

# FileMakerで クリニック



## CLARIS CONNECT

小児科領域のクラウドサービスと、医療DXのための電子カルテの部品のホームページ

(LineやConnectなしでは不要)

LINEで設定 (こちらをお使いください) (旧 20230531終了 Chatworkで設定)

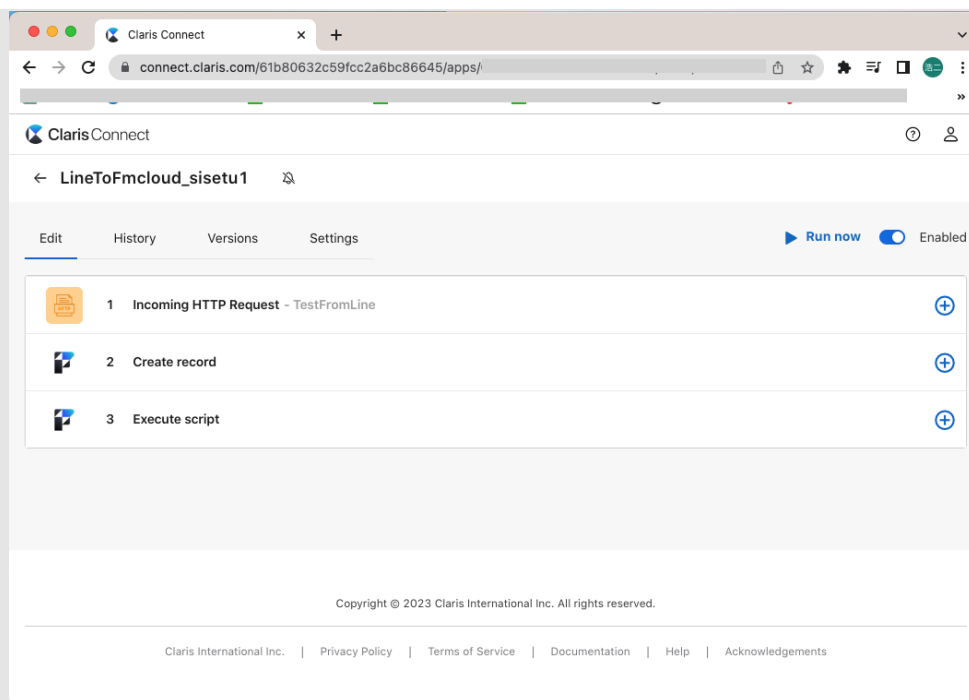
### LINEで設定 (現行)

- [LINEtoFmcloudSisetu1](#)
- [FmcloudToLine](#)
- [TestVacproFromGoogleToFMcloud](#) (これは上記Chatworkと同じ設定)

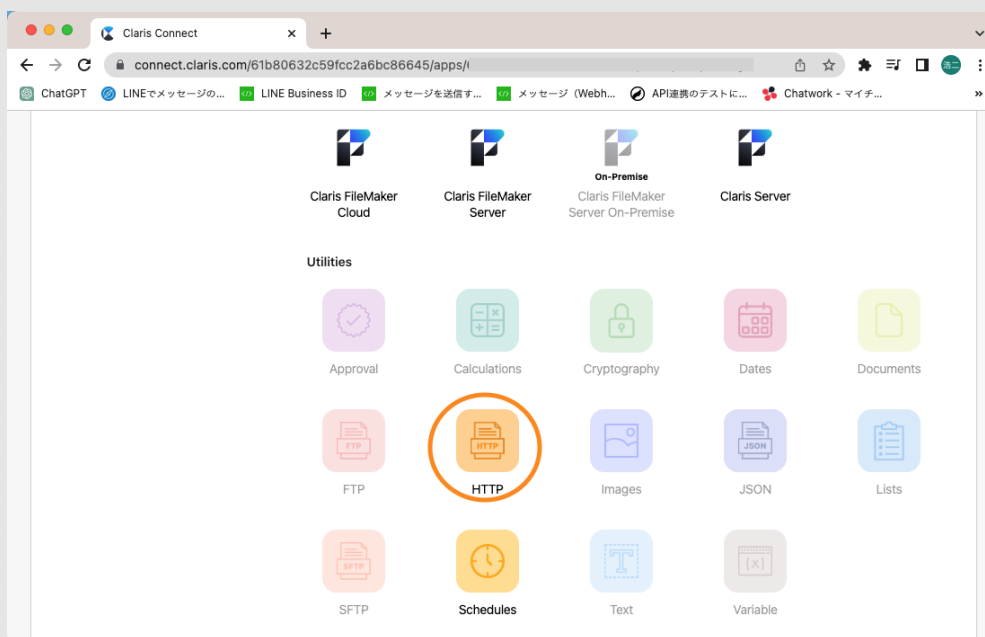
チャット1回のやり取りにAPIのステップが10個消費されます。

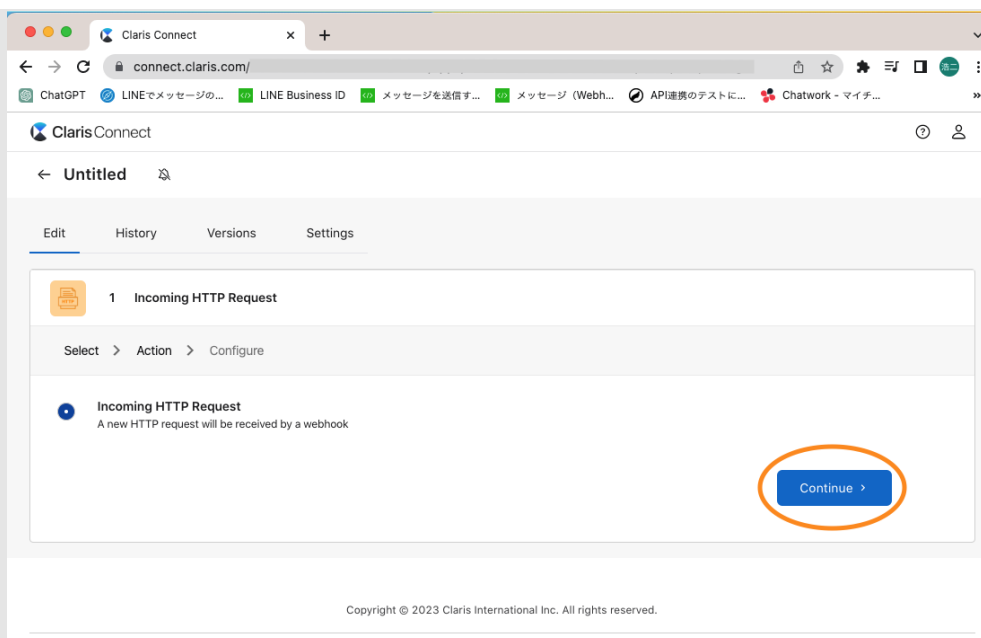
<LINEtoFmcloudSisetu1> (APIのステップは3個)

LineToFmcloud\_sisetu1

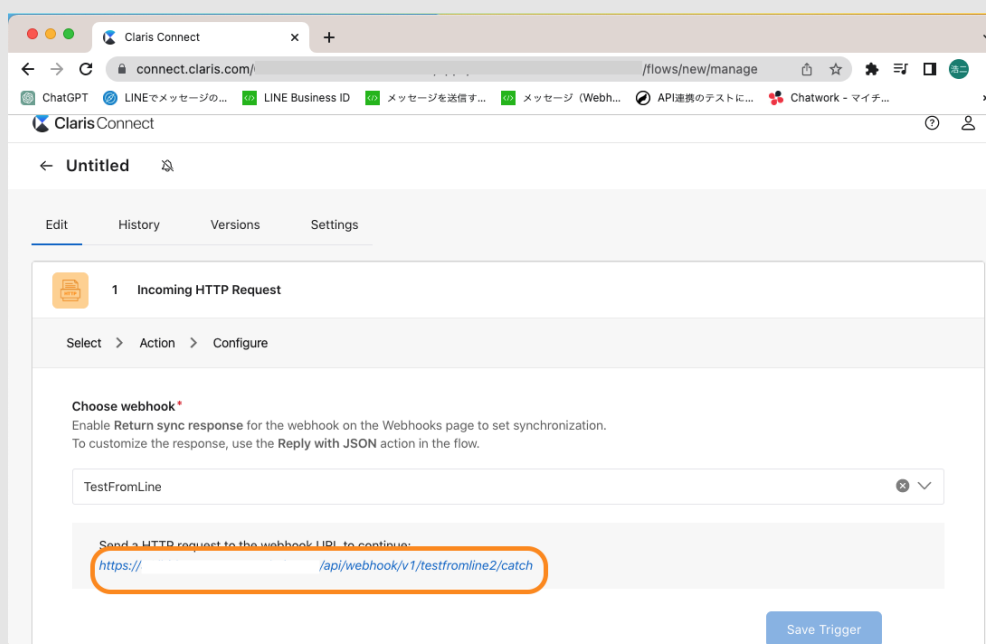


LINE にメッセージを出すとAPIが動作してFileMakerにデータが入るようにします。  
まずWebhookを作成します。

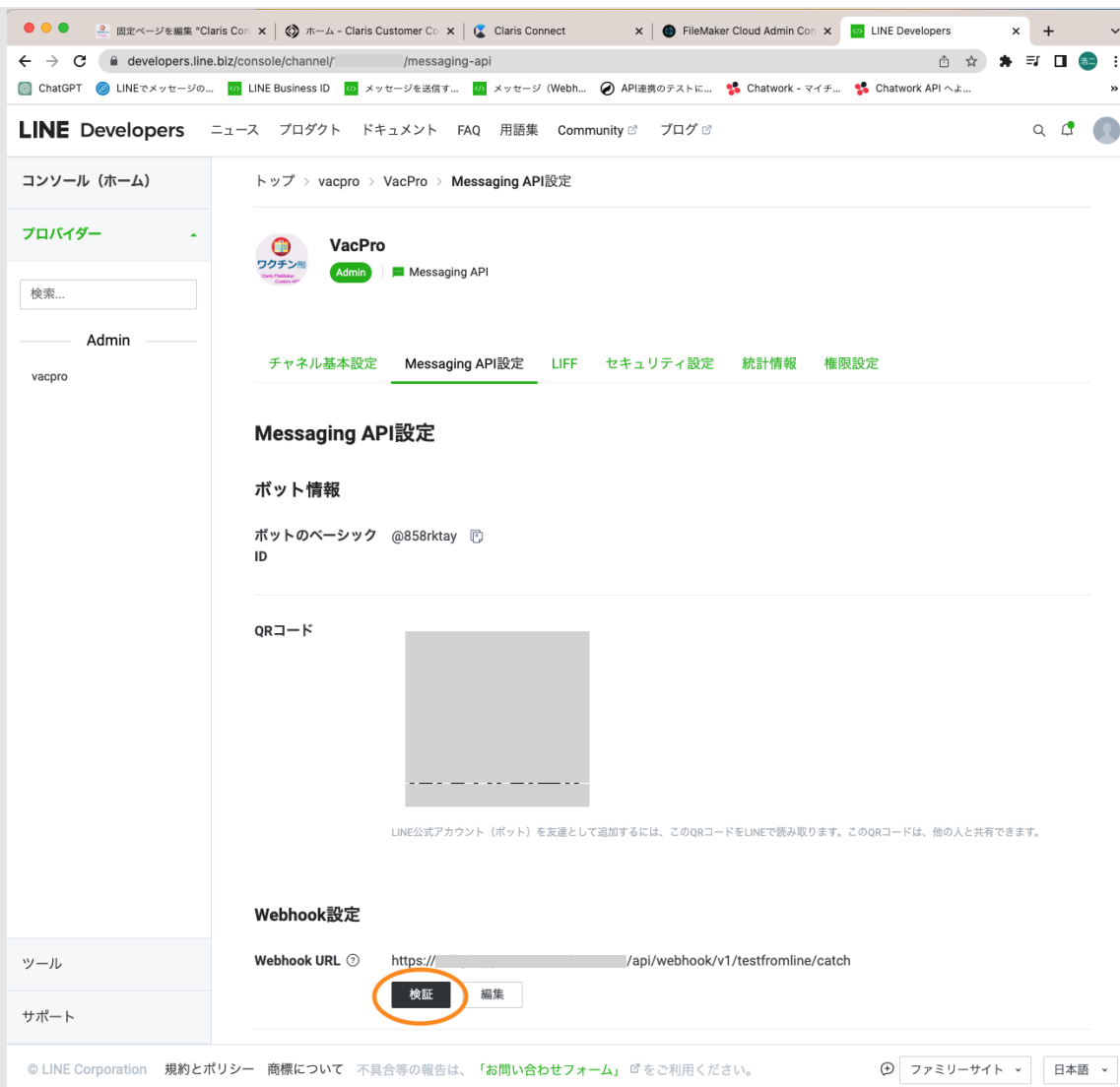




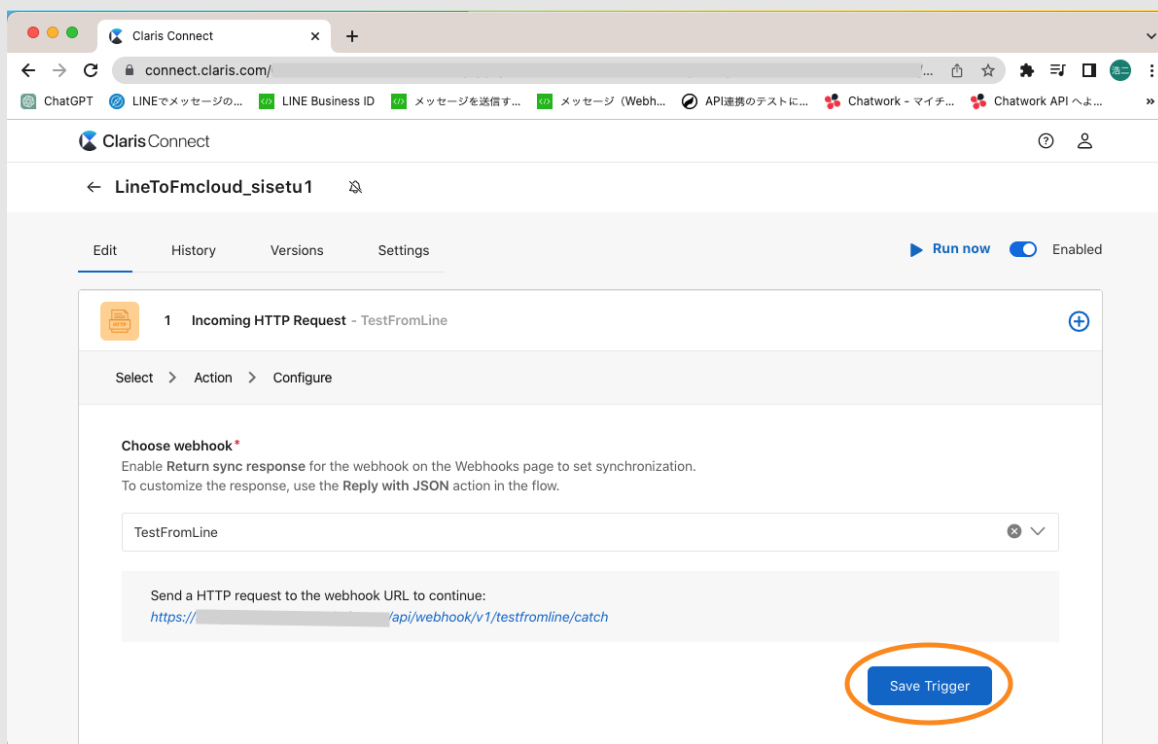
webhook URLをコピーします。まだボタンはグレー表示です。Flow名を、LineToFmcloud\_sisetu1としました（適宜）。



次にLine Developersの作成したプロジェクトのMessaging API設定を開きます。下図のWebhook設定のWebhook URLに上記コピーしたURLを貼り付けて、検証ボタンを押してください。



Save Triggerがアクティブになっていますので押します。



FileMaker Cloudで「Create record」を選択して下記の項目を選択します。

※現在はFileMaker Cloud はUTC時間で9時間遅れの場所にありますので、TO\_ME\_send\_timeやTO\_ME\_update\_timeは入力しないでください。日付ソートなどで不具合を起こし

ます。

- 1 Incoming HTTP Request - TestFromLine
- 2 Create record

Select > Action > Account > Configure

Layout

Review and Test

Layout name

Chatwork

TO\_ME\_from\_account\_id

Step 1 body.events.[0].source.userId

speak

ChatworkKaijyou\_施設番号会場番号::会場\_名

施設番号

1

会場番号

1

TO\_ME\_to\_account\_id

Step 1 body.destination

TO\_ME\_room\_id

Step 1 body.destination

TO\_ME\_message\_id

Step 1 body.events.[0].message.id

TO\_ME\_body

Step 1 body.events.[0].message.text

TO\_ME\_send\_time



離れて、tokenまで選択してSaveします。

時差なしシリアル



token



Data1



Data2



Data3



Save

次に「Execute script」を選択します。

Claris Connect



← LineToFmcloud\_sisetu1

Edit History Versions Settings

▶ Run now  Enabled

	1 Incoming HTTP Request - TestFromLine	
	2 Create record	
	3 Execute script	

Scriptは「Chatwork\_BOT\_予約プロセス\_Sisetu1」を選択して、parameterに下図のように縦棒!ではさんだものを入力します。

Clariss Connect ?

← LINEtoFmcloud1

Edit History Versions Settings ▶ Run now Enabled

- 1 Incoming HTTP Request - WebhookLine
- 2 Create record
- 3 Execute script

Select > Action > Account > Configure

Layout\* Review and Test

Layout name

Chatwork

Script\* Review and Test

Script name

LINE\_BOT\_予約プロセス\_Sisetu1

Script parameter

Enter a script parameter. Only strings are supported.

Step 1 body.events.[0].source.userId | Step 1 body.events.[0].message.text | Step 1 body.events.[0].message.id | Step 1 body.destina

Save

|body.events.[0].source.userId|body.events.[0].message.text|body.events.[0].message.id|body.destination|

## <FmcloudToLine> (APIのステップは7個)

全体像です。FileMaker Cloud をトリガーとして、その後自動コメントの改行の数により分岐するようにしています。LINEはオブジェクトが5個までの制限があるので5分岐になっています。

Clariss Connect

← UrlInsertToLine

Edit History Versions Settings Enabled

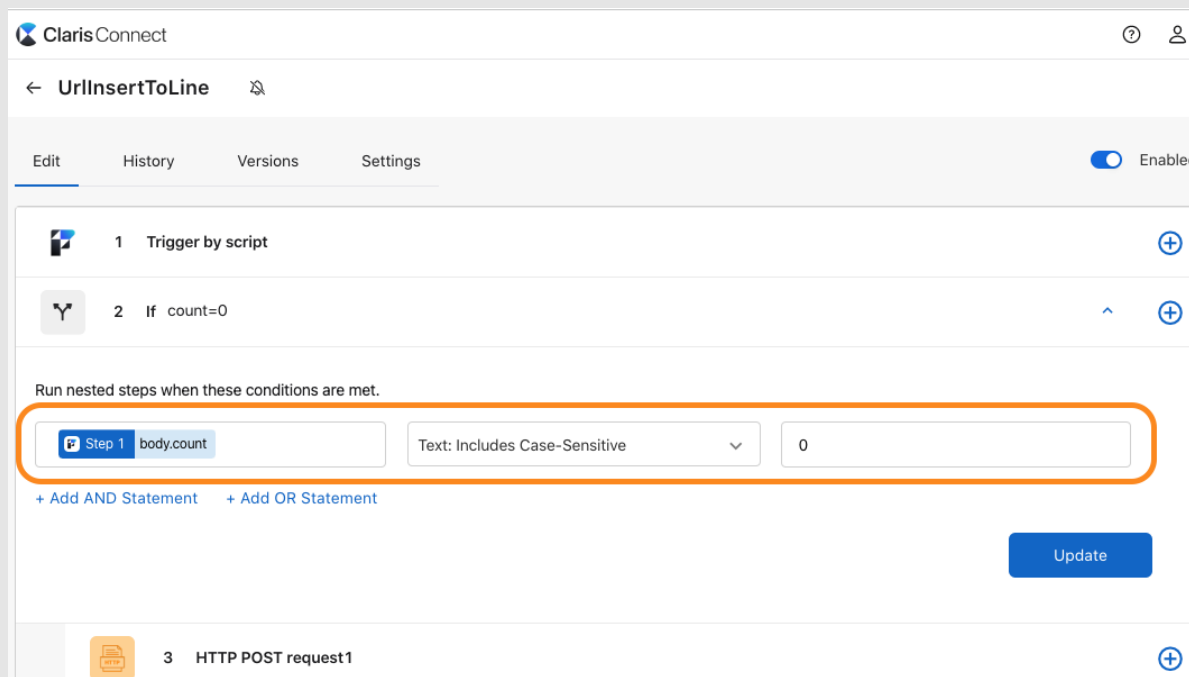
1	Trigger by script	+
2	If count=0	^ +
3	HTTP POST request1	+
4	If count=1	^ +
5	HTTP POST request2	+
6	If count=2	^ +
7	HTTP POST request3	+
8	If count=3	^ +
9	HTTP POST request4	+
10	If count=4	^ +
11	HTTP POST request5	+

一番めのトリガーのURLはワクチンProのスクリプト「Reply\_引数渡す\_Line」の下記の部分に入れます。

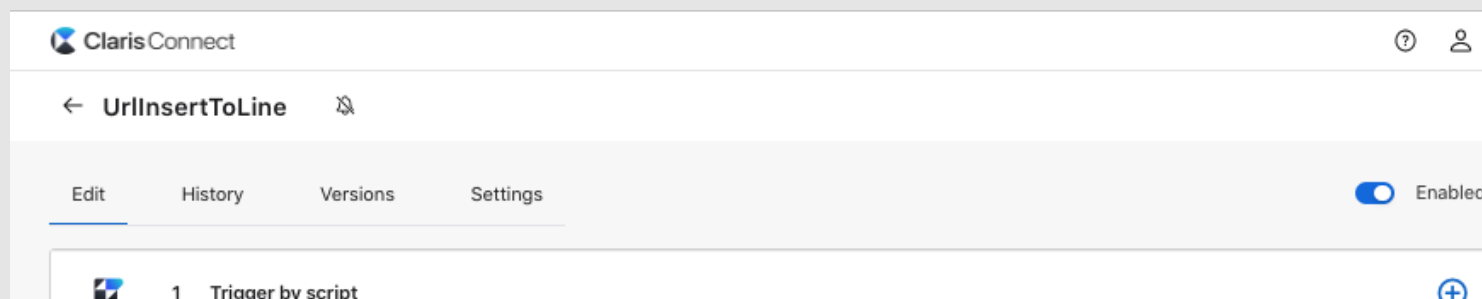




分岐はcount 4 まで1番目と2番目枠は同様にして、一番上の分岐から0,1,2,3,4とします。名前はわかりやすいように「count=1」, 「count=2」, 「count=3」, 「count=4」. 「count=5」とします。



HTTP POST request1 ( Data はJSON形式です。)  
 ※重要 Bearerとbody.headerの間には半角スペースがあります。



2 If count=0

3 HTTP POST request1

Select > Action > Configure

Review and Test

**URL \***  
Send request to this URL

https://api.line.me/v2/bot/message/reply

**Content-type \***  
Choose the content type of the request payload

application-json

**Data**  
Enter the JSON data

```
{"replyToken":" Step 1 body.replytoken ","messages":[{"type":"text","text":" Step 1 body.message 1 "}]}
```

**File**  
Enter a name and a link for a file object to be attached to the request. The request will be converted to multipart/form-data.

+ Add New File

**Header**  
Enter a key-value pair to be added as a headers in all requests

Authorization

Bearer Step 1 body.header

+ Add New Header

**Authentication**  
Enter a username | password (pipe-separated) for standard HTTP authentication

null

Save

HTTP POST request2 ( Data はJSON形式です。 )

※重要 Bearerとbody.headerの間には半角スペースがあります。

3 HTTP POST request1

4 If count=1

5 HTTP POST request2

Select > Action > Configure

**URL \*** Review and Test

Send request to this URL

**Content-type \***

Choose the content type of the request payload

**Data**

Enter the JSON data

```
{ "replyToken": "Step 1 body.replytoken", "messages": [{"type": "text", "text": "Step 1 body.message1"}, {"type": "text", "text": "Step 1 body.message2"} ] }
```

**File**

Enter a name and a link for a file object to be attached to the request. The request will be converted to multipart/form-data.

+ Add New File

**Header**

Enter a key-value pair to be added as a headers in all requests

+ Add New Header

**Authentication**

Enter a username | password (pipe-separated) for standard HTTP authentication

Save

6 If count=2

### HTTP POST request3 (DataはJSON形式です。)

※重要 Bearerとbody.headerの間には半角スペースがあります。

6 If count=2
⌵ Ⓡ

7 HTTP POST request3
Ⓡ

Select > Action > Configure
Review and Test

**URL\***  
Send request to this URL

https://api.line.me/v2/bot/message/reply

**Content-type\***  
Choose the content type of the request payload

application-json

**Data**  
Enter the JSON data

```

                {"replyToken": "Step 1 body.replytoken", "messages": [{"type": "text", "text": "Step 1 body.message1"}, {"type": "text", "text": "Step 1 body.message2"}, {"type": "text", "text": "Step 1 body.message3"}]}
            
```

**File**  
Enter a name and a link for a file object to be attached to the request. The request will be converted to multipart/form-data.

[+ Add New File](#)

**Header**  
Enter a key-value pair to be added as a headers in all requests

Authorization	Bearer Step 1 body.header

[+ Add New Header](#)

**Authentication**  
Enter a username | password (pipe-separated) for standard HTTP authentication

null

Save

HTTP POST request4 (Data はJSON形式です。)

※重要 Bearerとbody.headerの間には半角スペースがあります。

8 If count=3

9 HTTP POST request4

Select > Action > Configure

Review and Test

**URL \***  
Send request to this URL

https://api.line.me/v2/bot/message/reply

**Content-type \***  
Choose the content type of the request payload

application-json

**Data**  
Enter the JSON data

```
{ "replyToken": "Step 1 body.replytoken", "messages": [ { "type": "text", "text": "Step 1 body.message1" }, { "type": "text", "text": "Step 1 body.message2" }, { "type": "text", "text": "Step 1 body.message3" }, { "type": "text", "text": "Step 1 body.message4" } ] }
```

**File**  
Enter a name and a link for a file object to be attached to the request. The request will be converted to multipart/form-data.

+ Add New File

**Header**  
Enter a key-value pair to be added as a headers in all requests

Authorization Bearer Step 1 body.header

+ Add New Header

**Authentication**  
Enter a username | password (pipe-separated) for standard HTTP authentication

null

Save

HTTP POST request5 (Data はJSON形式です。)

※重要 Bearerとbody.headerの間には半角スペースがあります。

10 If count=4

11 HTTP POST request5

Select > Action > Configure

**URL\*** Review and Test

Send request to this URL

**Content-type\***

Choose the content type of the request payload

**Data**

Enter the JSON data

```
{ "replyToken": "", "messages": [ { "type": "text", "text": "" }, { "type": "text", "text": "" }, { "type": "text", "text": "" }, { "type": "text", "text": "" }, { "type": "text", "text": "" } ] }
```

**File**

Enter a name and a link for a file object to be attached to the request. The request will be converted to multipart/form-data.

+ Add New File

**Header**

Enter a key-value pair to be added as a headers in all requests

<input type="text" value="Authorization"/>	<input text"="" type="text" value="body.header"/>
<input type="text"/>	<input type="text"/>

+ Add New Header

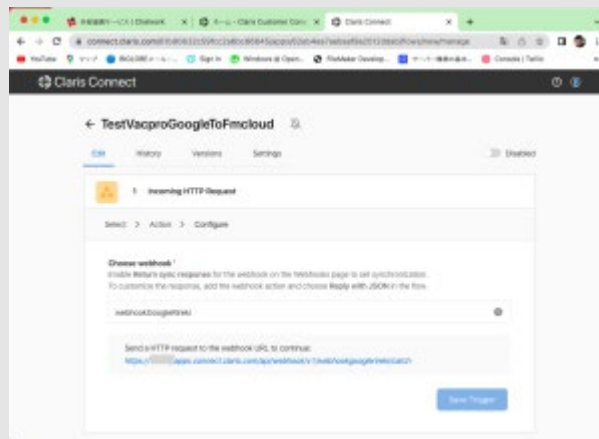
**Authentication**

Enter a username | password (pipe-separated) for standard HTTP authentication

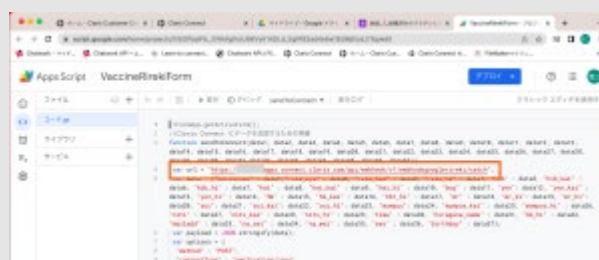
Save

## <TestVacproFromGoogleToFMcloud>

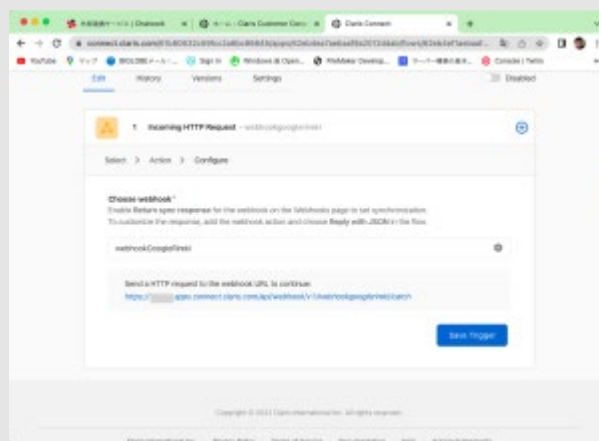
まずは、webhookを用意します。



このwebhookのURLをGoogle FormのApps scriptに貼り付けます。

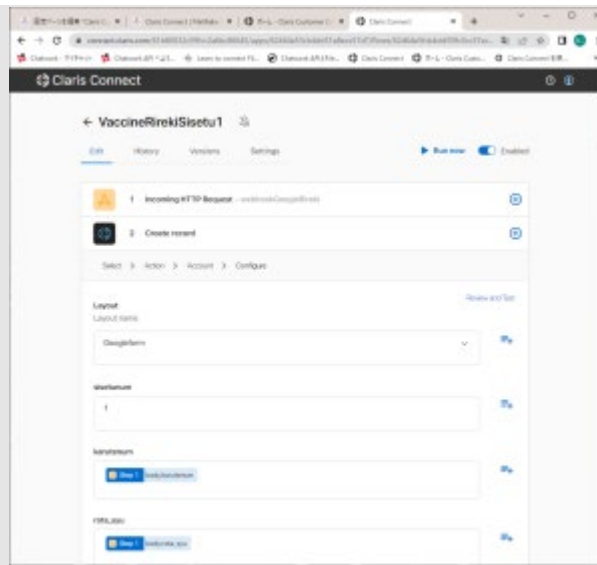


Google Formの設定方法一を参照してGoogle formのトリガーも準備しておきます。フォームを送ります。Save Triggerにいろが付きまして。押しておきます。

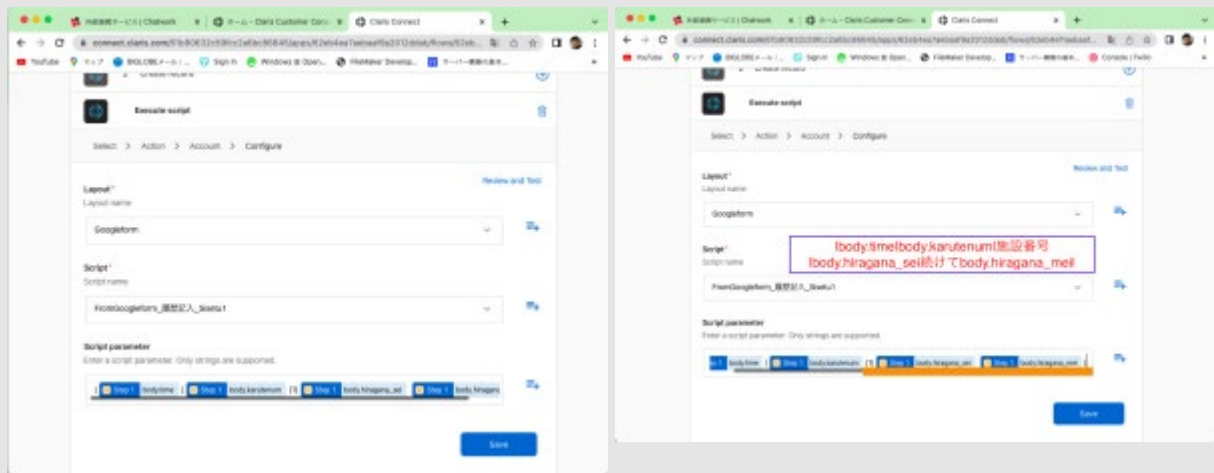


2番目のステップです。FileMaker CloudのActionであるCreate recordを選びます。動画に沿ってセットしてください。

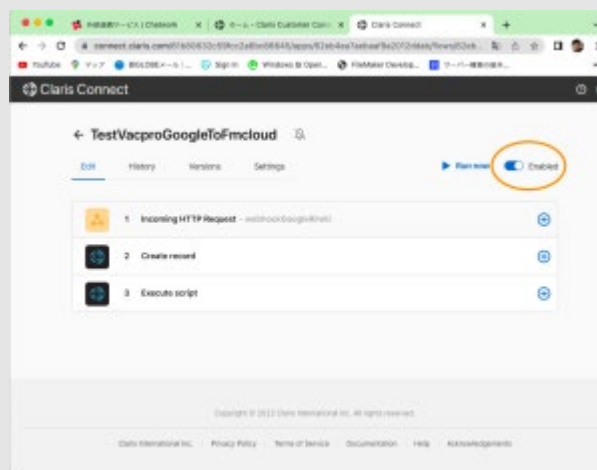
[RPReplay\\_Final1658121002](#)



最後のステップです。図のようにセットします。「」縦棒半角で挟んで、右図のようにbody.hiragana\_seiとbody.hiragana\_meiは続けてください。（データを渡すときは、ワクチンProの方がひらがなせいめいと一緒になっているので）



終われば有効にしておきます。

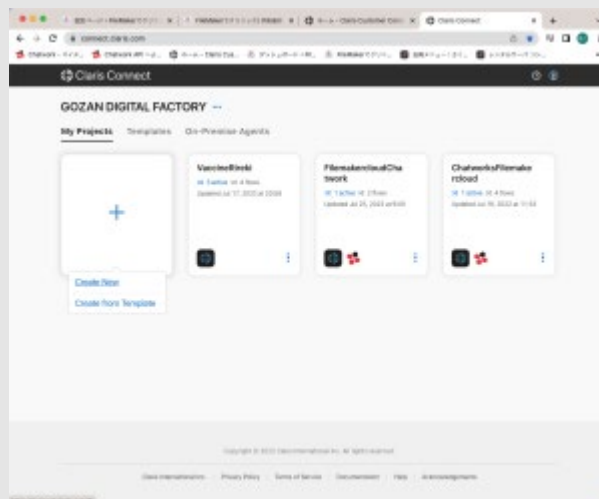




## Chatworkで設定\_旧

セットアップの前にChatBOT用にChatworkをセットしておいてください。このChatworkで接種者に返事をしていきます。

Claris Connectで、自分の好きな名前でプロジェクトを作成していきます。プロジェクトは横の+ボタンで新規作成が出来ます。下記の3プロジェクトを、例のような分かりやすい名前で作成してください。

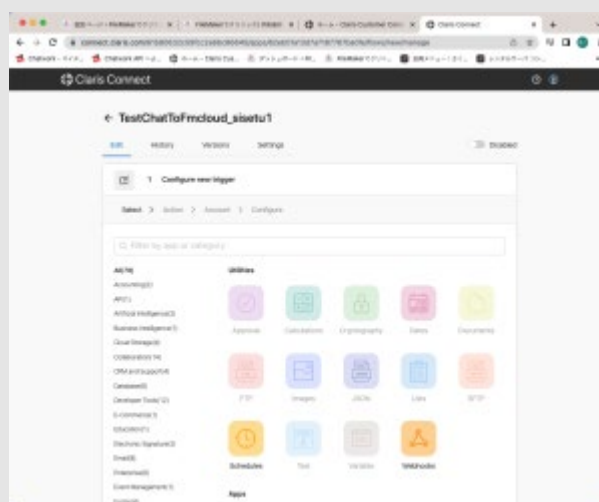


- [TestVacproChatToFmcloud](#)
- [TestVacProFmcloudToChat](#)
- [TestVacproFromGoogleToFMcloud](#) (使用中)

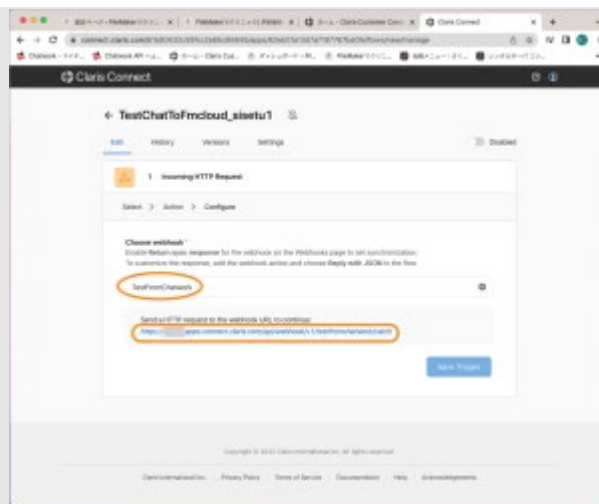
<TestVacproChatToFmcloud>

Chatworkにメッセージを出すとAPIが動作して、FileMakerにデータが入るようにします。Flowを作成します。たいていの場合は施設は1つですのでSisetu1のものを例として作っていきます。(注意 動画は一部がSisetu2のものになっていますのでSisetu1の置き換えてご覧ください。)

まずWebhookを作成します。Webhookを選択します。



Webhookに名前を付け、下のURLをコピーします。

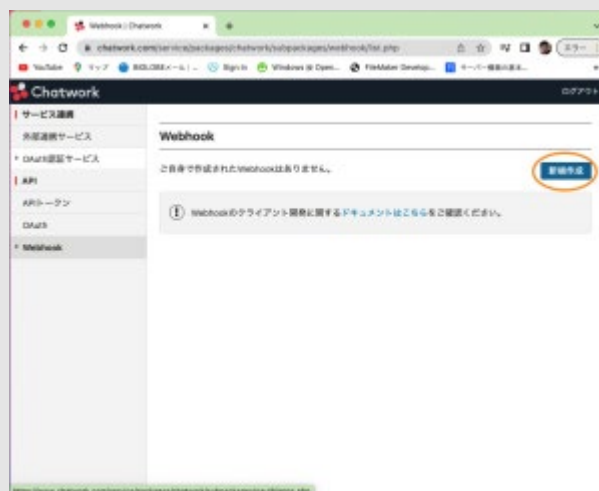


次にChatworkでこのwebhookを設定します。

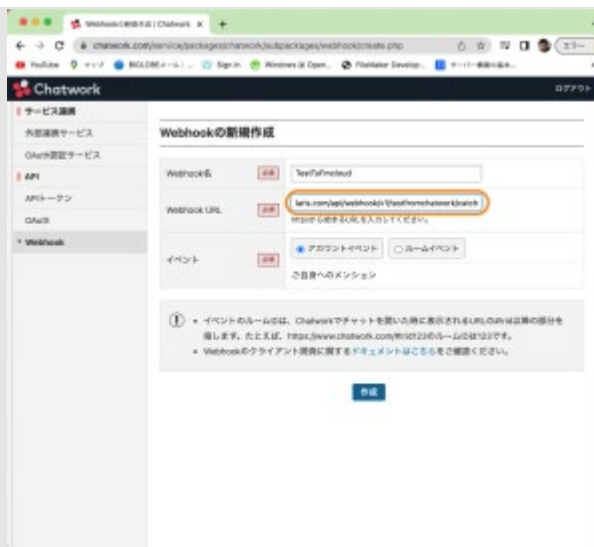
上の右隅のメニューをポップアップさせて>サービス連携を選択します。



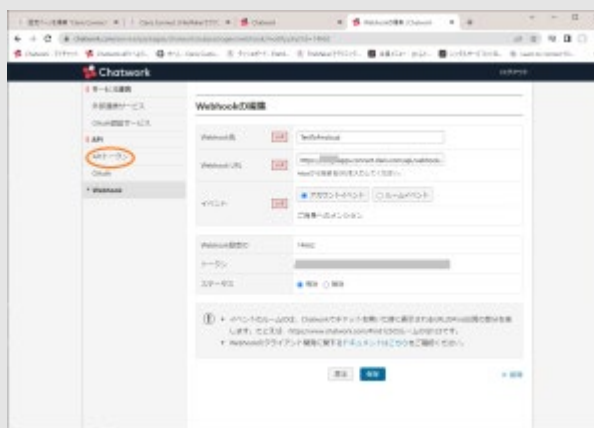
関連URLがひらきますので左メニューのWebhookを選択して新規作成します。



開いたところに先ほどコピーしたWebhookのURLを貼り付けます。ChatworkのAPIはここにデータを送ってきます。



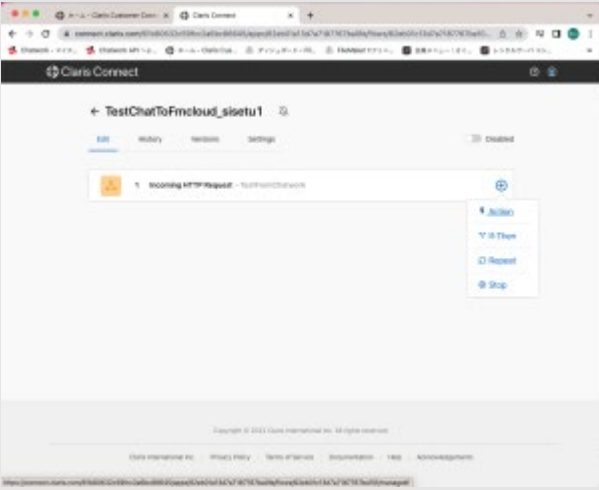
作成を押し、その後左メニューのAPIトークン開き値をコピーします。



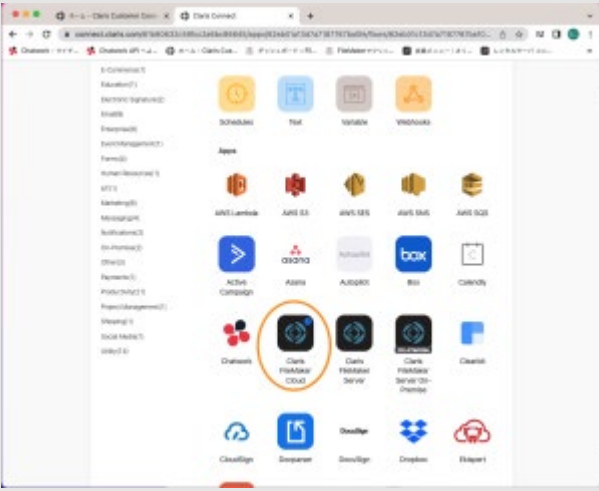
APIトークンをコピーしVacProの下図の位置に貼り付けます。



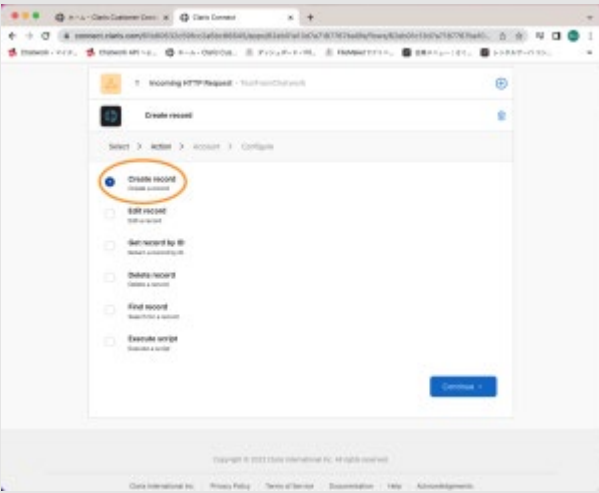
これでAPIが使えるVacPro側とChatwork側の準備ができました。Chatworkからメッセージを送ると「Save Trigger」に色が付きますのでテストします。次のステップです。FileMakerにデータを流し込む新規レコードを準備します。作成し終えたWebhookの右方の+ボタンを押し「Action」を選択します。



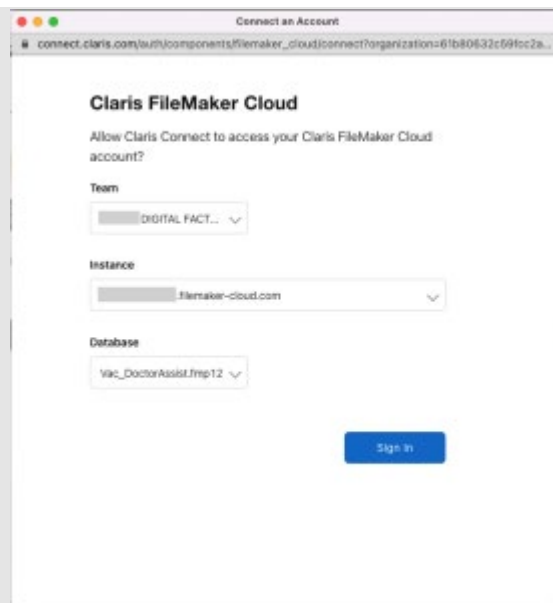
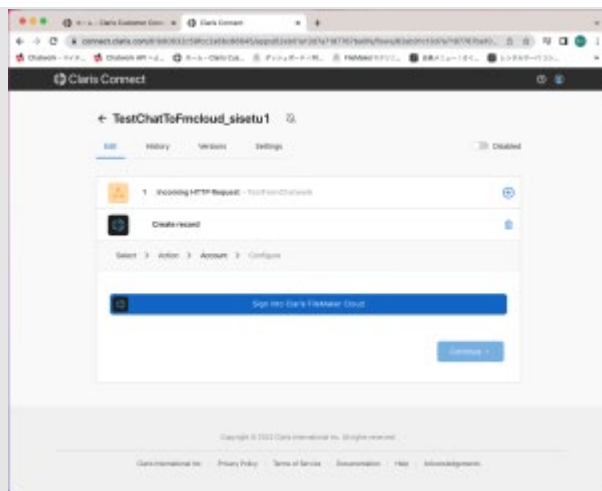
Appsの中にあるClaris FileMaker Cloudを選択します。



Create recordを選択します。

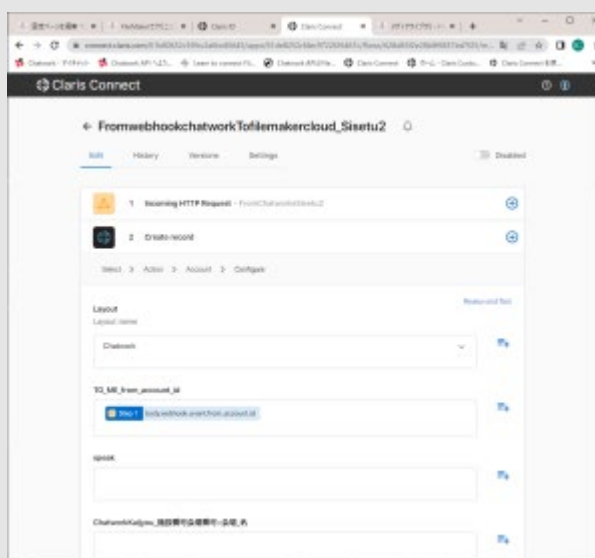


Claris FileMaker Cloudの認証を入れます。

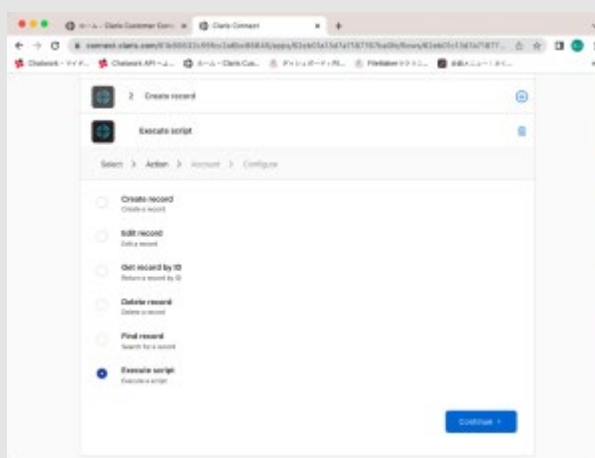


RPReplay\_Final1658112649

※施設番号に画像では「2」となっていますが、施設1の設定の時は「1」にしてください。

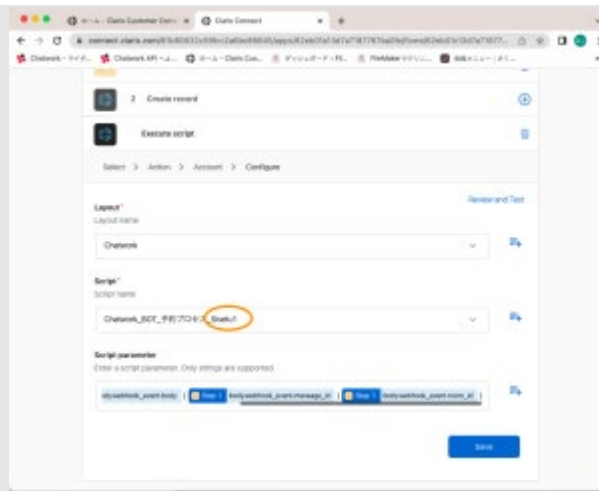


3番目のステップです。ActionにExecute scriptを選んでください。

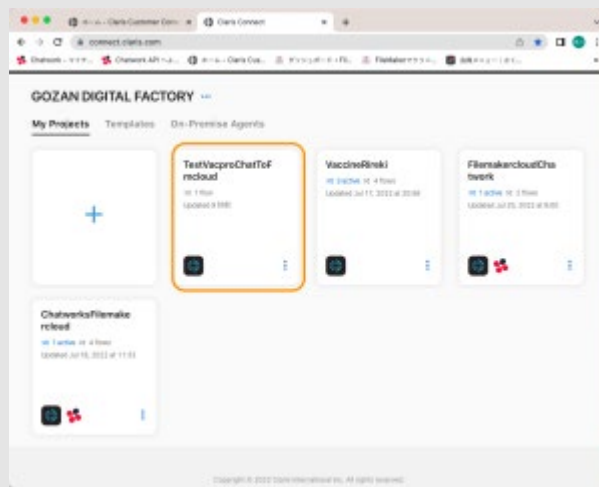


RPReplay\_Final1658114099

※Scriptは「Chatwork\_BOT\_Sisetu\_2」になっていますが施設1の時は「Chatwork\_BOT\_Sisetu\_1」でお願いします。

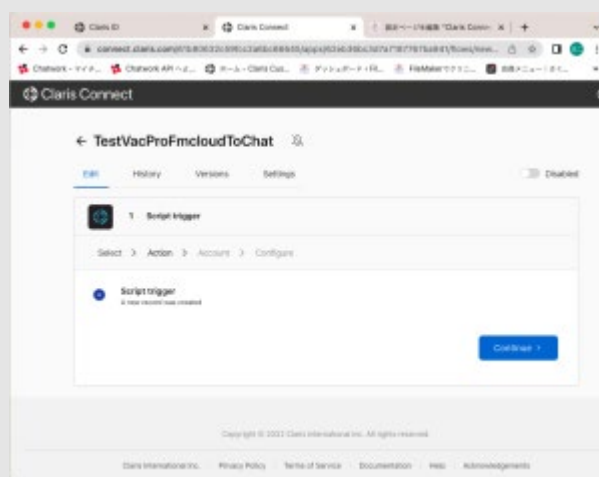


1つ目のプロジェクトの出来上がりです。

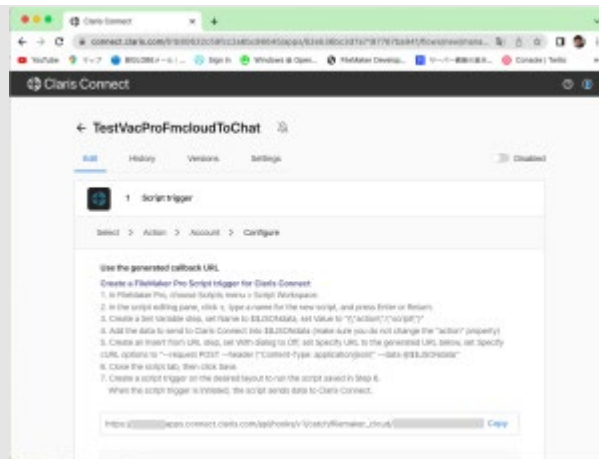


<TestVacProFmcloudToChat>

次のプロジェクトのTestVacProFmcloudToChatをセットします。今度はFileMaker CloudをトリガーにしてChatworkにデータを渡します。(FileMaker Server on premiseはトリガーにはできません。) ActionをScript triggerにします。



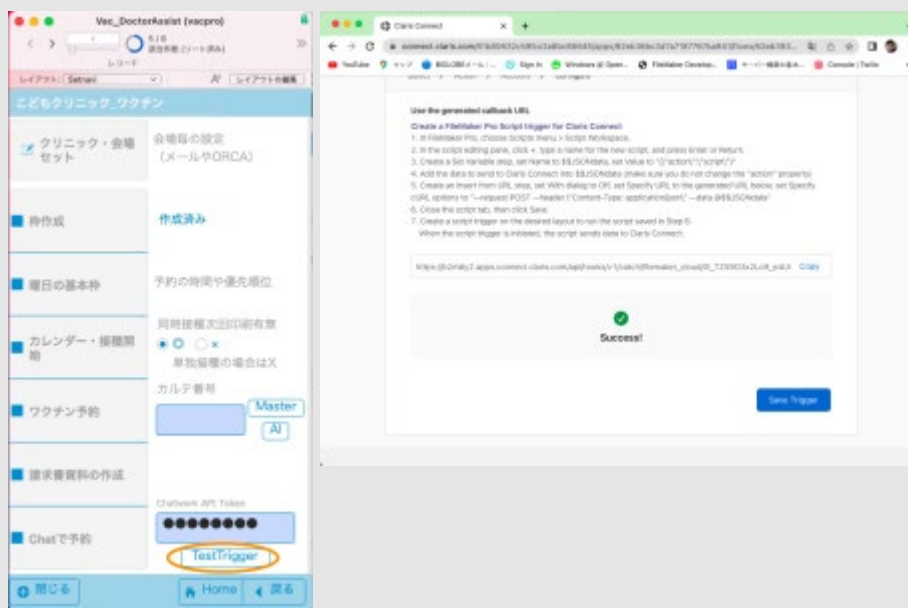
Continueを押した後1つ目のステップと同様にFileMaker Cloudの認証を通します。



書かれているURLをコピーしVacProのスクリプトに貼り付けます。ハイライトされているスクリプト「FMCloud\_connectTrigger\_引数渡す\_chatwork」の図の17行目に貼り付けてください。

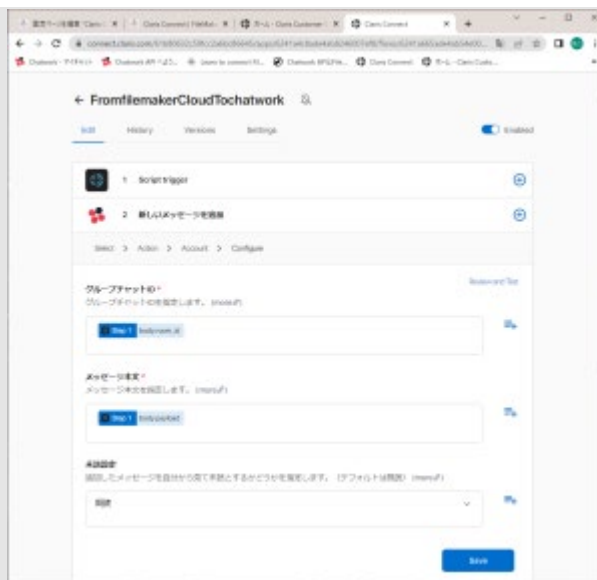


終わったらVacProのレイアウトの「Setnavi」レイアウトでTesttriggerボタンを押してみてください。Save triggerに色が付きますので押してください。これでトリガーが出来ました。



これを押しておいてください。終わればChatworkの情報を入れて、Actionの新しいメッセージの追加にチェックを入れて最終画面で下図のようにセットします。

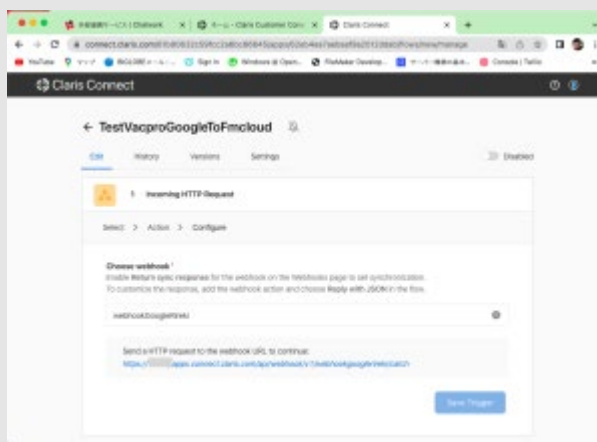




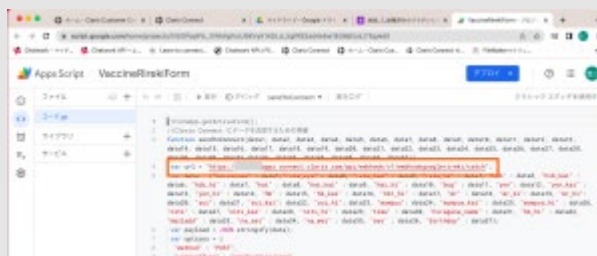
これでFileMaker(VacPro)からChatworkにデータが渡せるようになりました。

### <TestVacproFromGoogleToFMcloud> (LINEでも使用)

まずは、webhookを用意します。

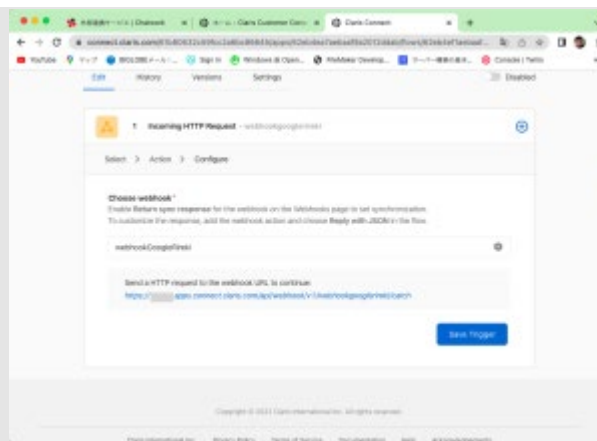


このwebhookのURLをGoogle FormのApps scriptに貼り付けます。



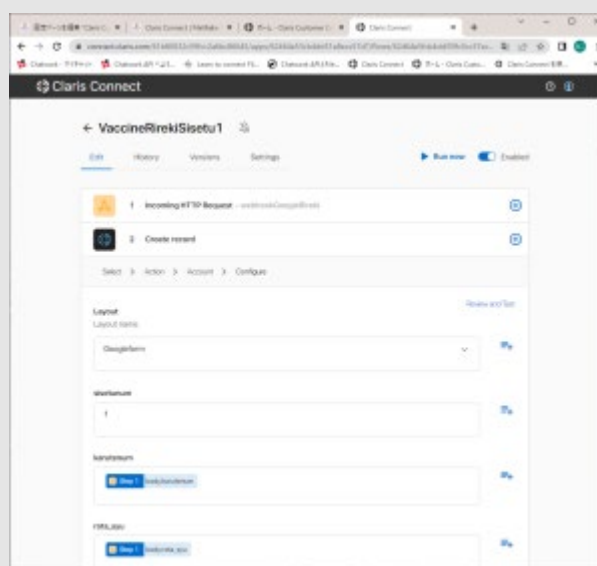
Google Formの設定方法一を参照してGoogle formのトリガーも準備しておきます。フォームを送ります。Save Triggerにいますのが付きました。押しておきます。



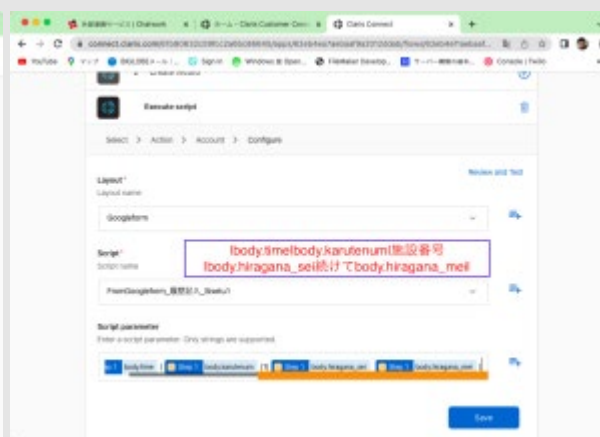
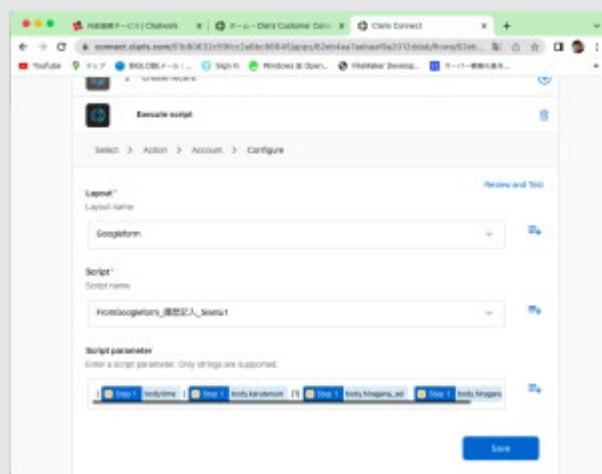


2番目のステップです。FileMaker CloudのActionであるCreate recordを選びます。動画に沿ってセットしてください。

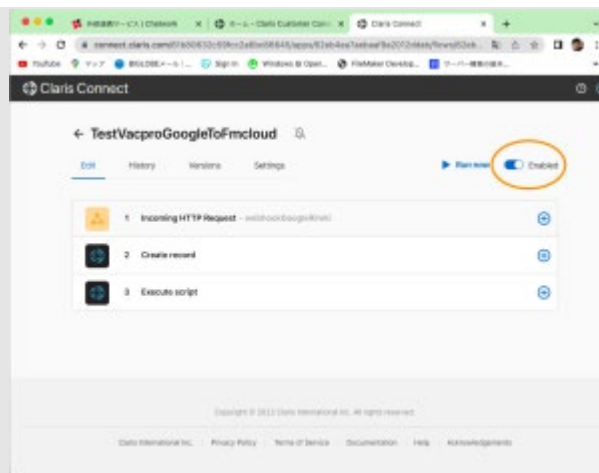
RPReplay\_Final1658121002



最後のステップです。図のようにセットします。「」縦棒半角で挟んで、右図のようにbody.hiragana\_seiとbody.hiragana\_meiは続けてください。（データを渡すときは、ワクチンProの方がひらがなせいめいと一緒になっているので）



終われば有効にしておきます。



以上。