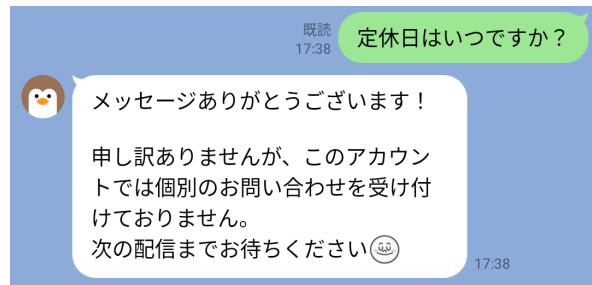
さて、ここまででは「LINE 公式アカウントからメッセージを送る」という、LINE 公式アカウント起点の話をしてきました。LINE 公式アカウント起点というのは、たとえば今月の定休日を知らせるメッセージを送るとか、新商品が出たときにメッセージを送るとかいうように、LINE 公式アカウント側が「送りたい！」と思った任意のタイミングでメッセージを送る、ということです。

でも片方が話したいときだけ一方的に話すのでは、それは対話とは言えません。ユーザーが友だち追加してくれたときや、ユーザーが何か質問してきたとき、ユーザーがリッチメニューをタップしたときなど、ユーザー起点のコミュニケーションにきちんと反応できるようになってこそそのボットと言えます。

LINE 公式アカウントで、ユーザーからメッセージに応対する方法は、大きく分けて 4 つあります。

2.6.1 方法 1. 固定の自動応答を設定しておいて個別の応対は一切しない

1つめは、LINE Official Account Manager で次のような「応答メッセージ」を設定しておいて、個別の応対は一切しないという方法です。(図 1.19)



▲図 2.1 自動の応答メッセージ

この自動の応答メッセージはデフォルトで設定されているので、色々な LINE 公式アカウントでよく似たメッセージを見たことがあるかもしれません。

これだと個別対応のコストが一切要らなくなるものの、ユーザー起点のコミュニケーションはできなくなるので、一方的な宣伝をするだけの場所になります。ただ、少なくとも話しかけても何も応答せずに無言でいられるよりは、まだ気分がいいかもしれません。

2.6.2 方法 2. 人間が手打ちのチャットで返信する

2つめは「中の人」が頑張って手打ちのチャットで返信する、という方法です。「中の人」が LINE Official Account Manager や管理アプリでユーザーからのメッセージを確認して、手打ちのチャットで頑張って応対します。

2.6.3 方法 3. 内容に応じた自動応答メッセージで応対する

3つめは内容に応じた自動応答メッセージを用意しておいて自動で返信する、という方法です。

LINE Official Account Manager や管理アプリには、応答メッセージと AI 応答メッセージというものがあります。応答メッセージはキーワードとメッセージを事前に設定しておくことで、ユーザーがそのキーワードと完全一致するメッセージを送ってきたら、自動でメッセージを返信する機能です。完全に一致する必要があるので、たとえば「メ

「メニュー」というキーワードの場合、ユーザーが「メニュー」とだけ送ってくれば応答できますが、「メニューは？」と送ってきた場合は応答できません。

一方、AI 応答メッセージはもう少し融通が利いて、ユーザーから送られたメッセージの内容を AI が判別して、適切なメッセージで自動で返信してくれます。たとえば飲食店なら、営業時間、住所、Wi-Fi、コンセント、駐車場、禁煙・喫煙など、質問が想定されることについてそれぞれ事前に返信メッセージを設定しておくことで、単語が完全に一致しなくてもよしなに回答してくれます。たとえば「Wi-Fi」に対して「申し訳ありませんが Wi-Fi はご利用いただけません」という回答をオンにしておくと、「Wi-Fi ってありますか？」「無線ってありますか？」「無料のわいふあいって使える？」といったメッセージでもきちんとその回答が返ってきます。

応答メッセージと AI 応答メッセージは、どちらか片方だけ使うこともできますし、キーワードに完全一致したら応答メッセージを返して、そうでない場合は AI 応答メッセージを返す、というように両方使うこともできます。

さらに方法 2 と方法 3 を併用することも可能です。LINE Official Account Manager や管理アプリで営業時間を設定しておくことで、営業時間内は中の人的手打ちのチャットで応答して、営業時間外は自動応答のメッセージに任せる、というような運用ができます。

2.6.4 方法 4. Messaging API で返信する

Messaging API では、この「ユーザー起点のコミュニケーション」に気づく方法として、Webhook というものがあります。Webhook とはいいったいどんなものなのでしょう？

【コラム】LINE 公式アカウントの「既読」はいつ付くのか？

普通の友だちとの LINE のメッセージは、相手がトークルームを開いてメッセージを見ることで「既読」がつきます。では LINE 公式アカウントの場合は、「既読」はいつ付くのでしょうか？

LINE Official Account Manager や管理アプリの応答設定からチャットをオンにしていると、中の人者がチャットの画面でメッセージを確認するまでは、LINE 公式アカウント側でメッセージを読んだことを示す「既読」マークがつきません。逆にチャットをオフにしていると、メッセージが送られた瞬間に自動的に「既読」がつきます。

Messaging API を使って自動応答しているのに、なぜか「既読」マークが付かない！ というときは、うっかり応答設定でチャットをオンにしないか確認してみてください。応答設定で「ボットだけをオンにしている」だと自動で既読が付きますが、「チャットだけをオンにしている」もしくは「チャットとボットの両方をオンにしている」状態だと、チャットの画面でメッセージを確認するまで既読が付きません。

チャットもボットもオンにしたいけど、チャット画面で確認するんじゃなくて自動で既読を付けたい、という場合は「既読 API」というものを叩くことで既読が付けられますが、残念ながら既読 API は法人ユーザー限定のオプション機能です。

- <https://developers.line.biz/ja/docs/partner-docs/mark-as-read/>

2.6.5 Webhook とは

通常、私たちはブラウザで URL を入力したり、リンクをクリックしたりすることで「このウェブページを見せてくれ！」とウェブサーバにリクエストを投げ、リクエストを受けたウェブサーバが「はい、どうぞ」とウェブページをレスポンスで返してくれます。このとき、ウェブサーバ側が「サーバ」で、自分やブラウザ側が「クライアント」です。

Webhook を使うとき、私たちは自分でサーバを用意する必要があります。このサーバは Webhook を受け取るためのサーバでボットサーバーと呼ばれます。Webhook を投げてくる LINE プラットフォームは「クライアント」です。LINE プラットフォームから「Webhook だよ～！ 受け取って～！」というリクエストが来るので、ボットサーバはそれを受け取って「ちゃんと受け取ったよ！ ステータスコード 200！」というレスポンスを返さなければいけません。

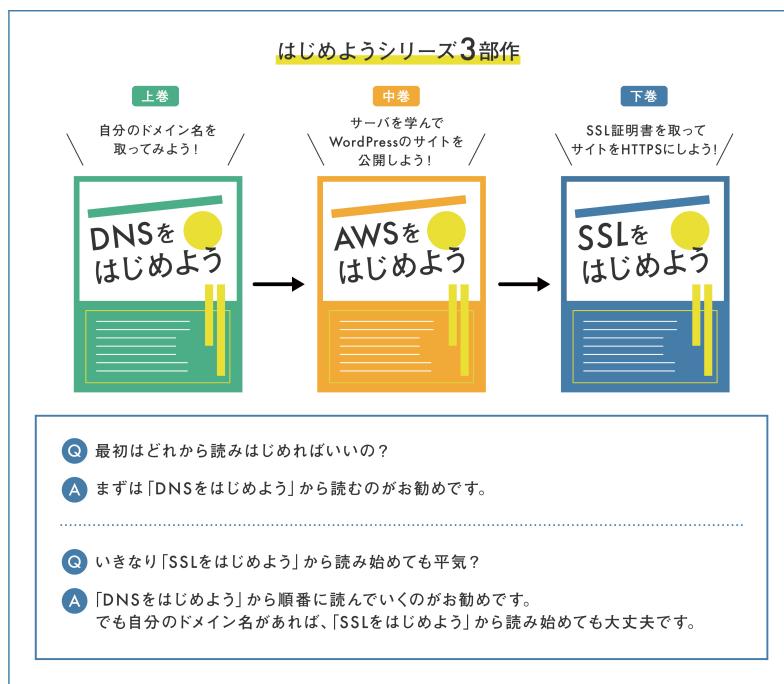
私たちはクライアントの経験はよくありますが、サーバの立場はあんまり経験がないのでなんだか不思議な感じがしますねが、こと Webhook においては、あなたは「ボットサーバを用意する側」なのです。

2.7 用意するもの

2.7.1 ポットサーバー

今回は Webhook を受け取って、200 を返して、応答メッセージを送るポットサーバーとして、AWS のサーバーレスサービス、AWS Lambda を使用します。

なお AWS アカウントの作成方法は、はじめようシリーズ3部作の「DNS をはじめよう」(図 1.20)*5で、また AWS とは何かについての説明や、無料利用枠の範囲、利用金額が一定額を超えたらアラートが飛ぶようにする設定などは「AWS をはじめよう」で詳しく紹介しています。まだ AWS のアカウントをお持ちでない場合は、そちらを参考にしてください。



▲図 2.2 はじめようシリーズ 3 部作

*5 <https://mochikoastech.booth.pm/>

2.7.2 エラーの統計情報

2.7.3 ポットサーバーが Webhook を受け取れなかったときの再送機能

たとえば LINE 公式アカウントが急に有名になり、一気に友だちが増えて、メッセージが大量に送られてきたとします。急なアクセス集中でサイトが落ちるよう、LINE プラットフォームから Webhook が飛んでくるリクエストによって、ポットサーバーが過負荷になって応答できなくなってしまったどうなるのでしょうか。

そんなときのために「Webhook の再送^{*6}」という機能があります。Webhook の再送はデフォルトではオフになっていますが、LINE Developers コンソールでオンにすることで、ポットサーバーが Webhook を受け取れなかったときに再送してくれるようになります。

ここで「Webhook を受け取れなかった」と判断されるのかというと、LINE プラットフォームから見て、「Webhook を受け取ってー！」というリクエストに対して、きちんとステータスコード 200 が返ってこなければ「Webhook を受け取れなかった」と判断されます。

ただし、ポットサーバーから見て「Webhook を受け取ってー！」というリクエストに対して、きちんとステータスコード 200 を返したつもりでも、ポットサーバーから LINE プラットフォームまでのネットワーク経路上で何か問題があって、きちんとレスポンスが LINE プラットフォームまで到達しないケースは考えられます。

その場合、Webhook の再送をオンにしたことで次のようなトラブルが考えられます。

1. ユーザーが『購入する』というメッセージを送信する
2. LINE プラットフォームからポットサーバーに『購入する』というメッセージの Webhook が飛ぶ
3. ポットサーバーで Webhook を受け取ってステータスコード 200 を返し、商品購入処理を行った上でユーザーに「購入完了しました！」という応答メッセージを送る
4. しかし何らかの理由でステータスコード 200 のレスポンスが LINE プラットフォームに到達しない
5. LINE プラットフォームは「Webhook を受け取れなかった」と判断して、『購入する』というメッセージの Webhook を再送する
6. ポットサーバーで Webhook を受け取ってステータスコード 200 を返し、商品購入処理を行った上でユーザーに「購入完了しました！」という応答メッセージを送る

^{*6} 受け取りに失敗した Webhook を再送する | LINE Developers <https://developers.line.biz/ja/docs/messaging-api/receiving-messages/#webhook-redelivery>

7. ユーザーは一度『購入する』と送っただけなのに、商品が2つ購入されてしまい、購入完了メッセージも2度届いて驚く

これを防ぐためには、商品購入処理を行う際に、Webhook のイベントごとに一意な値である webhookEventId を確認して、「既に購入処理を行った Webhook イベントと同一のイベントではないか？」を確認する必要があります。

Webhook の再送は便利な機能ですが、このように意図しない再送が行われたときのことを考えた重複チェックの処理がなければ、迂闊にオンにするべきではありません。

【コラム】Webhook へのレスポンスが先？ 応答メッセージが先？

Webhook を受け取って、その内容に応じた処理や応答メッセージの送信などをしてからステータスコード 200 を返す、という順番はオススメしません。

間に挟まる処理や応答メッセージ送信などの処理に時間がかかった場合に、レスポンスを返す前に Webhook のリクエストがタイムアウトしてしまう可能性があるためです。

なんとなく先に色々片付けてからレスポンスを返したい気持ちは分かりますが、郵便局員が郵便物を持って来てくれたとき、わざわざ玄関前で待たせて封筒を開けて中身を読んで、読み終わってから受領の印子を押すのはおかしいよね、と考えると納得しやすいのではないでしょうか。まずは受領の印子を押して郵便局員を帰らせてあげて、手紙を読んだり、返事を書いたらはそのあとでやりましょう。

【コラム】これは本当に LINE プラットフォームから来た Webhook ?

ポットサーバーに、LINE プラットフォームから Webhook が届いたら、その内容に応じて返信を送ったり、何か処理をしたりします。

このとき、届いたリクエストが本当に LINE プラットフォームから届いた Webhook なのか、それとも LINE プラットフォームを装った第三者からの攻撃な

のか、どうやって判別すればよいのでしょうか？

アクセス元の IP アドレスが分かれば、IP アドレス制限をかけることで、LINE プラットフォーム以外からのアクセスを遮断できますが、残念ながら LINE プラットフォームは IP アドレスのレンジを開示していません。^{*7}代わりに、署名の検証を行いましょう。

LINE プラットフォームから届く Webhook には、そのリクエストヘッダーに必ず「x-line-signature」という署名が含まれています。Messaging API チャネルのチャネルシークレットを秘密鍵として、届いた Webhook のリクエストボディのダイジェスト値を取得し、さらにそのダイジェスト値を Base64 エンコードした値と、リクエストヘッダーの x-line-signature の署名の一致を確認できれば、これが本当に LINE プラットフォームから届いた Webhook である、というセキュリティの担保ができます。

この検証をしないで「テキストメッセージの Webhook イベントが届いたらユーザーに返信する」のような処理をしていると、LINE Bot を介して特定のユーザーに大量のメッセージを送りつける攻撃などが可能になってしまいます。

署名検証の各言語ごとのコードサンプルは、公式ドキュメントの「署名を検証する^{*8}」にあります。

2.8 メッセージを送る

2.8.1 友だち追加されたときのあいさつメッセージ

2.8.2 友だち全員にメッセージを一斉配信する

【コラム】URL を送る前に OGP の見た目を事前に確認したり、キャッシュを消したりしたい

^{*14} Webhook | LINE Developers <https://developers.line.biz/ja/reference/messaging-api/#webhooks>

^{*15} 署名を検証する | LINE Developers <https://developers.line.biz/ja/reference/messaging-api/#signature-validation>

LINE で URL を含むメッセージを送ると、こんなふうにプレビューが表示されます。(図 1.21)



▲図 2.3 URL のプレビュー

実際にメッセージを送る前に、このプレビューでどんな画像やテキストが表示されるのか、確認したかったらどうすればいいのでしょうか？

実は PagePoker という公式のツールを使うと、対象ページの OGP タグ^{*9}を読み込んで、どんなふうにプレビューが表示されるのかを事前に確認できます。^{*10}

- Pagepoker <https://poker.line.naver.jp/>

「Clear Cache」にチェックを入れることで、LINE サーバー側のキャッシュも削除できるということなので、ウェブサイト側で OGP の画像を差し替えた後にここでキャッシュを削除すれば、「URL を投げたらうっかり古い画像がプレビューで表示されてしまった」という事態も避けられます。

OGP タグの書き方については、LINE Developers サイトの「トークと LINE VOOM 内の URL プレビューはどのようにして生成されますか？」を参考してください。

- <https://developers.line.biz/ja/faq/#how-are-the-url-previews-generated>

2.8.3 特定の属性を指定してメッセージを送る

2.8.4 ユーザー ID を指定して特定の人に送る

2.8.5 メッセージ送信元のアイコンと表示名を変更する

2.8.6 無料通数を超えたときに自動で追加課金されるのか

2.9 メッセージを Webhook で受け取る

2.9.1 画像などのメディアファイルを指定して取得する

2.10 メッセージ以外のイベントを Webhook で受け取る

2.10.1 友だちになったとき

2.10.2 ブロックされたとき

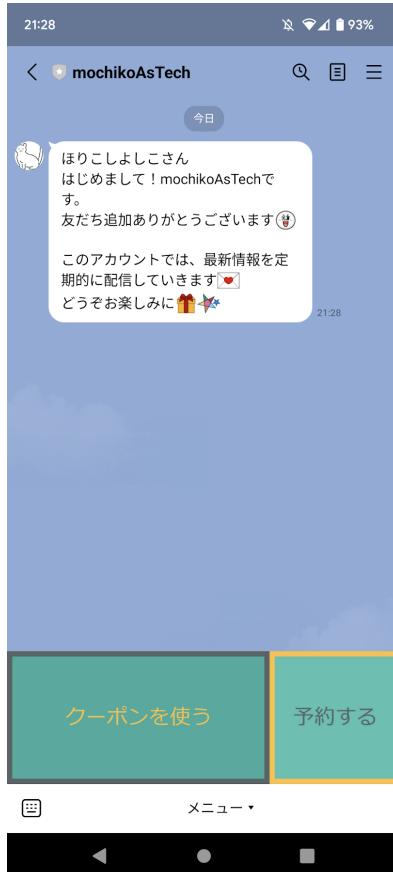
2.10.3 ユーザーが何かをタップしたとき

2.11 リッチメニュー

LINE 公式アカウントと友だちになると、トーク画面の下部にこんなメニューが表示されます。テキストだけでなく、画像でできた贅沢な多彩なメニュー…つまりリッチなメニュー…リッチなメニューなので…リッチメニュー！ そう、これがリッチメニューです。わあ、リッチ！（図 1.22）

*¹⁶ OGP は Open Graph Protocol の略です。HTML にメタデータとして「og:image」のような OGP タグを書いておくことで、Twitter や LINE などでその URL を共有したときに、URL そのままではなく対象ページのタイトルや概要、画像などがカードのように表示されます。

*¹⁷ Twitter の Card Validator とか、Facebook のシェアデバッガーみたいなものですね。Card Validator は気づいたらレビュー確認機能がなくなっていたけれど。<https://cards-dev.twitter.com/validator> <https://developers.facebook.com/tools/debug/>



▲図 2.4 リッチメニュー

このリッチメニューは、一番下の「メニュー」と書いてあるバーをタップすることで閉じたり開いたりできます。このバーのテキストも、デフォルトの「メニュー」から「メニューはこちら」「お問い合わせの入力はこちら」「ほっとするブレイクタイムを」「会員証」「お役立ち情報はこちら」「人気ランキングをチェック！」「こちらもチェック」「荷物の追跡・再配達・集荷受付」といった様々なテキストに変更できます。

リッチメニューの実態は1枚の画像であり、画像内の各領域にそれぞれリンクを設定することで、公式サイトや予約ページ、特定の機能などにユーザーを誘導できます。

2.12 リッチメニュープレイグラウンドでリッチメニューを体験してみる

リッチメニューの様々な機能は、文字で説明するより体験してみるのが一番分かりやすいです。公式で提供されているリッチメニュープレイグラウンドという、「リッチメニューを体験するための LINE 公式アカウント」と友だちになってみましょう。(図 1.23)



▲図 2.5 リッチメニュープレイグラウンドと友だちになる

「リッチメニューを開く」をタップすると、「メッセージアクションを試す」と表示されます。これは「ユーザーがリッチメニューをタップすることで、特定のメッセージをユーザーから自動送信させる」という機能です。早速試してみましょう。

あなたが入力した訳でも送った訳でもないのに、「message sent successfully!」というメッセージがあなたから送信されました。これがメッセージアクションです。メッセージアクションによって、ユーザーから LINE 公式アカウントに対してメッセージが送られ、それによってこんな Webhook がボットサーバーに届きましたよ、というのを解説してくれています。

たとえば「再配達の申し込みをしたかったら、LINE で『再配達』というメッセージを送ってね」のように、ユーザーに手入力を促さなくても、リッチメニューに「再配達の申し込み」というボタンを用意して、そのボタンの領域をタップしたら「再配達」というメッセージが自動送信されるよう、対象の領域にメッセージアクションを設定しておけばいいのです。

2.13 リッチメニューの画像を作る

LINE Official Account Manager でリッチメニューの作成画面を開くと、「デザインガイド」というボタンがあり、そこからリッチメニューのテンプレート画像がダウンロードできます。本書ではこのテンプレートをそのまま使ってリッチメニューを設定してみます。

リッチメニューの画像は、LINE Official Account Manager で作成することも可能ですし、Canva などで自作しても構いません。サイズの制約は横幅が 800px 以上 2500px 以下、高さが 250px 以上となっています。参考までに 3 マスが 2 段重ねで 6 マス分のテンプレート画像だと横幅が 2500px、高さが 1686px で作られていました。

2.13.1 LINE Official Account Manager と Messaging API でのリッチメニューの制約の違い

リッチメニューは Messaging API で頑張って JSON を組み立てて作らなくても、実は LINE Official Account Manager という管理画面上で、GUI でぼちぼち作ることも可能です。

前述のあいさつメッセージのように、GUI の LINE Official Account Manager と、CUI の Messaging API は、どちらか一方しか使えないというものではなく、リッチメニューは LINE Official Account Manager で設定し、メッセージの送信は Messaging API を使う、というように機能によって使い分けが可能です。

ただし、LINE Official Account Manager では大が 7 種類、小が 5 種類という特定のサイズと形からしか選べません。また全員に同じリッチメニューを出す機能しかないので、会員証を持っている人にだけ会員向けのリッチメニューを出す、といった出し分けもできません。Messaging API であれば、縦横のサイズや、それぞれの領域も自由に設定でき、ユーザーごとに別々のリッチメニューを表示させることも可能です。Messaging API なら、タブ切り替えのようなリッチメニューを作ることもできます。

限定された機能で構わないので GUI の管理画面からラクに設定したいなら LINE Official Account Manager を使えばいいし、色んな機能を使い倒して限界まで自由にやりたいのであれば、Messaging API を使って自分だけの管理画面を作るのがオススメです。

2.13.2 リッチメニューを設定する

リッチメニューは 3 つのステップで設定します。

1. リッチメニューを作る
2. そのリッチメニューに画像をアップロードする
3. それをデフォルトのリッチメニューとして設定する

2.14 リッチメニューの設定の種類

2.15 3つの設定方法と表示の優先順位

2.15.1 OA Manager で設定したリッチメニュー

2.15.2 Messaging API で設定したリッチメニュー

2.15.3 ユーザー単位のリッチメニュー

2.16 どんな見た目のメッセージが送れるのか

2.16.1 テキストメッセージ

2.16.2 スタンプメッセージ

2.16.3 画像メッセージ

LINE Creative Lab を使うと、画像メッセージとして送るような素敵なクリエイティブが簡単に作れます。

- LINE Creative Lab
 - <https://creativelab.line.biz/>

送れるスタンプの種類

自社のスタンプを使う方法

2.16.4 Flex Message

2.16.5 Flex Message Simulator は便利

でも Flex Message Simulator で出力される JSON は、messages 直下の Flex Message 全体じゃなくて、コンテナ (contents 以下) なので要注意です。あとログインが必要です。

- <https://developers.line.biz/flex-simulator/>

2.17 やりとりする場所

2.17.1 ユーザーとのトーク

2.17.2 グループトーク

2.17.3 複数人トーク

2.18 Messaging API 以外のプロダクトとの組み合わせ

2.18.1 LINE Social Plugins

自分のサイトに「友だち追加」ボタンを設置してみよう

2.18.2 LINE VOOM

2.18.3 LINE ログイン

2.18.4 LIFF や LINE ミニアプリ

2.18.5 プロバイダーページ

2.19 開発を始める

2.20 Messaging API SDK を使おう

2.20.1 Java

2.20.2 PHP

2.20.3 Go

2.20.4 Perl

2.20.5 Ruby

2.20.6 Python

- <https://github.com/line/line-bot-sdk-python>

1. mkdir python

2. cd python
3. pip install line-bot-sdk -t . --no-user
4. SDK のコードが詰まった python ディレクトリを zip ファイルに圧縮する
5. AWS Lambda で「レイヤーを作成」
6. zip ファイルを選択
7. 互換性のあるランタイムで Python 3.9 を選択する
8. x86_64 にチェックを入れる
9. カスタムレイヤーで選ぶ
10. API タイプは HTTP API で API gateway 作る

ディレクトリを line-bot-sdk-python にしてたらこれでひっかかった* <https://aws.amazon.com/jp/premiumsupport/knowledge-center/lambda-import-module-error-python/>

1. OpenAI でアカウント作る
2. SECRET_KEY を取得する
3. mkdir python
4. cd python
5. pip install openai -t . --no-user

ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the source of the following dependency conflicts.
line-bot-sdk 2.4.1 requires aiohttp==3.8.3, but you have aiohttp 3.8.4 which is incompatible.
line-bot-sdk 2.4.1 requires requests==2.28.1, but you have requests 2.28.2 which is incompatible.
Successfully installed aiohttp-3.8.4 aiosignal-1.3.1
async-timeout-4.0.2 attrs-22.2.0 certifi-2022.12.7 charset-normalizer-3.1.0
colorama-0.4.6 frozenlist-1.3.3 idna-3.4 multidict-6.0.4 openai-0.27.2 requests-2.28.2
tqdm-4.65.0 urllib3-1.26.15 yar

pip check No broken requirements found. なので無視してそのまま行く

2.20.7 Node.js

2.21 コミュニティに参加しよう

2.21.1 LINE Developers community

あとがき

あとがき。ねこはかわいい。

2023年5月
mochikoAsTech

PDF 版のダウンロード

本書（紙の書籍）をお買い上げいただいた方は、下記の URL から PDF 版を無料でダウンロードできます。

- ダウンロード URL
 - <https://mochikoastech.booth.pm/items/xxxxxx>
- パスワード
 - *****

Special Thanks:

- 謝辞を捧ぐみなさんのお名前

レビュアー

- レビュアーのお名前

参考文献

- はじめてでもできる！ LINE ビジネス活用公式ガイド - LINE 株式会社
 - <https://book.impress.co.jp/books/1121101033>
- LINE BOT を作ろう！ Messaging API を使ったチャットボットの基礎と利用例 - 立花 翔
 - <https://techbooster.booth.pm/items/586727>
- LINE API 実践ガイド - LINE API Expert 認定メンバー
 - <https://book.mynavi.jp/ec/products/detail/id=117310>
- Hands-on LINEBOT - しんぶんぶん
 - <https://techbookfest.org/product/3EUnJ5WbexvCDyMgSJS1qb>
- Real World HTTP - 渋川 よしき
 - <https://www.oreilly.co.jp/books/9784873118048/>

著者紹介

mochiko / @mochikoAsTech

テクニカルライター。元 Web 制作会社のインフラエンジニア。ねこが好き。「分からない気持ち」に寄り添える技術者になれるように日々奮闘中。技術書典で頒布した「DNS をはじめよう 改訂第 2 版」「AWS をはじめよう 改訂第 2 版」「SSL をはじめよう」の「はじめようシリーズ 3 部作」は累計で 11,000 冊を突破。

- <https://twitter.com/mochikoAsTech>
- <https://mochikoastech.booth.pm/>
- <https://note.mu/mochikoastech>
- <https://mochikoastech.hatenablog.com/>
- <https://www.amazon.co.jp/mochikoAsTech/e/B087NBL9VM>

Hikaru Wakamatsu

目次、挿絵デザインを担当。

Shinya Nagashio

表紙、章扉デザインを担当。

LINE Botをつくってみよう

APIを試して学んでしっかりわかる

2023年5月20日 技術書典14 初版

著者 mochikoAsTech

デザイン Hikaru Wakamatsu / Shinya Nagashio

発行所 mochikoAsTech

(C) 2023 mochikoAsTech