

Programming Boot Camp Learning Phase #3

Bubble Advance

2023/11/11

目次

- Bubble での外部連携
- Bubble での共同作業の Tips
- チーム開発演習
- まとめ

やることの確認

- ここではまず、Bubble で作ったアプリケーションを外部のサービスと連携させる方法を学びます
- 次に、1つのアプリケーションを使った共同作業に関するコツや注意点をご紹介します
- 最後に、Development Phaseのチームでアプリ開発の演習と発表を行います

Bubble での外部連携

.bubble

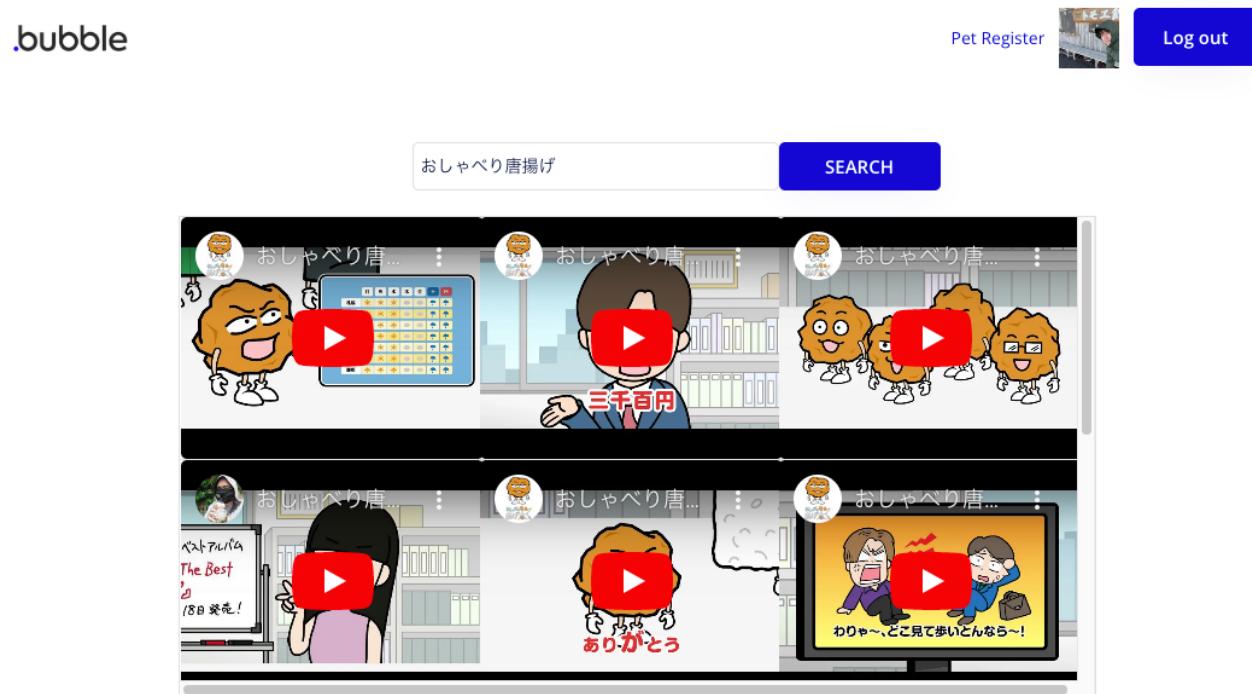
Pet Register

Log out



機能としては下記を盛り込んだ画面となります

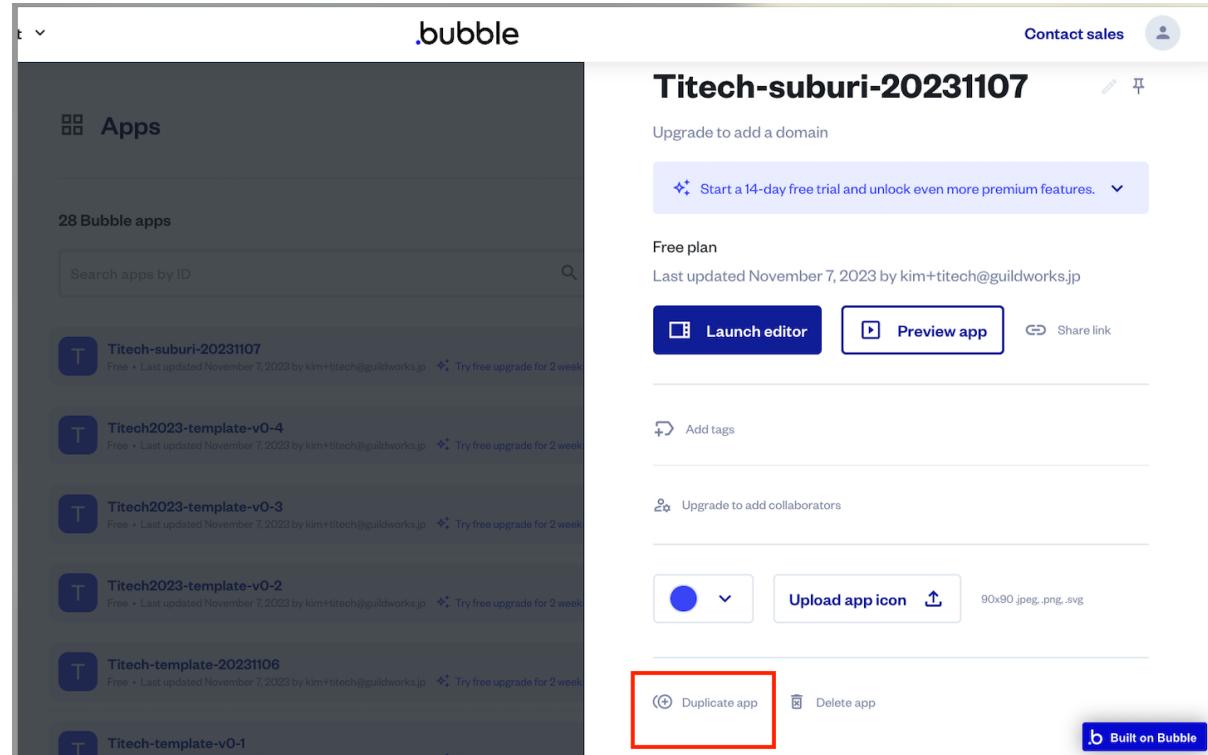
- YouTube の動画一覧を表示
- Google アカウントを使ったソーシャルログイン
- YouTube の動画一覧を動的に検索



まずは土台となるアプリケーションを新たに用意

- Bubble 1回目の講義で使ったアプリケーションをコピーして使い回したいと思います
- <https://bubble.io/home?tab=apps> にアクセス

- 各自のアカウントで作成している Bubble アプリの一覧が表示されます
- その中で、先ほどまでの基礎編で作ってきたアプリケーションを選択して、ポップアップの最下部にある **Duplicate app** をクリック



- コピー後のアプリケーション名の入力ダイアログが表示されるので、適当な名前を入力
- 以前のデータは使い回さないので **Copy the application database content** のチェックは OFF のまま
- COPY をクリック

[Back to main menu](#)

Copy app

Name of this copy of titech-suburi-20231107

bubble-advanced

You can add a domain name later.

Copy the application database content

Copy

- コピーが終わると、コピーしたアプリケーションの編集画面（index）が表示されればOKです

Page: index Pick an element... Edit Saved

UI Builder Responsive
See reference → Elements Tree

Design Only show hideable

Workflow Page index + Group Sign Up/Log in 2 col

Data Search elements

Visual Elements

T Text

Button

Icon

Link

Image

Shape

Alert

Video

HTML

Map

Built on Bubble

Expression

Javascript to Bubble

Line/Bar Chart

List Item Expression

PetLog

Name

John Doe

Email

johndoe@gmail.com

Password

Re-enter Password

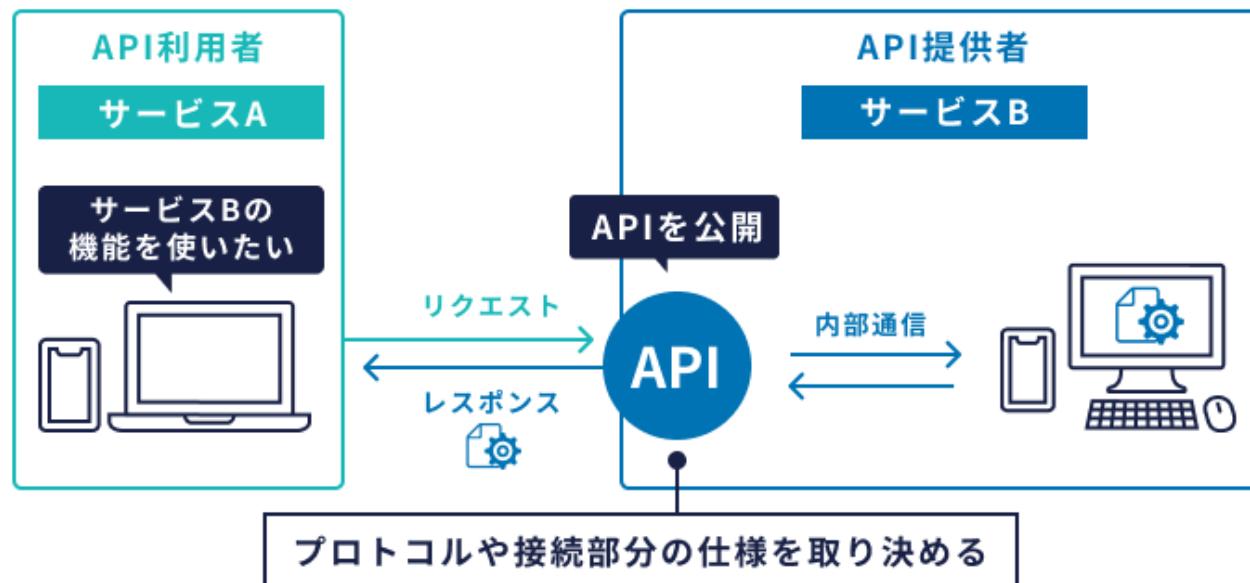
Role

Choose an role...

Upgrade to deploy Preview

APIって何？

「API (Application Programming Interface)」のことで、ソフトウェアやアプリケーションが他のソフトウェアやサービスと通信するための方法です。APIを通じて、アプリケーションはデータを共有したり、他のアプリケーションの機能を公開し、利用することができます。



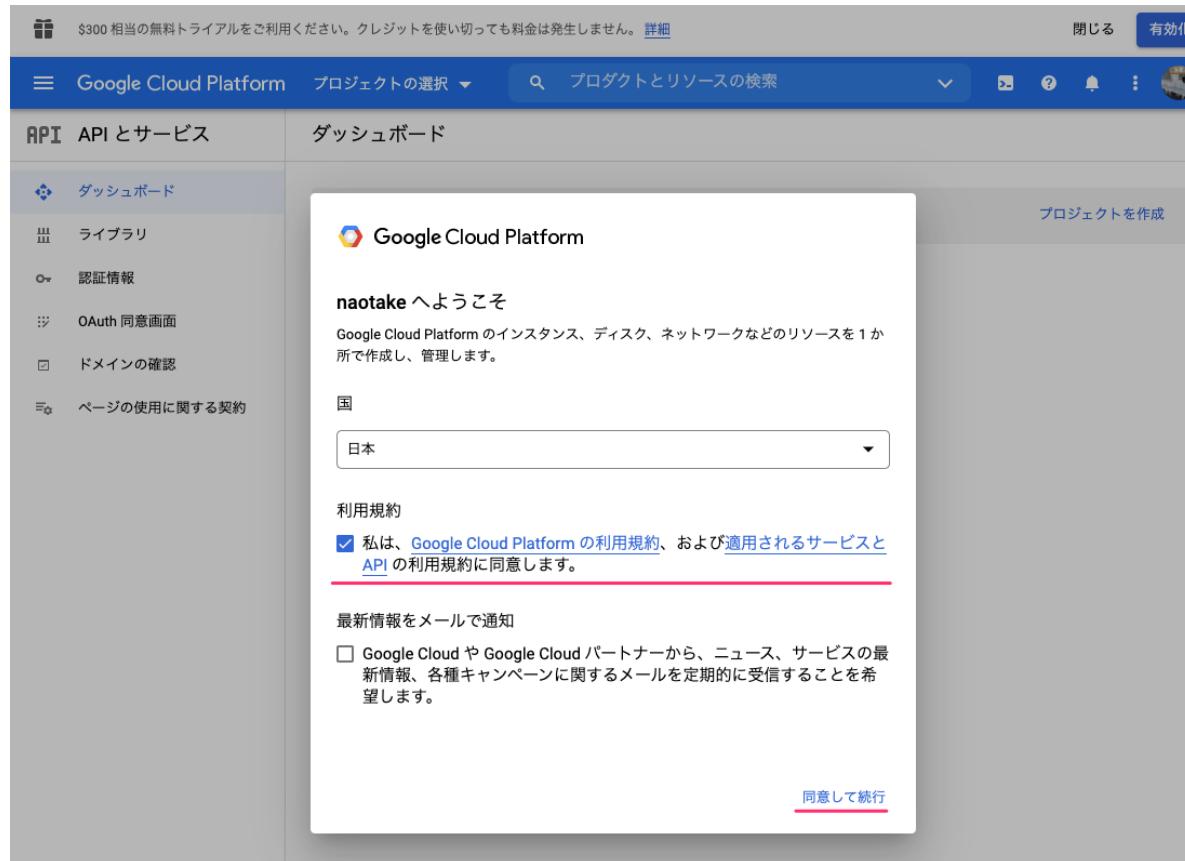
YouTube 連携の事前準備

- 続いて、YouTube の動画一覧を表示するために必要な下準備を行います
- 今回の講義では Google が提供しているクラウドサービス "Google Cloud Platform" に用意されている各種 API を使っていきます
 - Amazon が提供している "Amazon Web Service" (AWS) の Google 版だと思ってください

- 下記 URL にアクセス
 - <https://console.developers.google.com/>
- ご自身の Google アカウントでログインしてください
 - もし、Google アカウントをお持ちでない方は、アカウントの作成をお願いします



- ログイン後、はじめて Google Cloud Platform(GCP) の画面にアクセスするとこのようなダイアログが表示されますので、国を選択し、利用規約にチェックを入れて「同意して続行」



- 同意すると、API とサービスのダッシュボードが表示されると思いますので、プロジェクトを作成のリンクを押下して今回の講義用のプロジェクト（箱）を作成します

The screenshot shows the Google Cloud Platform API & Services dashboard. On the left, there's a sidebar with icons for Dashboard, Library, Authentication, OAuth Consent Screen, Domain Verification, and Terms of Use. The main area is titled "ダッシュボード" (Dashboard) and contains a message: "このページを表示するには、プロジェクトを選択してください。" (Please select a project to view this page.) Below this message is a red underlined link labeled "プロジェクトを作成" (Create Project). At the top of the screen, there's a banner for a \$300 free trial, a "閉じる" (Close) button, and a "有効化" (Enable) button.

- 新しいプロジェクトの作成画面が表示されるのでプロジェクト名に適当な名前を指定して作成

新しいプロジェクト

⚠ 割り当て内の残りのプロジェクト数は 12 projects 件です。プロジェクトの増加をリクエストするか、プロジェクトを削除してください。[詳細](#)

[MANAGE QUOTAS](#)

プロジェクト名 * ?

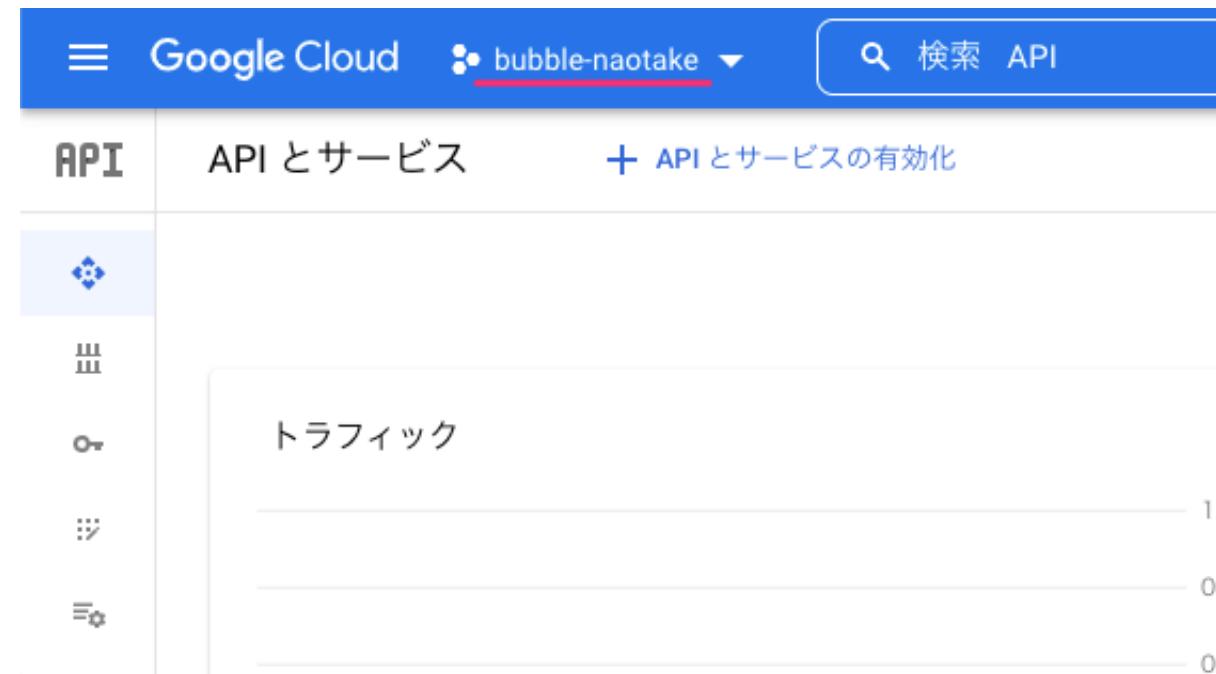
プロジェクト ID: **bubble-advance**。後で変更することはできません。 [編集](#)

場所 * 参照

親組織またはフォルダ

[作成](#) [キャンセル](#)

- 作成が完了すると、作成したプロジェクトのダッシュボードが表示されます
- 画面上部の Google Cloud の左隣に作成したプロジェクト名が表示されていれば OK



- 続いて YouTube の動画一覧を表示するために必要な認証キーを発行していきます
- API とサービスというメニューを開くため、画面上部の検索ボックスに "API" と入力
- サジェスト候補の中から "プロジェクトとページ" ブロックにある "API とサービス" を選択



- API とサービスページが開いたら、左メニューから「認証情報」を選択
- 右パネルが認証情報に変わるので、画面上部の「+ 認証情報を作成」をクリック
- ポップアップから「API キー」を選択

The screenshot shows the Google Cloud Platform interface for managing APIs and services. On the left, there's a sidebar with the following items:

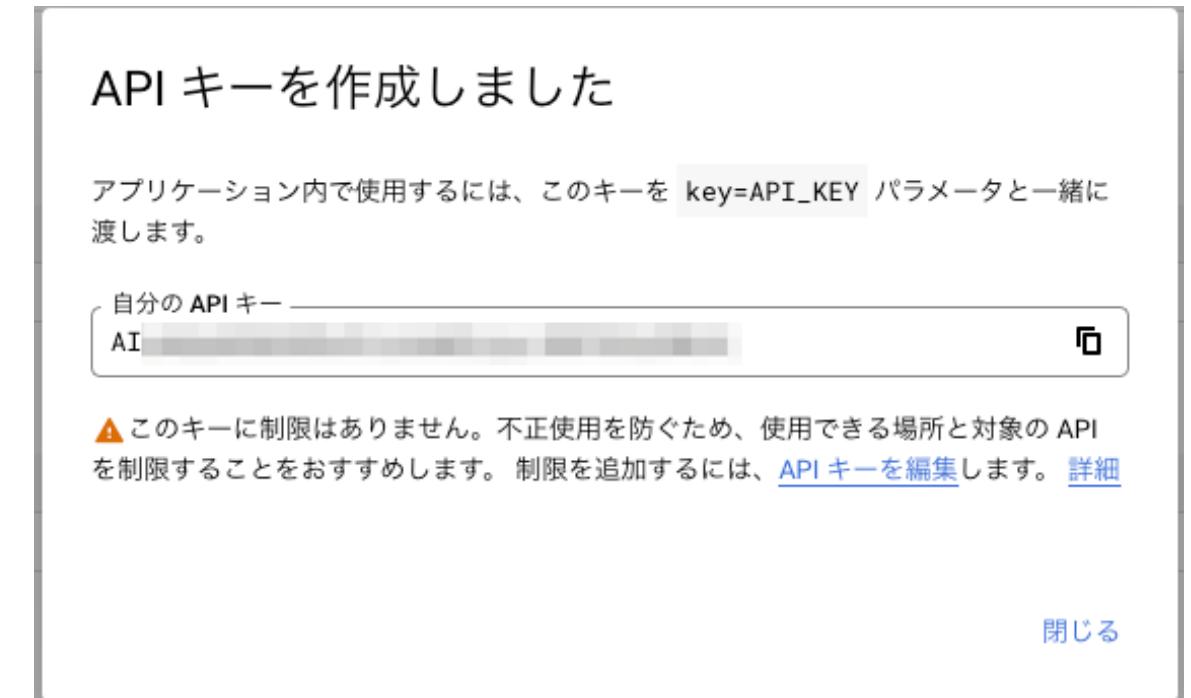
- 有効な API とサービス
- ライブラリ
- 認証情報** (highlighted in blue)
- OAuth 同意画面
- ページの使用に関する契約

The main content area has a header "認証情報" with two buttons: "+ 認証情報を作成" (highlighted with a red underline) and "削除". Below this, there are three options:

- API キー**: Described as "シンプル API キーを使用してプロジェクトを識別し、割り当てとアクセスを確認します". A warning message says "必ず、API キーを生成する前に、このクライアント ID を登録しておいてください" (Always, generate an API key before registering this client ID).
- OAuth クライアント ID**: Described as "ユーザーのデータにアクセスできるようにユーザーの同意をリクエストします".
- サービス アカウント**: Described as "ロボット アカウントによるサーバー間でのアプリレベルの認証を有効にします".

At the bottom of the modal, there's a section for "表示する API キー" (Select API key to display) and a "名前" (Name) input field.

- すると API キー作成完了のポップアップが表示されるので、その値を控えておきます
- キーを制限しろと言われますが、講義の最後にこれらのキーを削除しますので一旦このまま行きます



- これで Bubble から YouTube の一覧を取得するためのキーが発行できたので Bubble 側の設定を行っていきます
- まずは Bubble から YouTube の一覧を API を介して取得するための準備を行います

Bubble の API 連携の準備

- 左メニューから Plugins を選択し、Add plugins ボタンをクリック
- "API" で検索し、API Connector というプラグインをインストールします

Install New Plugins

Filters

Sort by: name usage rating date price

API

Types (deselect all)

- Action
- Api
- Background Services
- Element
- Event
- Login Service
- Storage

Categories (deselect all)

- Analytics
- Blog
- Compliance
- Containers
- Customer Support
- Data (things)
- Ecommerce
- Email
- Energy
- Health & Fitness
- Input Forms
- Internationalization
- Location
- Media
- Mobile

API Connector

The API lets you define your own API calls directly in the Bubble Editor and use them in your app.

By Bubble

289,709 apps

Install

Zapier (Legacy)

Zapier connects the web apps you use to easily move your data and automate tedious tasks. Trigger any Zap from your workflows.

By Bubble

12,498 apps

Install

Discord API (OUTDATED)

Integrate your web app with Discordapp services Supplies Discord login service plus Webhooks, and guilds a user is in. Previously known as Discordapp API

By Tristian Forcum

10,372 apps All versions 3.5/5.0 Current version 3.5/5.0 Plugin page

Install

Widgio API Connector

Connect to the Widgio API to create fully customized web widgets for your Bubble application.

By twobrothers.

6,170 apps All versions 3.0/5.0 Current version 3.0/5.0 Plugin page

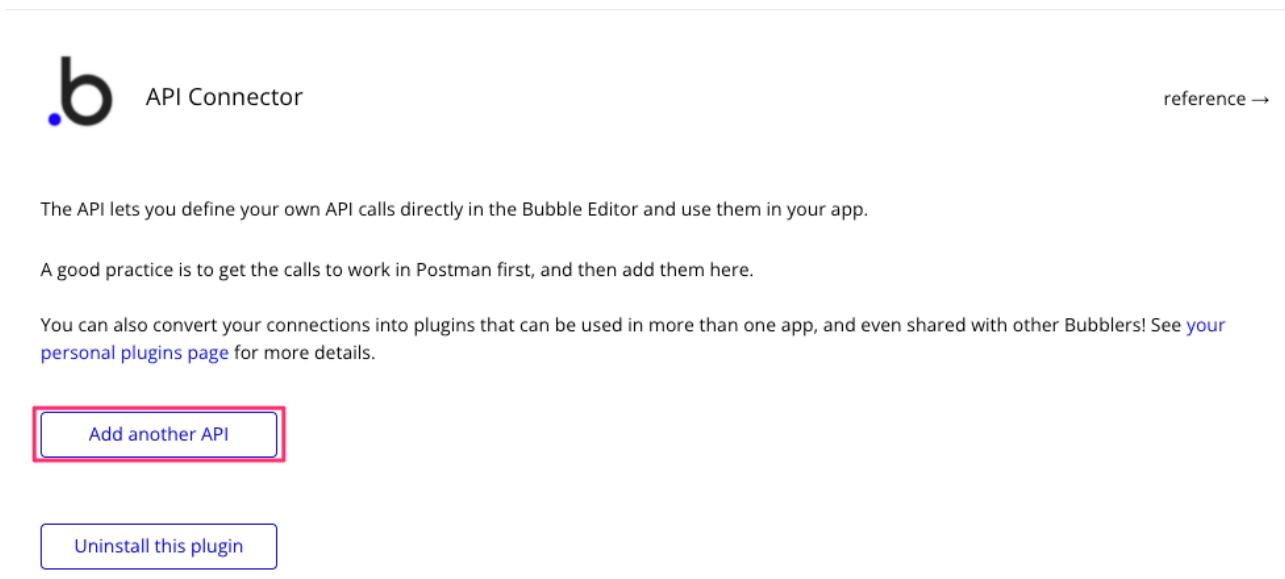
Install

DONE

⚠ Recurring subscriptions to paid plugins are on paid plans only. Click here to adjust your application plan.

- このプラグインは Bubble のアプリから、世の中で公開されている API を使って、API の向こう側の情報と Bubble を繋げるためのプラグインとなります
- いくつかの Bubble のプラグインの中には、特定の API に特化させたものもあり、それらのプラグインはこの API Connector よりも設定が単純なものもあります
- 今回は一番の基本形となる API Connector プラグインを介して YouTube API を使い、動画の一覧を表示してみましょう

- それでは YouTube API を使うための設定を行っていきます
- Plugins タブで Installed Plugins の中から "API Connector" を選択
- 右パネルにこのプラグインの概要が記載されているので、その下にある "Add another API" をクリックし、新しい API 連携の設定を開始します



- "Add another API" をクリックすると、このような API 設定部品が表示されると思います
- 各項目の簡単な説明と設定をしていきます

collapse

API Name Authentication delete

Shared headers for all calls

Add a shared header

Shared parameters for all calls

Add a shared parameter

expand

Name

Import another call from cURL

Add another call expand all calls

1. API Name: この API 連携のグループ名

- 今回は YouTube

The screenshot shows the configuration screen for a new API integration. At the top, there are fields for 'API Name' (set to 'YouTube') and 'Authentication' (set to 'None or self-handled'). A 'collapse' button is located in the top right corner. Below these, there are sections for 'Shared headers for all calls' and 'Shared parameters for all calls', each with an 'Add a shared header' and 'Add a shared parameter' button respectively. A large 'expand' button is positioned above a section for adding API calls. At the bottom, there is a 'Import another call from cURL' field and a blue 'Add another call' button. A 'expand all calls' button is located in the bottom right corner.

① API Name YouTube ② Authentication None or self-handled collapse

③ Shared headers for all calls
Add a shared header

④ Shared parameters for all calls
Add a shared parameter

Name API Call expand

Import another call from cURL
Add another call expand all calls

2. Authentication: このグループ配下の API で行う共通の認証方式

- 今回は "None or self-handled"

The screenshot shows the configuration interface for an API group. At the top, there are fields for 'API Name' (set to 'YouTube') and 'Authentication' (set to 'None or self-handled'). A 'collapse' button is located in the top right corner. Below these, there are sections for 'Shared headers for all calls' and 'Shared parameters for all calls', each with an 'Add a shared header' and 'Add a shared parameter' button respectively. A large 'expand' button is positioned above a section labeled 'Name API Call'. At the bottom left, there is a link to 'Import another call from cURL' and a blue 'Add another call' button. A 'expand all calls' button is located at the bottom right.

① API Name YouTube

② Authentication None or self-handled ▾

collapse

③ Shared headers for all calls

Add a shared header

④ Shared parameters for all calls

Add a shared parameter

expand

Name API Call

Import another call from cURL

Add another call

expand all calls

3. Shared headers for all calls: このグループ配下の API すべてで指定する共通のヘッダー情報

- 今回は特に指定しません

The screenshot shows the configuration interface for an API call. At the top, there are fields for 'API Name' (set to 'YouTube') and 'Authentication' (set to 'None or self-handled'). A red circle labeled ① points to the 'API Name' field, and a red circle labeled ② points to the 'Authentication' dropdown.

Below these, a section labeled ③ 'Shared headers for all calls' contains a button 'Add a shared header'. A red circle labeled ④ points to this button.

Further down, a section labeled ④ 'Shared parameters for all calls' contains a button 'Add a shared parameter'. Below this, there is a collapsed section labeled 'Name API Call' with an 'expand' button.

At the bottom left, there is a link 'Import another call from cURL' and a blue button 'Add another call'. At the bottom right, there is an 'expand all calls' button.

4. Shared parameters for all calls: このグループ配下の API すべてで指定する共通のパラメータ情報

- 今回は特に指定しません

The screenshot shows the configuration interface for shared parameters across all API calls. At the top, there are fields for 'API Name' (set to 'YouTube') and 'Authentication' (set to 'None or self-handled'). Below these are sections for 'Shared headers for all calls' and 'Shared parameters for all calls'. Each section has a 'Add a shared header' or 'Add a shared parameter' button. A large expandable box at the bottom contains an 'Import another call from cURL' field and an 'Add another call' button.

collapse

① API Name YouTube ② Authentication None or self-handled ▾

③ Shared headers for all calls

Add a shared header

④ Shared parameters for all calls

Add a shared parameter

expand

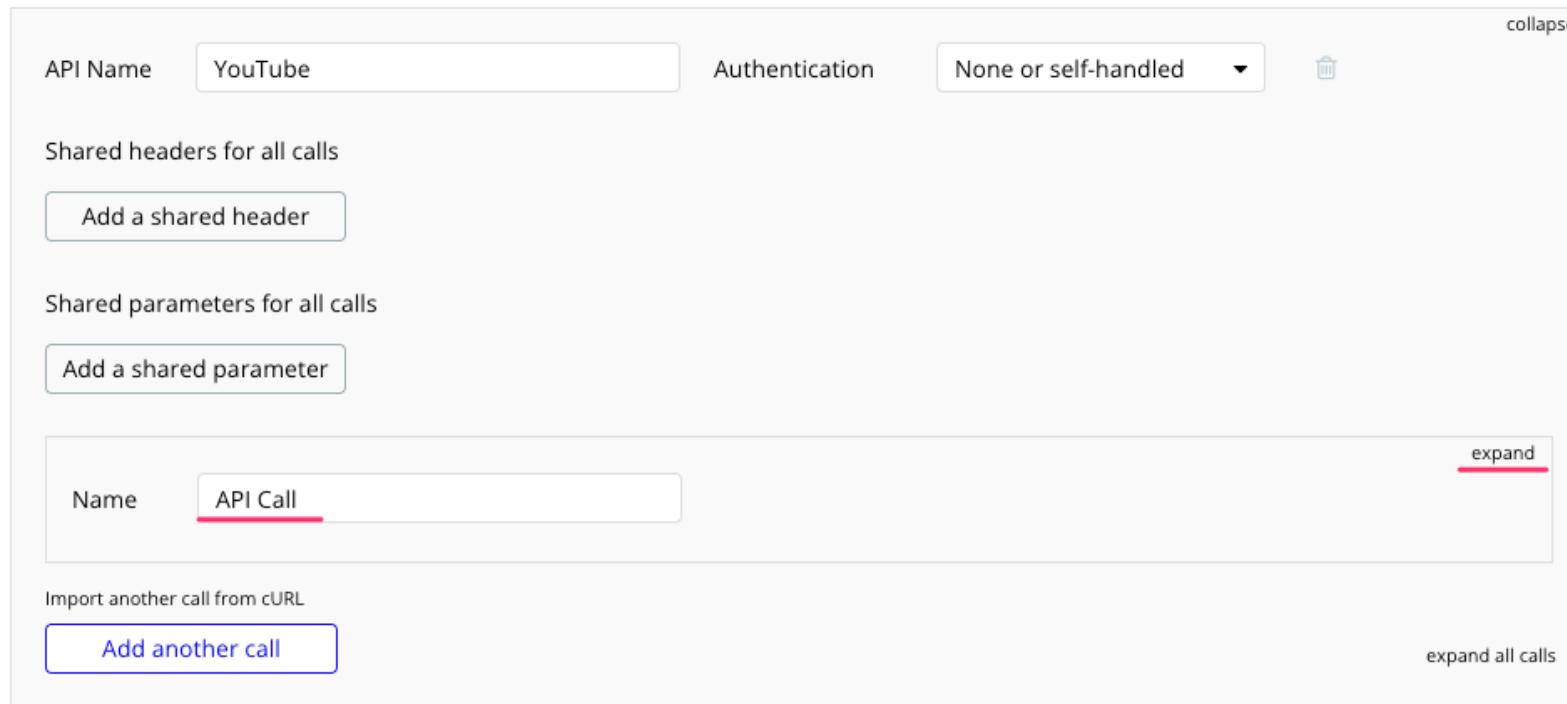
Name API Call

Import another call from cURL

Add another call

expand all calls

- 続いて、このグループに対して具体的な API の内容を設定していきます
- 先ほどの画面で "API Call" となっているブロックの右側に expand というリンクがあるのでそれをクリック



- すると、具体的な API の設定項目が表示されますので、簡単に説明します

collapse

Name API Call Use as Data ▾ Data type JSON ▾ Delete

Method GET (use [] for params)

Attempt to make the call from the browser

Headers

Add header

Parameters

Add parameter

Include errors in response and allow workflow actions to continue

Capture response headers

⚠ You need to initialize this call before it will work.

Initialize call Manually enter API response

1. Name: 具体的な API の名前

- 今回は "search" と入力します

The screenshot shows the configuration of an API call within a workflow editor. The configuration includes:

- Name:** search (marked with ①)
- Method:** GET (marked with ②)
- URL:** https://www.googleapis.com/youtube/v3/search
- Data type:** JSON
- Headers:** A section labeled ③ Headers with an Add header button.
- Parameters:** A section labeled ④ Parameters with two entries:
 - Key: key, Value: AlzaSyAA3b34Zbc9r zx nw, Private: checked, Allow blank: unchecked, Optional: unchecked
 - Key: q, Value: おしゃべり唐揚げ, Private: checked, Allow blank: unchecked, Optional: unchecked
- Advanced Options:** Includes checkboxes for "Include errors in response and allow workflow actions to continue" and "Capture response headers".
- Initialization:** A warning message: "⚠ You need to initialize this call before it will work." followed by two buttons: "Initialize call" (highlighted with a blue border) and "Manually enter API response".

2. Path: 具体的な API の URL

- 今回は `https://www.googleapis.com/youtube/v3/search` を入力します

The screenshot shows a workflow editor interface with the following configuration:

- Name:** search (marked with ①)
- Method:** GET (marked with ②)
- URL:** `https://www.googleapis.com/youtube/v3/search`
- Data Type:** JSON
- Headers:** (marked with ③) - Add header button
- Parameters:** (marked with ④)
 - Key: key, Value: AlzaSyAA3b34Zbc9rznw, Private: checked, Allow blank: unchecked, Optional: unchecked
 - Key: q, Value: おしゃべり唐揚げ, Private: checked, Allow blank: unchecked, Optional: unchecked
- Include errors in response and allow workflow actions to continue:** Unchecked
- Capture response headers:** Unchecked
- Warning:** `⚠ You need to initialize this call before it will work.`
- Buttons:** Initialize call (highlighted), Manually enter API response

3. Headers: この API 固有のヘッダー情報

- 今回は特に指定しません

The screenshot shows a configuration panel for an API call named "search". The "Name" field is set to "search". The "Use as" dropdown is set to "Data" and the "Data type" dropdown is set to "JSON". The "Method" dropdown is set to "GET" and the "URL" field contains "https://www.googleapis.com/youtube/v3/search". The "Headers" section is collapsed, indicated by a red circled "③". The "Parameters" section is expanded, indicated by a red circled "④". It contains two parameters: "key" with value "AlzaSyAA3b34Zbc9r zx nw.", and "q" with value "おしゃべり唐揚げ". Both parameters have the "Private" checkbox checked, "Allow blank" unchecked, and "Optional" unchecked. There are "Add header" and "Add parameter" buttons. At the bottom, there are checkboxes for "Include errors in response and allow workflow actions to continue" and "Capture response headers", both of which are unchecked. A warning message "⚠ You need to initialize this call before it will work." is displayed in red. At the very bottom, there are two buttons: "Initialize call" (highlighted with a blue border) and "Manually enter API response".

Name: search
Use as: Data
Data type: JSON
Method: GET
URL: https://www.googleapis.com/youtube/v3/search
(use [] for params)
collapse
①
②
③ Headers
Add header
④ Parameters
Key: key Value: AlzaSyAA3b34Zbc9r zx nw.
Private: Allow blank: Optional:
Key: q Value: おしゃべり唐揚げ
Private: Allow blank: Optional:
Add parameter
Include errors in response and allow workflow actions to continue:
Capture response headers:
⚠ You need to initialize this call before it will work.
Initialize call
Manually enter API response

4. Parameters: この API 固有のパラメータ情報

- 今回は下記 2 つのパラメータを指定します。Private / Allow blank / Optional は添付の通り
 - i. Key: `key` 、 Value: 先ほど Google アカウント側で発行した API キー
 - ii. Key: `q` 、 Value: YouTube で動作検索をする際のキーワード（ご自由に指定ください）

- すべて設定するとこんな感じです

collapse

Name	① search	Use as	Data	Data type	JSON	<input type="button" value="collapse"/>				
GET	② https://www.googleapis.com/youtube/v3/search	(use [] for params)				<input type="button" value="collapse"/>				
③ Headers										
<input type="button" value="Add header"/>										
④ Parameters										
Key	key	Value	AlzaSyAA3b34Zbc9rznw.	Private	<input checked="" type="checkbox"/>	Allow blank	<input type="checkbox"/>	Optional	<input type="checkbox"/>	<input type="button" value="collapse"/>
Key	q	Value	おしゃべり唐揚げ	Private	<input checked="" type="checkbox"/>	Allow blank	<input type="checkbox"/>	Optional	<input type="checkbox"/>	<input type="button" value="collapse"/>
<input type="button" value="Add parameter"/>										
Include errors in response and allow workflow actions to continue <input type="checkbox"/>										
Capture response headers <input type="checkbox"/>										
A You need to initialize this call before it will work.										
<input type="button" value="Initialize call"/>				Manually enter API response						

- 設定が終わったら、設定内容が正しいかを確認します
- 下部にある "Initialize call" ボタンをクリック

- おそらく Status code 403 のエラーメッセージが返ってきたはずです
- 内容は書いてある通りですが、要約するとこんな感じです

YouTube Data API v3 はあなたのプロジェクトで無効になっています。
この URL にアクセスして有効にしてから再試行してください。

API Connector

There was an issue setting up your call.

Raw response for the API

Status code 403

```
{  
  "error": {  
    "code": 403,  
    "message": "YouTube Data API v3 has not been used in project 298727815636 before or it is disabled. Enable it by visiting https://console.developers.google.com/apis/api/youtube.googleapis.com/overview?project=298727815636 then retry. If you enabled this API recently, wait a few minutes for the action to propagate to our systems and retry.",  
    "errors": [  
      {  
        "message": "YouTube Data API v3 has not been used in project 298727815636 before or it is disabled. Enable it by visiting https://console.developers.google.com/apis/api/youtube.googleapis.com/overview?project=298727815636 then retry. If you enabled this API recently, wait a few minutes for the action to propagate to our systems and retry.",  
        "domain": "usageLimits",  
        "reason": "accessNotConfigured",  
        "extendedHelp": "https://console.developers.google.com"  
      }  
    ],  
    "status": "PERMISSION_DENIED",  
    "details": [  
      {  
        "@type": "type.googleapis.com/google.rpc.Help",  
        "links": [  
          {  
            "text": "View help center",  
            "url": "https://cloud.google.com/apis/docs/getting-started"  
          }  
        ]  
      }  
    ]  
  }  
}
```

- どうやら今回利用した YouTube Data API が各人の Google アカウント上で有効になっていないようです
- これは Google さんが全員が使わないであろう機能は最初から OFF にしていて、必要ある人だけ自分で ON にしてね、という状態になっています
- なので今回は YouTube Data API を使いたいので、メッセージに記載されている URL にアクセスして有効化しましょう

[https://console.developers.google.com/apis/api/youtube.googleapis.com/overview?
project={各人のプロジェクトID}](https://console.developers.google.com/apis/api/youtube.googleapis.com/overview?project={各人のプロジェクトID})

- すると親切に今回利用したい "YouTube Data API v3" の画面へ遷移したはずです
- 画面にあるとおり「有効にする」をクリックして有効にします



- しばらくするとこんな画面に遷移するはずです
- 画面上部の赤線部分が「API を無効にする」となっていれば有効化は成功です

API API とサービス [← API / サービスの詳細](#) [■ API を無効にする](#)

有効な API とサービス		
 YouTube Data API v3		
The YouTube Data API v3 is an API that provides access to YouTube data, such as videos, playlists, and channels.		
所有者: Google		
サービス名: youtube.googleapis.com	タイプ: 公開 API	ステータス: 有効
LEARN MORE	API EXPLORE	すい
指標	割り当て	認証情報
グラフを選択 — 4 個のグラフ		
30 日 ▾		

- YouTube Data API が有効化できたので、再度 Bubble の画面に戻り "Initialize call" ボタンをクリック
- 成功すると "Returned values - search" というポップアップが表示されるはずです

Returned values - search

You can modify the data types that are returned by the call. This affects how you can use the data in Bubble. If you chose 'Ignore field', the fields won't be shown in the dropdowns.

kind youtube#searchListResponse	<input type="text"/>
etag bMKu0aiX56rekNwT5RVHZLOYR-o	<input type="text"/>
nextPageToken CAUQAA	<input type="text"/>
regionCode US	<input type="text"/>
pageInfo totalResults 1000000	<input type="number"/>
pageInfo resultsPerPage 5	<input type="number"/>
items (list) (see fields below)	<input type="text"/> search item
kind youtube#searchResult	<input type="text"/>
etag WahfcvNgYipX-i3rjRIPUrHi10	<input type="text"/>
id kind youtube#channel	<input type="text"/>
id channelId UCcRigzl_jBAZh_UySjv8xuw	<input type="text"/>
id videoId L2SaFPQI4_g	<input type="text"/>

- これは今回設定した
["https://www.googleapis.com/youtube/v3/search"](https://www.googleapis.com/youtube/v3/search) の API を実行した
 結果（レスポンス）の情報を表しています

Returned values - search

You can modify the data types that are returned by the call. This affects how you can use the data in Bubble. If you chose 'Ignore field', the fields won't be shown in the dropdowns.

kind	youtube#searchListResponse	<input type="text"/>
etag	mjrgGMgtIMWG6aVWolrNtjmgtG8	<input type="text"/>
nextPageToken	CAUQAA	<input type="text"/>
regionCode	US	<input type="text"/>
pageInfo totalResults	946629	<input type="number"/>
pageInfo resultsPerPage	5	<input type="number"/>
items (list)	(see fields below)	<input type="text"/>
kind	youtube#searchResult	<input type="text"/>
etag	WahfIcVNgyipX-l3rjRiPUrHi10	<input type="text"/>
id kind	youtube#channel	<input type="text"/>
id channelId	UCcRigzI_jBAZh_UySjv8xuw	<input type="text"/>
id videoId	S094aNshTSU	<input type="text"/>

詳しくは割愛しますが、ポイントとしては下記 3 つです。

- `items(list)` となっている部分が検索結果の動画一覧が格納されている一覧になります
- その型となるのが `search item` という型となり、検索結果に含まれる複数系のデータであることを表しています
- そして、この後の設定で特に重要なのが `id videoId` となっている項目です

- YouTubeを見たことがある人なら分かると思いますが、このVideoIdというのが、それぞれの動画を一意に識別する情報となっており、画面表示するときにもこのIdが必要となります
- その他、特に参照しない項目については紛らわしいのでプルダウンの中から `Ignore field` を選択しておきます
- こうすることで、Dynamic dataとして扱う時に、選択肢の中に表示しないようになり、設定項目を選ぶ際に悩まずに済みます

- すべて設定するとこんな感じです
- 設定が完了したら "SAVE" をクリックして、設定内容を保存しておきます

Returned values - search

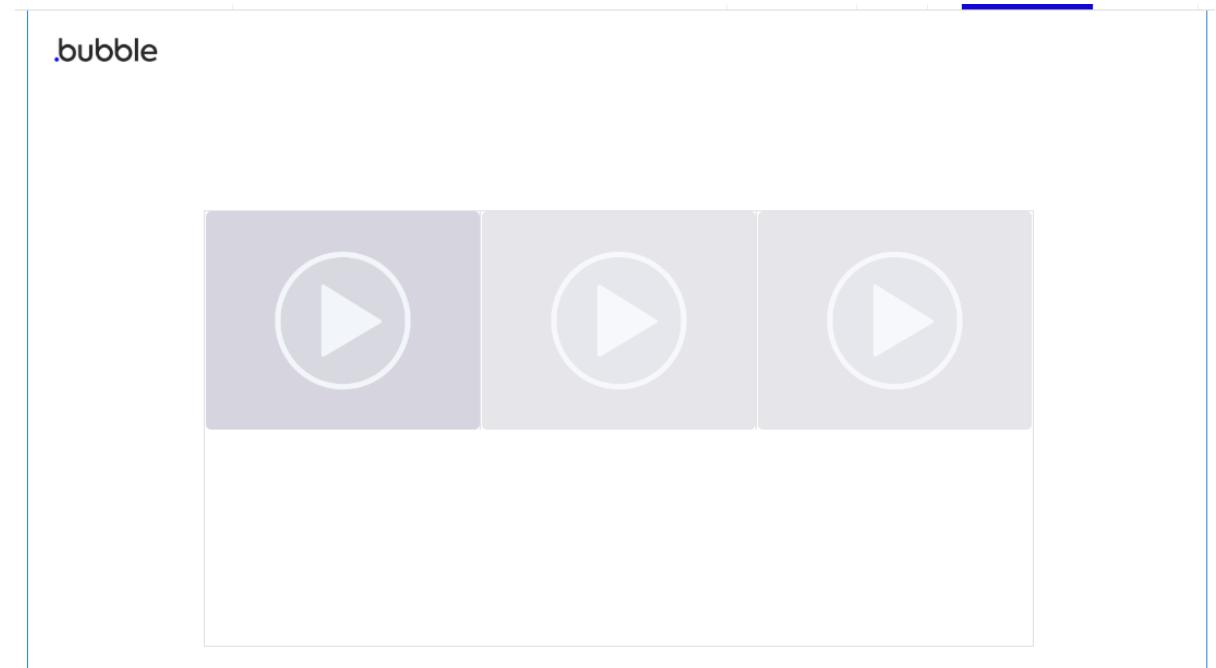
You can modify the data types that are returned by the call. This affects how you can use the data in Bubble. If you chose 'Ignore field', the fields won't be shown in the dropdowns.

kind youtube#searchListResponse	Ignore field
etag mjrgGMgtIMWG6aVWolrNtjmgG8	Ignore field
nextPageToken CAUQAA	Ignore field
regionCode US	Ignore field
pageInfo totalResults 946629	Ignore field
pageInfo resultsPerPage 5	Ignore field
items (list) (see fields below)	search item
kind youtube#searchResult	Ignore field
etag WahfICvNgYipX-l3rjRIPUrHi10	Ignore field
id kind youtube#channel	Ignore field
id channelId UCcRigzl_jBAZh_UySjv8xuw	Ignore field
id videoId SO94aNshTSU	text

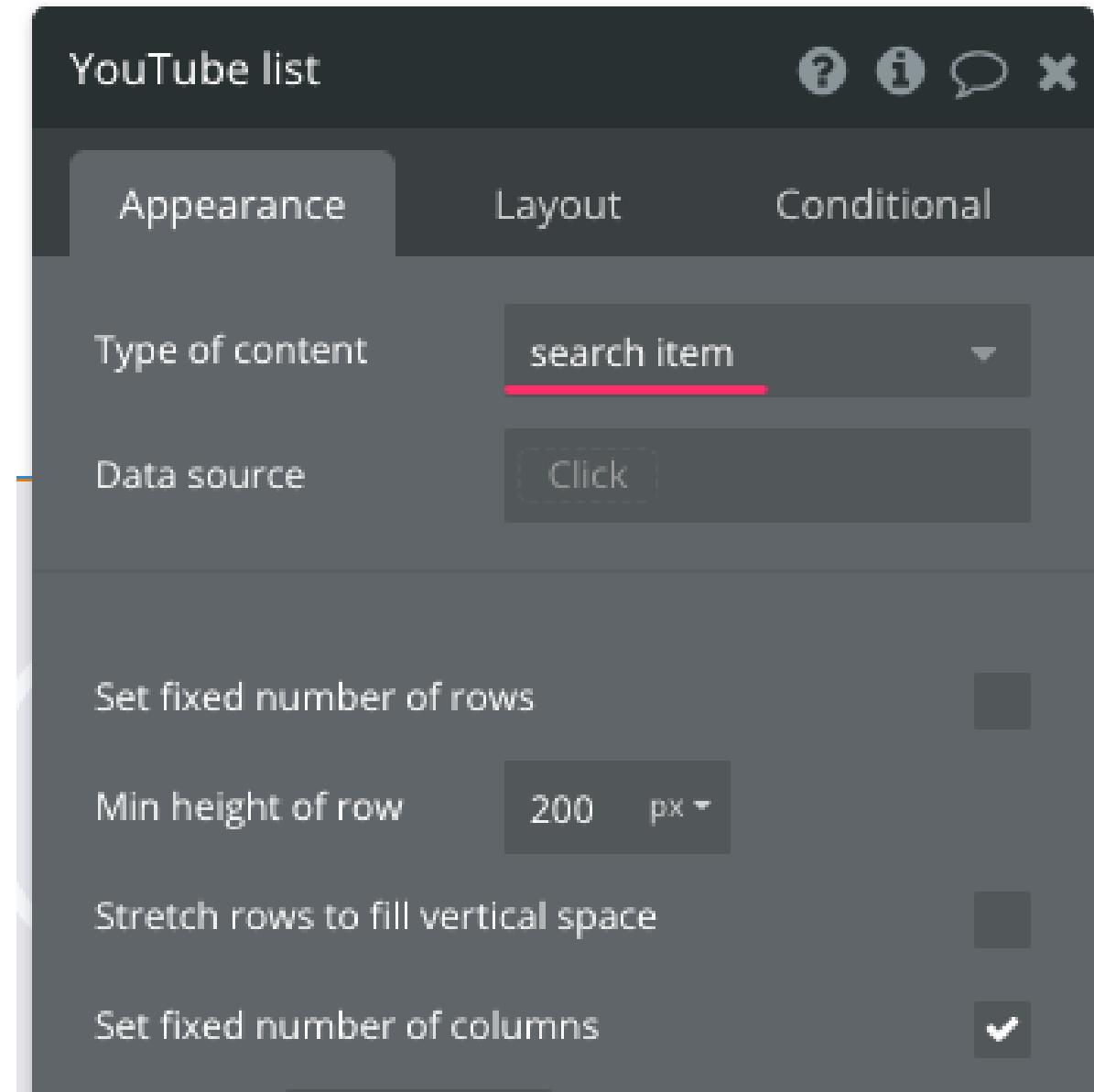
画面表示の準備

- Bubble の API 連携の準備が出来たので、いよいよ画面の設定をしていきます
- 画面の設定は前回と今回の講義でマスターしていると思いますので、概要だけお伝えして画面レイアウトを組み立ててみてください

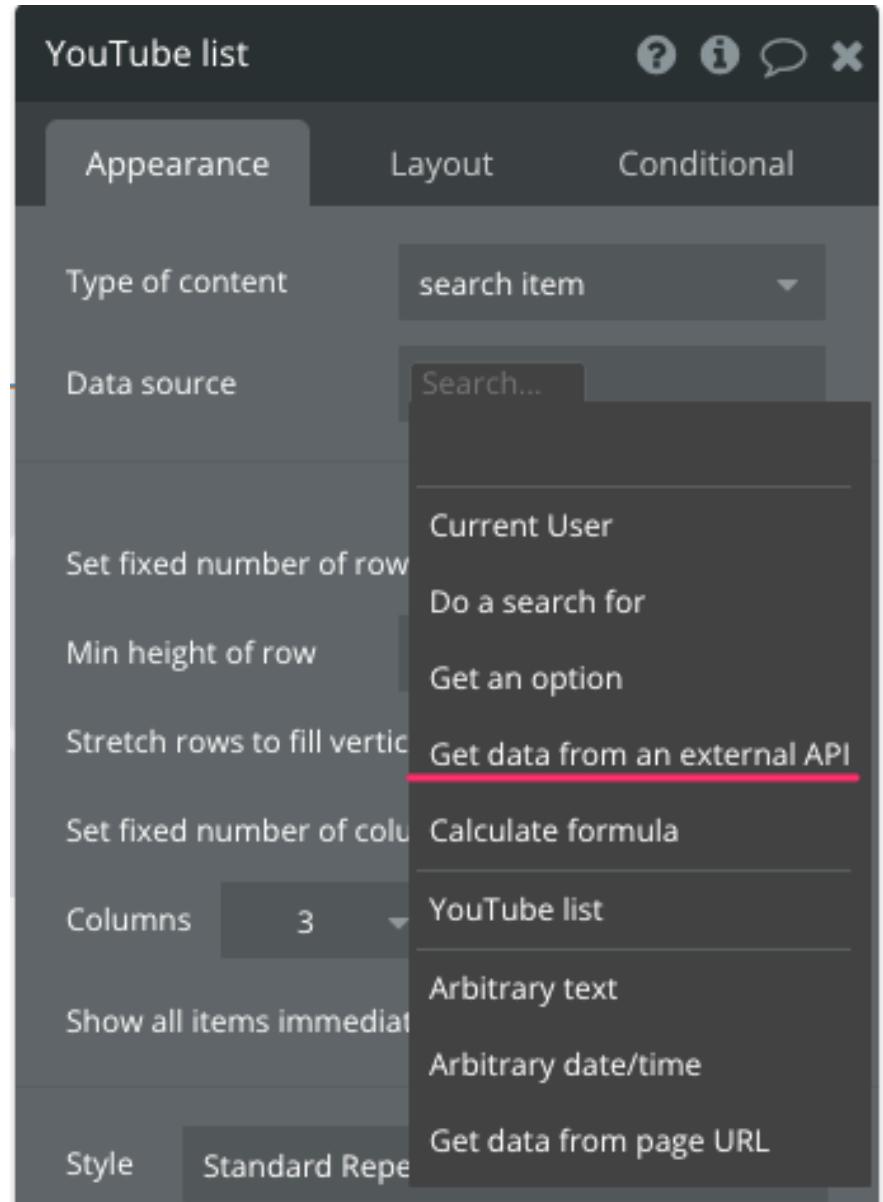
- pet_list ページを clone する形で "video_list" 画面を作成
- Repeating Group の各セルの中身を全て削除
- 代わりに Visual elements の Video をセル内いっぱいになるよう設置
- こんな感じですね



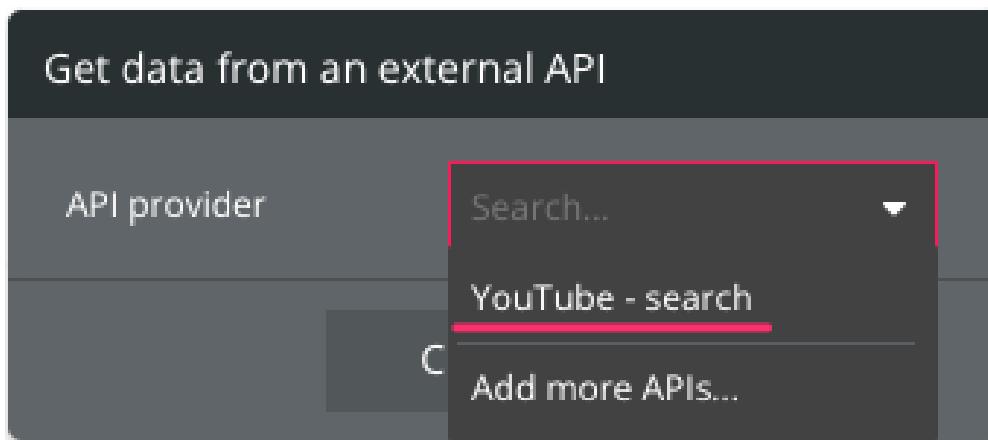
- レイアウトが出来たので実際に YouTube の一覧表示を設定していきます
- まず Repeating Group の Type of content は、先ほど API 実行時に確認した `search item` 型となります
- これで繰り返しするデータの種類が YouTube の動画 1つ1つとなります



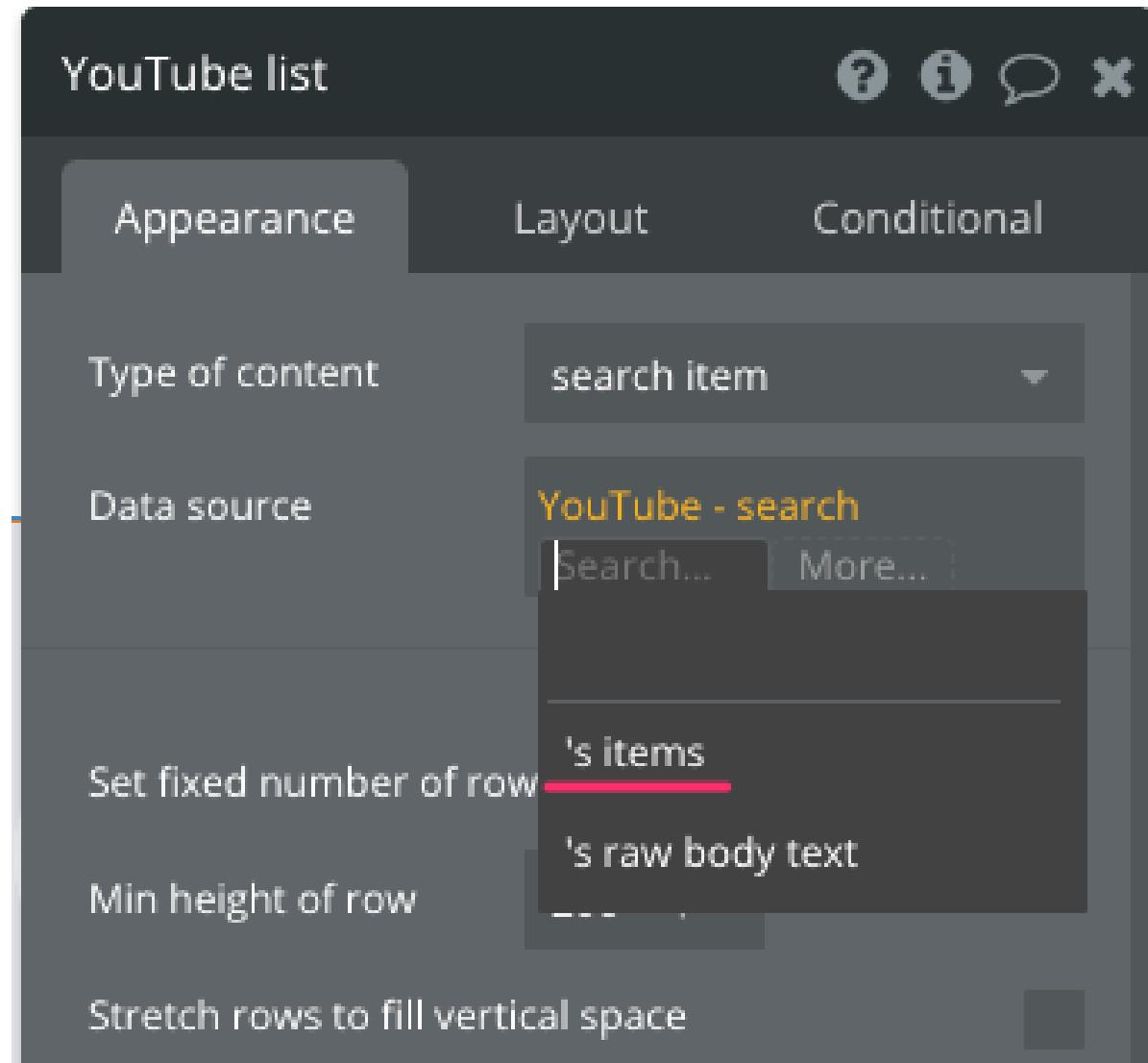
- 次に Data source ですが、今回は API から取得した YouTube の動画一覧を対象とします
- Click から候補を見てみると Get data from an external API という項目があると思うのでそちらを選択



- 隣に Get data from an external API のポップアップが表示されるので、API provider の プルダウンを選んでください
- すると、プルダウンの中に先ほど設定した YouTube – search の候補があるのであると思いますのでそれを選択してください
 - ハイフンの前が API のグループ名で、後ろが具体的な API の名前です



- API Provider が設定できたら、CLOSE で閉じます
- そして元の Repeating Group の Data source の選択として、さらに 's items を選択します
- これで、API の結果に含まれる動画一覧 (items) を繰り返し表示する、という指定ができました

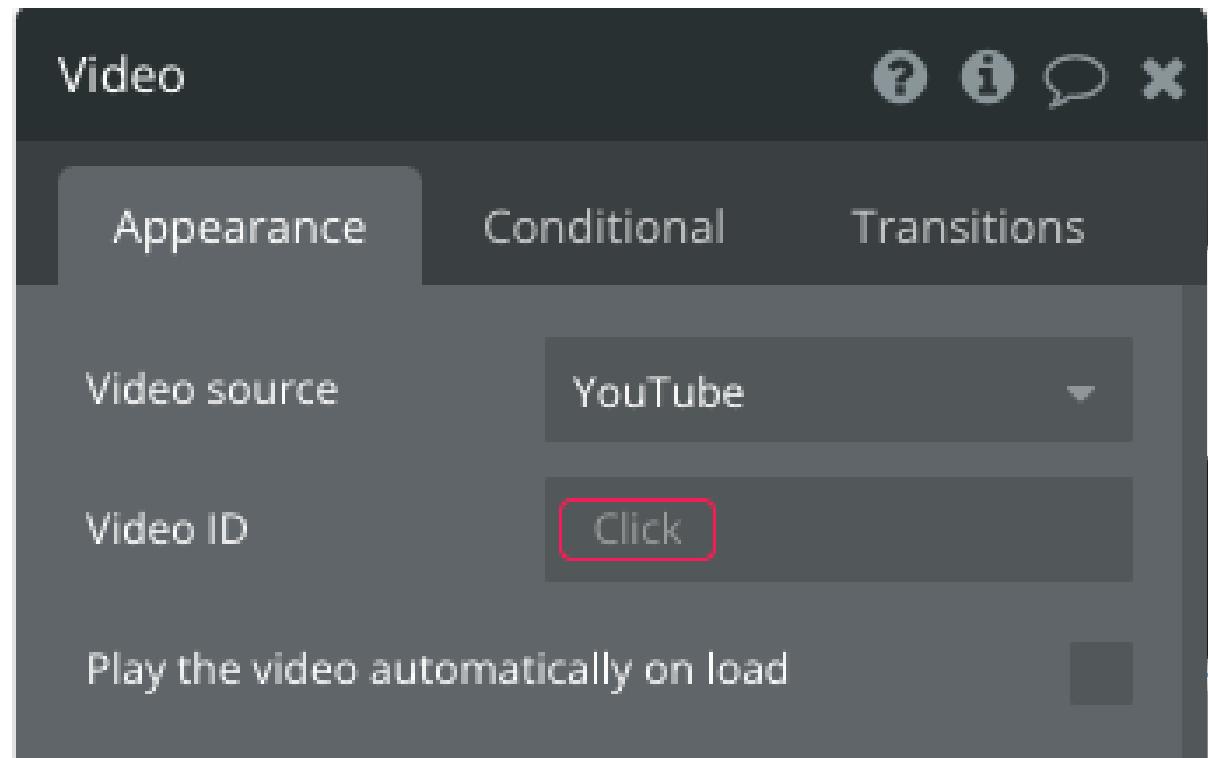


- 次に各セルの設定をしていきますが、まずは皆さん設定してみましょう

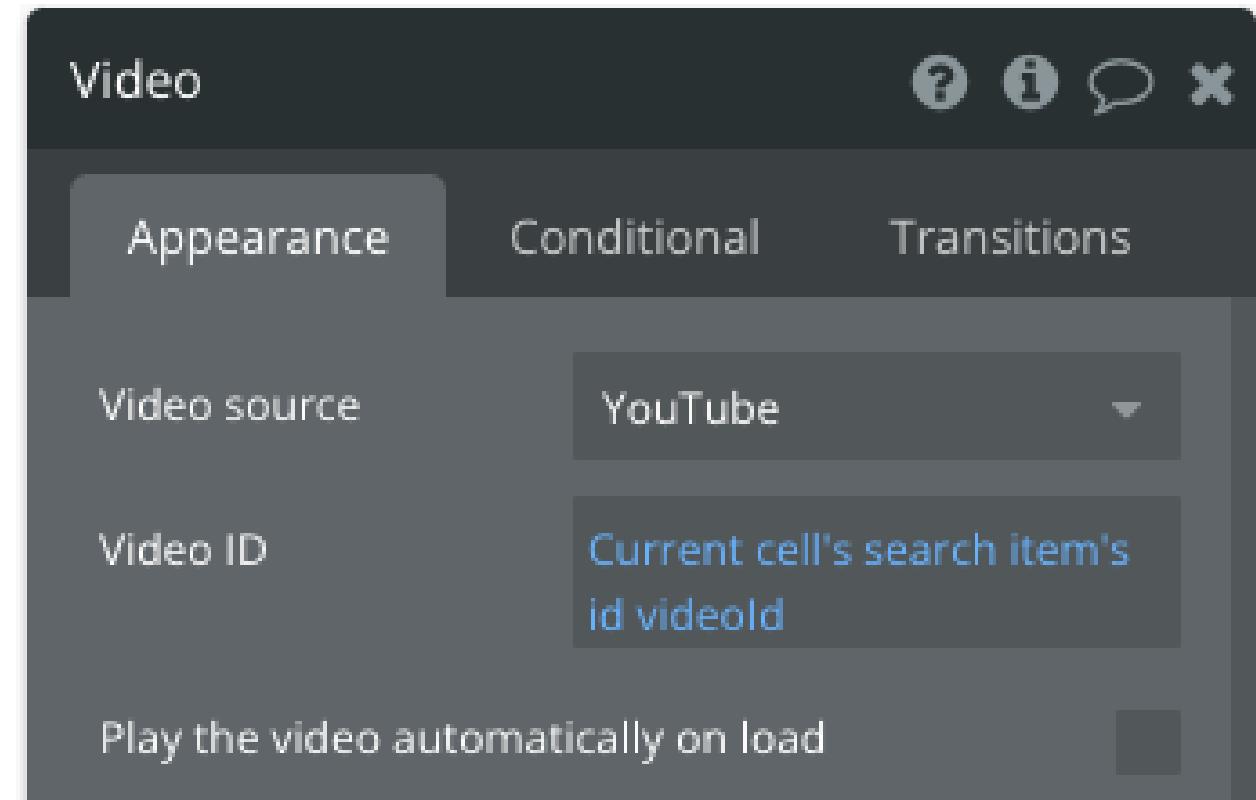
- Hint 

- Video source は YouTube
 - すると Video ID という項目が表示されるので、先ほど API 設定をしたときにみた

Video Id を参照すればよさ
そうですね



- こんな感じですね



- これで設定完了です！さっそくプレビューしてみましょう！
- 事前に API で指定したキーワードの動画一覧が表示されましたか？



Google ログインを試してみよう

- 続いて Bubble アプリへのログインを Google アカウントを使ってログインしてみましょう！
- まずは Google のプラグインをインストール

The screenshot shows the 'Install New Plugins' page on the Bubble platform. A search bar at the top contains the text 'Google'. On the left, there are two filter sections: 'Types' and 'Categories'. Under 'Types', 'Google' is selected. Under 'Categories', several checkboxes are checked, including 'Analytics', 'Blog', 'Compliance', 'Containers', 'Customer Support', 'Data (things)', 'Ecommerce', 'Email', 'Energy', 'Health & Fitness', 'Input Forms', 'Internationalization', 'Location', 'Media', and 'Mobile'. To the right, a list of plugins is displayed. The first plugin is 'Google Material Icons' with 276,535 apps, followed by 'Google Analytics' with 33,048 apps. The third plugin in the list is 'Google', which is highlighted with a red border. Its description states: 'Let users log in with their Gmail account, and fetch their profile data.' It has 32,150 apps. Below it is 'Google Places' with 22,982 apps. At the bottom right of the page is a blue 'DONE' button.

Install New Plugins

Filters

Sort by: name usage rating date price

Google

Types (deselect all)

- Action
- Api
- Background Services
- Element
- Event
- Login Service
- Storage

Categories (deselect all)

- Analytics
- Blog
- Compliance
- Containers
- Customer Support
- Data (things)
- Ecommerce
- Email
- Energy
- Health & Fitness
- Input Forms
- Internationalization
- Location
- Media
- Mobile

Google Material Icons

Material design system library is a free library of icons built by Google, using Material Design principles.

By Bubble

276,535 apps

Install

Google Analytics

Google Analytics lets you track visits on your website and get some insights on your audience for free.

By Bubble

33,048 apps

Install

Google

Let users log in with their Gmail account, and fetch their profile data.

By Bubble

32,150 apps

Install

Google Places

Fetch restaurant, schools, etc. from the Google Places API.

By Bubble

22,982 apps

Install

DONE

⚠ Recurring subscriptions to paid plugins are on paid plans only. Click here to adjust your application plan.

- インストールした Google プラグインを選択
- 右パネルに表示されている `Use a generic redirect URL` の欄にチェックを入れる
 - この URL もメモしておくのですが、画面上で選択ができないため、下記 URL のアプリ名部分だけご自身のアプリ名に読み替えてメモしておいてください 😊
 - `https://{{Your App Name}}.bubbleapps.io/api/1.1/oauth_redirect`



Google

Service page →

Let users log in with their Gmail account, and fetch their profile data.

App Secret

App ID/API Key

Use a generic redirect URL (https://advanced-bubble-kyogoku.bubbleapps.io/api/1.1/oauth_redirect)



App Secret - dev.

(optional)

App ID/API Key - dev.

(optional)

Plugin content

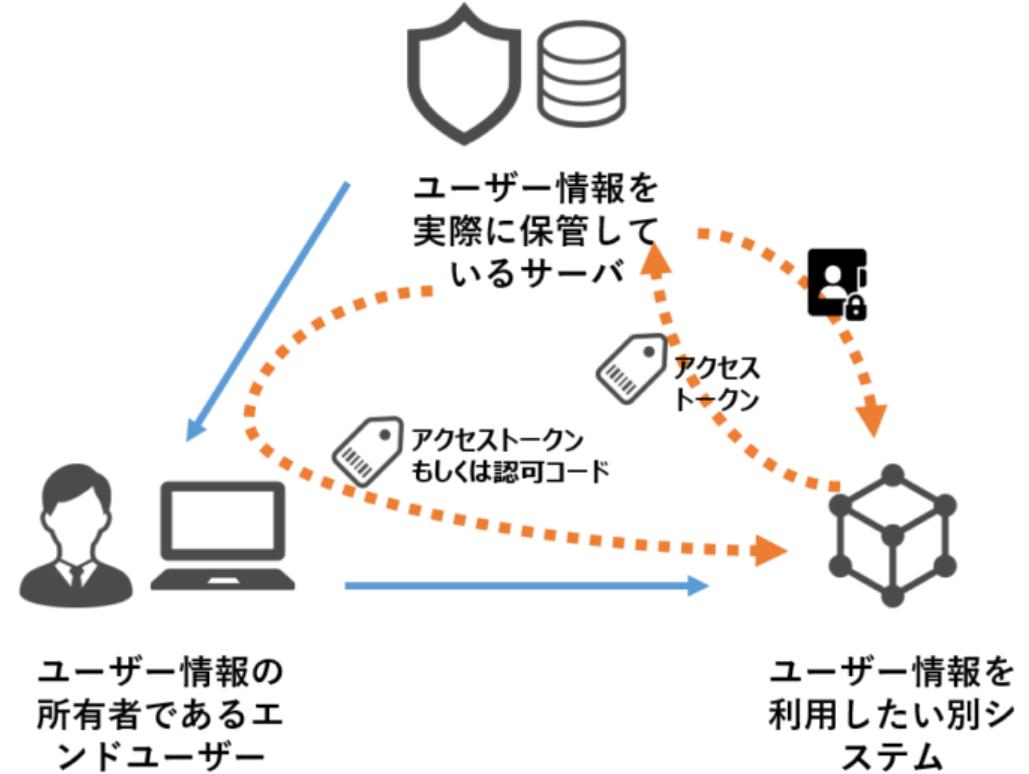
This plugin is an authentication provider.

OAuthとは何か

OAuthはインターネット上で、あるサービスが別のサービスにパスワードを明かすことなく、認証とリソースアクセスを可能にするための仕組みになります。あなたが他のサービス（例えばペット管理サービス）を使いたいけど、そのために新しいアカウントを作りたくないとき、OAuthを使って既存のアカウント（例えばGoogleやFacebookのアカウント）でログインできます。

どうやって動くのか

- ログインのリクエスト: ペット管理サービスに「Googleアカウントでログイン」を選ぶ。
- 許可の確認: Googleが「このサービスにあなたの情報の一部を使ってもいいか」と尋ねる。
- 「鍵」の渡し: 許可すると、Googleはそのサービスに特別な「鍵」を渡す。すると、そのサービスはあなたの名前やメールアドレスなど、限られた情報にアクセスできるようになります。



Googleへのログイン後にアプリに戻ってくるための認証準備

- Google Cloud Platform 画面にアクセス

<https://console.developers.google.com/>

- 先程と同じ API とサービスの画面を表示

- プロジェクトを選択し、左メニューから "OAuth 同意画面" を選択

The screenshot shows the Google Cloud Platform interface for configuring an OAuth consent screen. The left sidebar has a 'OAuth 同意画面' tab selected. The main area displays the configuration for the 'User Type' section, specifically for '内部' (Internal) users. It includes descriptive text about how the app is configured and registered, and links to detailed user type information. A large blue '作成' (Create) button is at the bottom.

API API とサービス

OAuth 同意画面

有効な API とサービス

アプリをどのように構成および登録するか（ターゲット ユーザーを含む）を選択します。プロジェクトに関連付けることができるアプリは 1 つだけです。

ライブラリ

認証情報

OAuth 同意画面

ページの使用に関する契約

User Type

内部 [?](#)

組織内のユーザーのみが使用できます。確認を受けるためにアプリを送信する必要はありません。 [ユーザーの種類の詳細](#)

外部 [?](#)

Google アカウントを持つすべてのテストユーザーが使用できます。アプリはテストモードで起動し、アプリを使用できるのは、テストユーザーのリストに追加されたユーザーに限られます。アプリを本番環境に移す準備ができたら、アプリの確認が必要となる場合があります。 [ユーザーの種類の詳細](#)

作成

Google の OAuth に関する [ご意見やご要望をお聞かせください](#)。

- User Type に "外部" を選択して作成をクリック

API API とサービス OAuth 同意画面

◆ 有効な API とサービス
■ ライブラリ
○ 認証情報
□ OAuth 同意画面
≡ ページの使用に関する契約

User Type

内部 ?
 外部 ?

組織内のユーザーのみが使用できます。確認を受けるためにアプリを送信する必要はありません。 [ユーザーの種類の詳細](#)

Google アカウントを持つすべてのテストユーザーが使用できます。アプリはテストモードで起動し、アプリを使用できるのは、テストユーザーのリストに追加されたユーザーに限られます。アプリを本番環境に移す準備ができたら、アプリの確認が必要となる場合があります。 [ユーザーの種類の詳細](#)

作成

Google の OAuth に関する [ご意見やご要望をお聞かせください。](#)



- 次ページで必要事項を入力していきます

- アプリ情報
 - アプリ名: bubble で作成したアプリ名
 - ユーザーサポートメール: ご自身の Google アカウントのメールアドレス

アプリ登録の編集

① OAuth 同意画面 — ② スコープ — ③ テストユーザー — ④ 概要

アプリ情報

この情報は同意画面に表示されるため、デベロッパーのユーザー情報とデベロッパーへの問い合わせ方法をエンドユーザーが把握できます。

アプリ名*
titech2022-advance-naotake

同意を求めるアプリの名前

ユーザー サポートメール*
kyougoku2bubble@gmail.com

ユーザーが同意に関して問い合わせるために使用

アプリのロゴ

参照

ユーザーがアプリを認識できるように、同意画面に 1 MB 以下の画像をアップロードします。使用できる画像形式は、JPG、PNG、BMP です。最適な結果を得るには、ロゴを 120 x 120 ピクセルの正方形にすることをおすすめします。

- 承認済みドメイン

- bubbleapps.io

- デベロッパーの連絡先情報

- ご自身の Google アカウント
のメールアドレス

- "保存して次へ"

承認済みドメイン ?

同意画面または OAuth クライアントの構成でドメインが使用されている場合は、ここで事前登録する必要があります。アプリの検証が必要な場合は、[Google Search Console](#) にアクセスして、ドメインが承認済みであるかどうかを確認してください。承認済みドメインの上限の[詳細](#)をご覧ください。

承認済みドメイン 1*

bubbleapps.io

+ ドメインの追加

デベロッパーの連絡先情報

メールアドレス *

kyougoku2bubble@gmail.com x

これらのメールアドレスは、プロジェクトの変更について Google からお知らせするために使用します。

保存して次へ

キャンセル

- スコープは特に設定変更せず "保存して次へ"

- テストユーザーにはご自身の Google メールアドレスを入力しておく
- 設定したら保存して次へ

アプリ登録の編集

✓ OAuth 同意画面 — ✓ スコープ — 3 テストユーザー — 4 概要

テストユーザー

公開ステータスが「テスト中」に設定されている間は、テストユーザーのみがアプリにアクセスできます。アプリの確認前の許可済みユーザー数の上限は 100 で、この上限はアプリの全期間でカウントされます。[詳細](#)

+ ADD USERS

≡ フィルタ プロパティ名または値を入力 ?

ユーザー情報

kyougoku2bubble@gmail.com



保存して次へ

キャンセル

- これで登録は完了したので続いて認証情報を発行します
- 左メニューから認証情報を選択して、「+ 認証情報を作成」をクリック
- サブメニューから OAuth クライアント ID を選択

API API とサービス	認証情報	+ 認証情報を作成 削除
❖ 有効な API とサービス 盤 ライブラリ 🕒 認証情報 🕒 OAuth 同意画面 ≡❖ ページの使用に関する契約	<p>有効な API にアクセス</p> <p>API キー</p> <p>OAuth クライアント ID</p> <p>サービス アカウント</p> <p>OAuth 2.0 クラ</p> <p>表示する OAuth クライアントがありません</p>	<p>API キー シンプル API キーを使用してプロジェクトを識別し、割り当てとアクセスを確認します</p> <p>OAuth クライアント ID ユーザーのデータにアクセスできるようにユーザーの同意をリクエストします</p> <p>サービス アカウント ロボット アカウントによるサーバー間でのアプリレベルの認証を有効にします</p> <p>ウィザードで選択 使用する認証情報の種類を決定するため、いくつかの質問をします</p>

- OAuth クライアント ID の作成画面
が表示されるので下記の通り入力
し「保存」
 - アプリケーションの種類: ウェブアプリケーション
 - 名前: Bubble OAuth

← OAuth クライアント ID の作成

クライアント ID は、Google の OAuth サーバーで個々のアプリを識別するために使用します。アプリが複数のプラットフォームで実行される場合、それぞれに独自のクライアント ID が必要になります。詳しくは、[OAuth 2.0 の設定](#)をご覧ください。OAuth クライアントの種類の[詳細](#)

アプリケーションの種類*
ウェブアプリケーション

名前*
Bubble OAuth
OAuth 2.0 クライアントの名前。この名前はコンソールでクライアントを識別するためにのみ使用され、エンドユーザーには表示されません。

● 下で追加する URI のドメインは、[OAuth 同意画面](#)に承認済みドメインとして自動で追加されます。

承認済みの JavaScript 生成元 [?](#)
ブラウザからのリクエストに使用します
[+ URI を追加](#)

承認済みのリダイレクト URI [?](#)
ウェブサーバーからのリクエストに使用します
URI 1*
https://titech2022-advance-naotake.bubbleapps.io/api/1.1/oauth_redirect
[+ URI を追加](#)

注: 設定が有効になるまで 5 分から数時間かかることがあります

[作成](#) [キャンセル](#)

- 承認済みのリダイレクト URI:
Bubble 側で Google プラグインインストール時に控えた URL

```
https://{{Your App  
Name}}.bubbleapps.io/api/1.1/  
oauth_redirect
```

← OAuth クライアント ID の作成

クライアント ID は、Google の OAuth サーバーで個々のアプリを識別するために使用します。アプリが複数のプラットフォームで実行される場合、それぞれに独自のクライアント ID が必要になります。詳しくは、[OAuth 2.0 の設定](#)をご覧ください。OAuth クライアントの種類の[詳細](#)

アプリケーションの種類*
ウェブアプリケーション

名前*
Bubble OAuth

OAuth 2.0 クライアントの名前。この名前はコンソールでクライアントを識別するためにのみ使用され、エンドユーザーには表示されません。

ⓘ 下で追加する URI のドメインは、[OAuth 同意画面](#)に承認済みドメインとして自動で追加されます。

承認済みの JavaScript 生成元 ?
ブラウザからのリクエストに使用します

[+ URI を追加](#)

承認済みのリダイレクト URI ?
ウェブサーバーからのリクエストに使用します

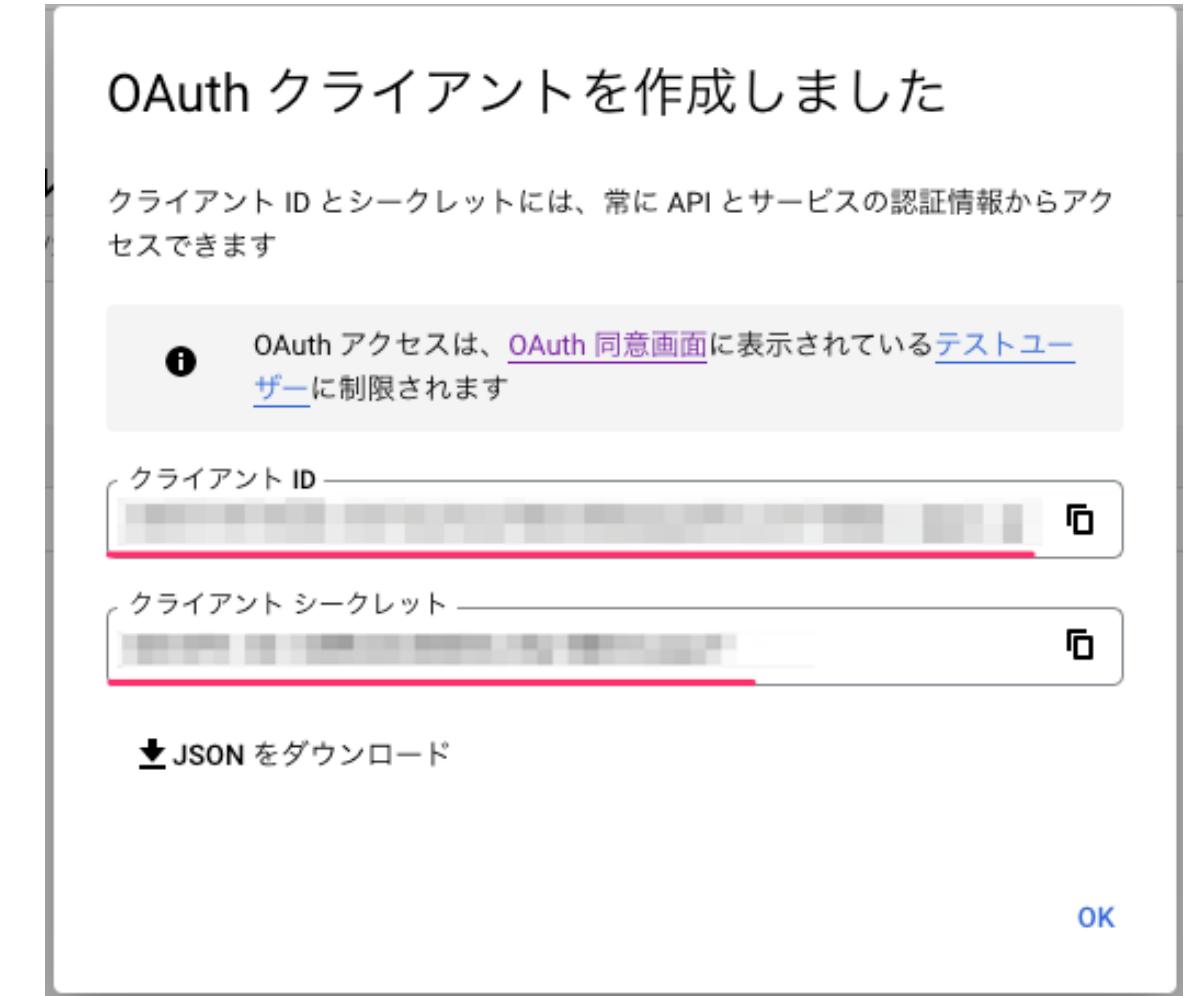
URI 1*
https://titech2022-advance-naotake.bubbleapps.io/api/1.1/oauth_redirect

[+ URI を追加](#)

注: 設定が有効になるまで 5 分から数時間かかることがあります

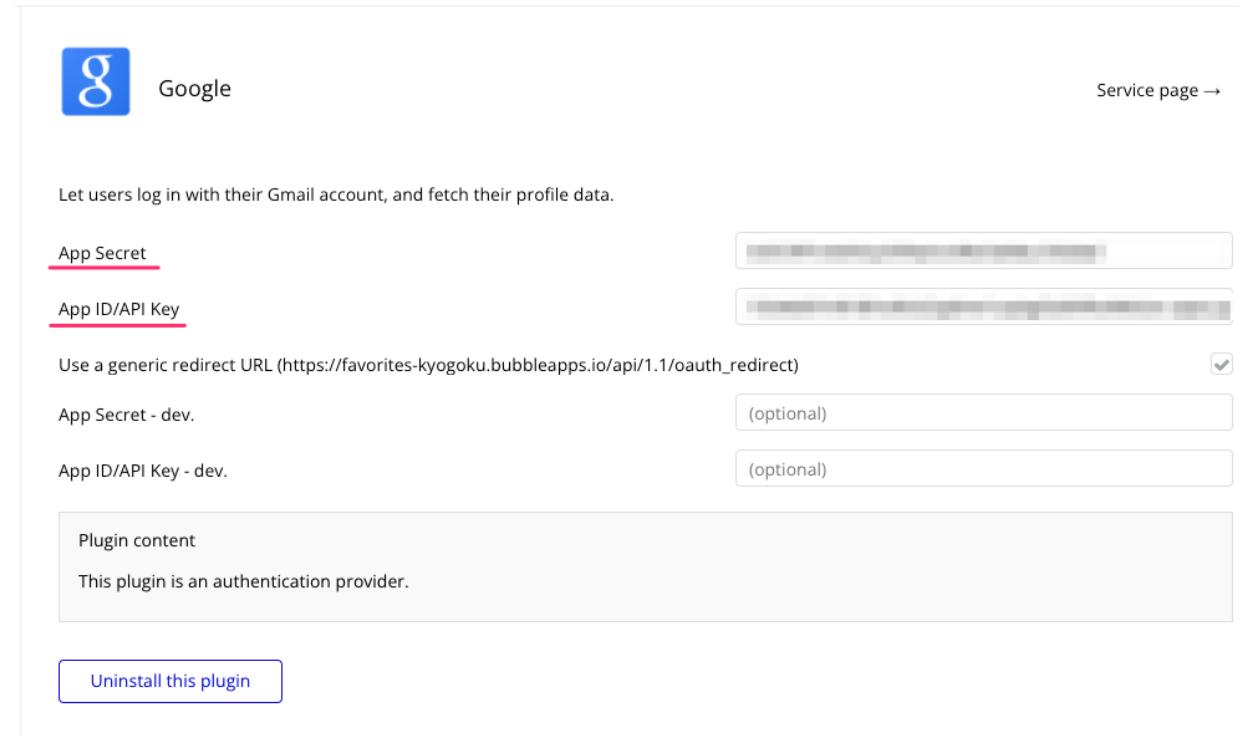
[作成](#) [キャンセル](#)

- "作成" クリック
- すると「OAuth クライアントを作成しました」というポップアップが表示され、そこにクライアント ID とクライアントシークレットが表示されているので、それをメモしておく
 - メモだと忘れるかもしれない場合は JSON をダウンロードしても OK
- メモしたら OK でクローズ



発行した認証情報を Bubble 側に設定

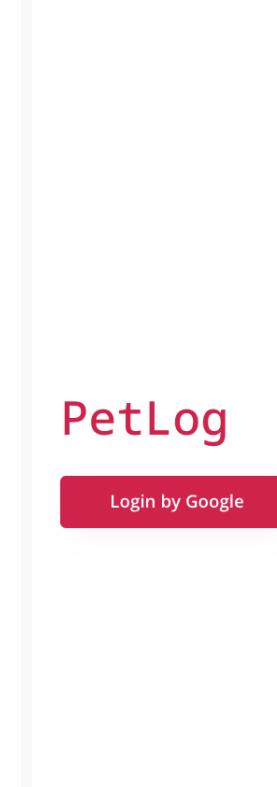
- Bubble の Google プラグイン画面に戻り、先ほど取得したキーをそれぞれの項目へ入力する
 - クライアントID → AppID/API Key
 - クライアントシークレット → App Secret



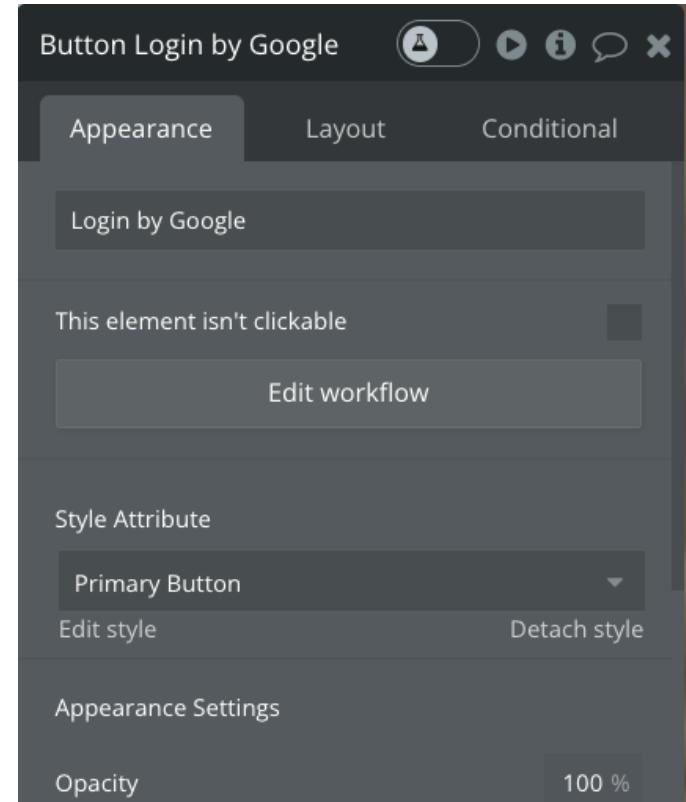
- これで事前設定は完了
- 続いてログイン機能を作り込んでいきます

ログイン機構を Google に置き換える

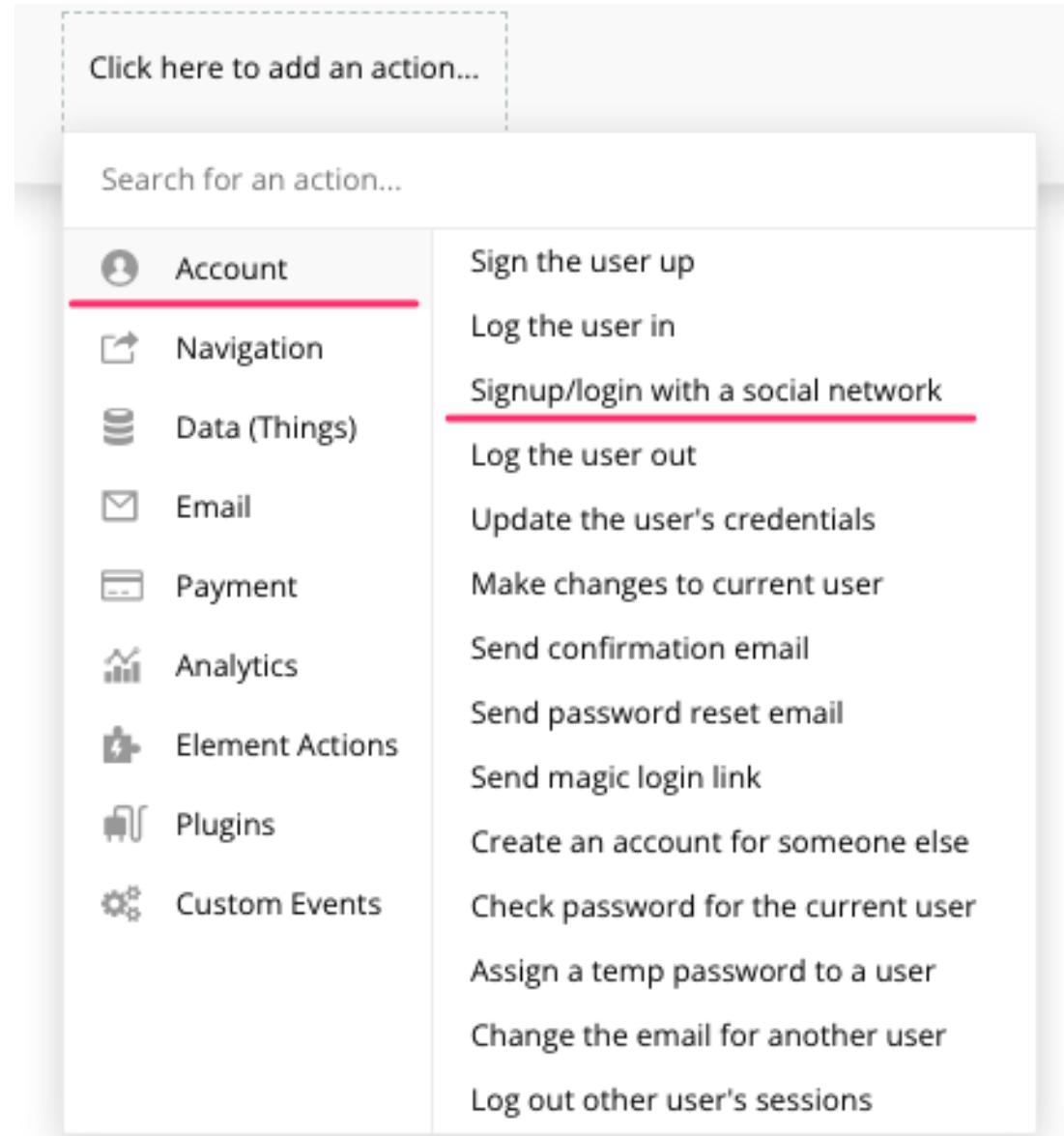
- index画面を開き、サインアップのテキストボックスやボタンをすべて削除しましょう
- 新たにボタンを置いて、`Login by Google` といったラベルに変更しましょう



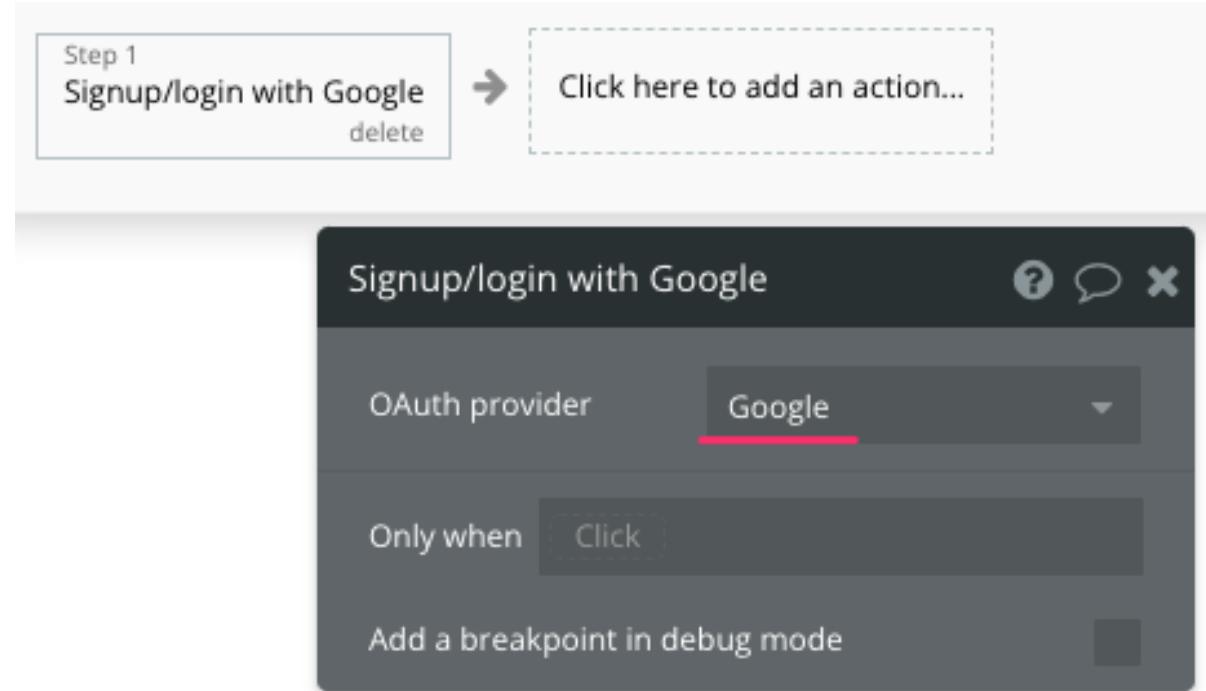
- 設置したボタンの設定ウィンドウで **Edit workflow** をクリックして、ボタンクリック時のワークフロー作成を開始しましょう



- Google ログインのワークフローを設定していきます
- Click here to add an action... を選択
- Account から Signup/login with a social network を選択



- Signup/login with Google のポップアップが表示されます
- OAuth provider に Google を選択



ログインができた後に、ペットリストに遷移するように指定しましょう

- Click here to add an action... を選択
- Navigation > Go to page を指定する
- Destinationに pet_list を指定する

これだけでGoogleアカウントでのログインができるようになりました。簡単ですね。

- 折角なのでログイン成功したらヘッダー上部にアカウント名と画像を表示してみましょう

Pet Register



Log out

- Reusable elements の header を開きます

The screenshot shows the Bluebeam interface with the following elements:

- Search bar:** "Search a page or a reusable..." with a dropdown arrow.
- Navigation:** "Pick an element..." with a dropdown arrow, "Edit", and "Saved" buttons.
- Pages:** A list of pages including "index", "pet_list", "pet_weight_register", "reset_pw", and "404".
- Reusable elements:** A list of reusable elements including "pet_detail", "pet_register", "video_list", and "header".
- Add buttons:** "Add a new page..." and "Add a new reusable element...".

A blue underline highlights the word "header" in the reusable elements list.

- Design タブに切り替え、現在非表示になっている下記 3 つのコンポーネントを Elements tree から表示しておきます
 - こうしておかないと、画像追加後のイメージが分かりづらいため

UI Builder Responsive

Design

Workflow

Data

Styles

Plugins

Only show hideable

Reusable header

- Group Header Standard wi...

+ GroupFocus Menu

- Group Header Standard

Image Logo

- Group Menu

Icon Menu

- Group Buttons

Link Pet Register

Button Log In (Head...)

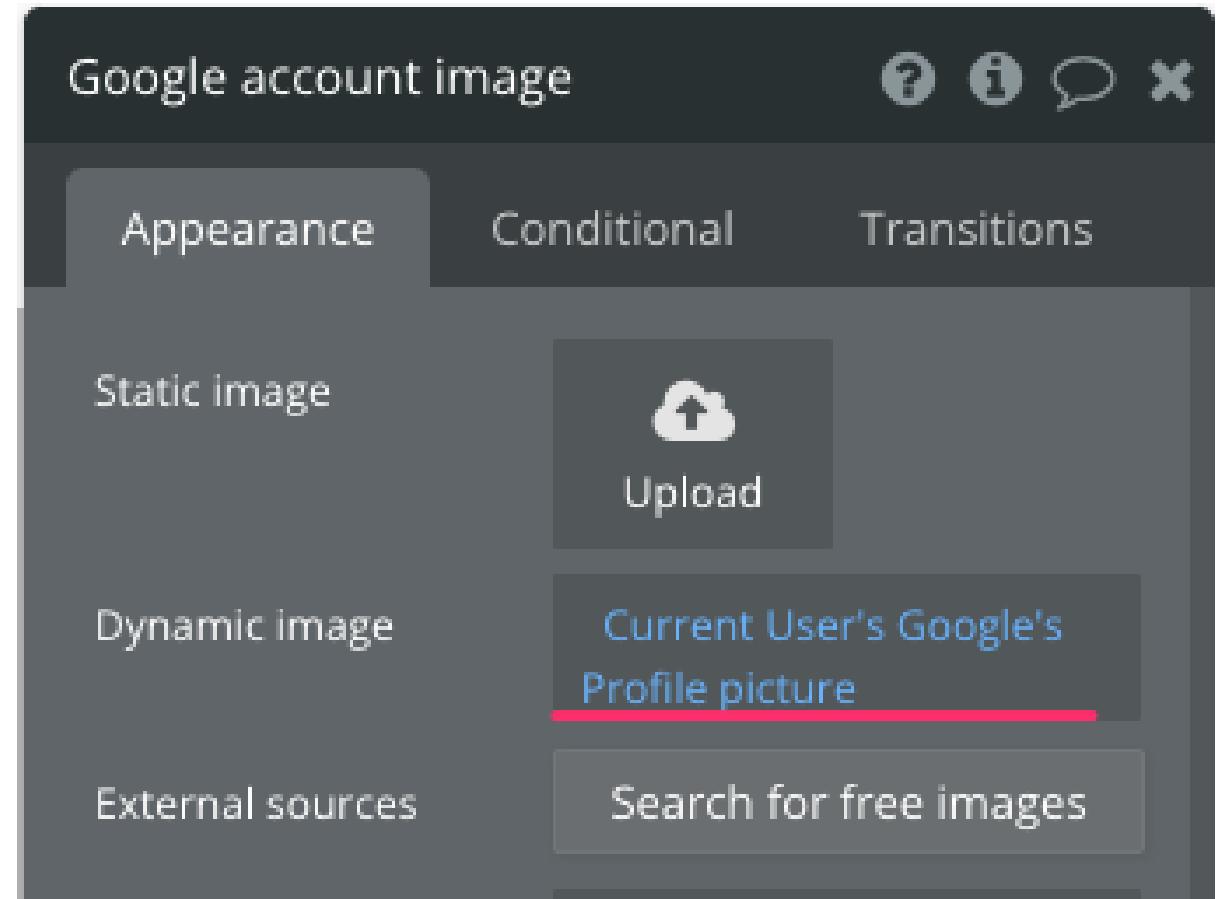
Button Log out (Hea...)

- Visual elements から Image を選択し、"Pet Register" リンクと "Log in" ボタンの間に画像を配置します
 - 縦横 50 x 50 の正方形にしておきましょう



- そして、画像の参照元を設定します
- Bubbler の皆さんなら設定方法はイメージつきますよね
- 動的な表示設定は Dynamic image
- 表示したいのは現在ログインしているユーザの Google アカウントのプロフィール画像

- こんな感じですね



- ではプレビューしてみましょう！
 - 事前に Data の User にこれから動作確認する際に使用する Google メールアドレスと同じメールアドレスのデータがある場合は削除しておいてください
- ログインボタンを押すと Google のログイン画面に遷移し、そこでログインをすると Bubble 側に戻ってきましたよね？
- ちなみに、ログインすると Bubble 側の User data にログインした Google アカウントのメールアドレス情報が登録されます
 - もちろんパスワードは保存されていないのでご安心を 😊



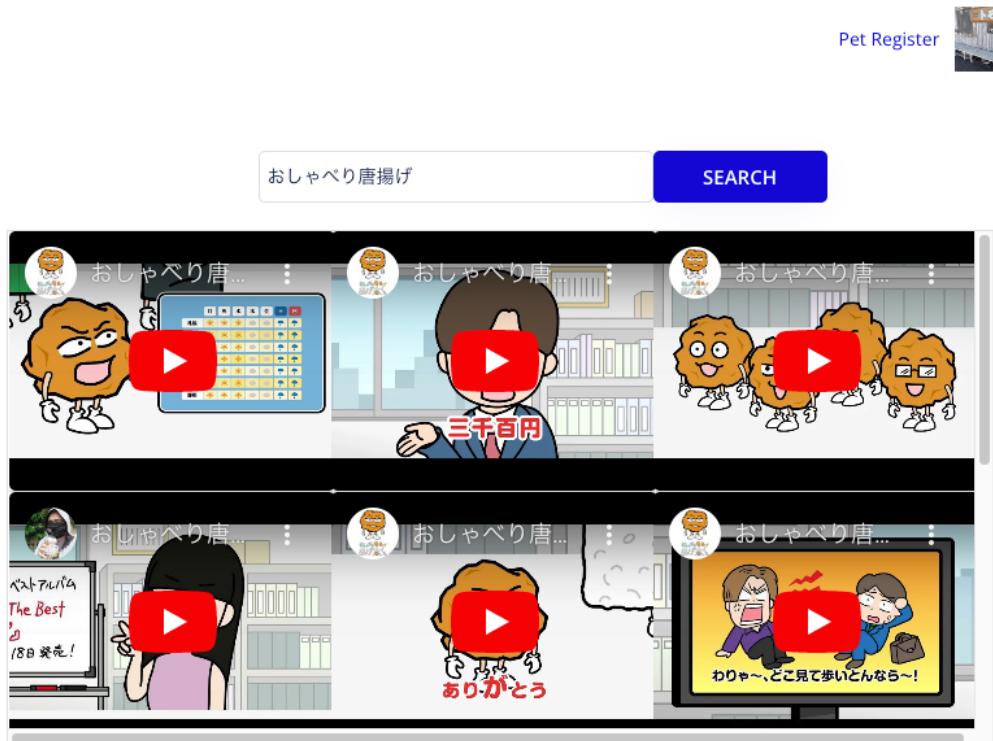
<Advanced>

最後に YouTube API を使った検索機能を組み込んでみましょう

.bubble

Pet Register

Log out



<Advanced>

- 現状では YouTube API の認証キーと検索キーワードは固定になっています
- なので、これを下記のように変えてみましょう
 - 認証キーはログインしているユーザ情報から取得する（新たに field を追加する）
 - 検索キーワードは画面から入力・検索できるようにする

<Advanced>

次のページにヒントを書いておきますのでまずはトライしてみましょう！

<Advanced>

- まずは User に対して `key` という field を追加し、事前に設定しておく
 - 本来ここもユーザ自身に入力させるなどの導線が自然ですが今回は時間の都合上、API キーの設定画面は割愛します
- API のパラメータを処理の中で動的に設定する場合、まずは固定値を無くし `Private` のチェックを外すことで、DataSource などで API 呼び出しする際にパラメータを指定できます
- `video_list` ページに新たに検索用の要素（テキストボックスと検索ボタン）を配置し、検索ボタンがクリックされた時に、YouTube API 経由で検索を行い、結果を Repeating Group にセットしてあげます



<Advanced>

- 重要なポイントを解説します
- まずは User の型に対して `key` という field を追加します

Fields for type User

Type name	User	
key	text	default
email	text	Built-in field
Modified Date	date	Built-in field
Created Date	date	Built-in field
Slug	text	Built-in field

[Create a new field](#)

<Advanced>

- そして、今回取得した認証キー（API キー）を登録済みの User に事前にセットしておきます

The screenshot shows a user interface for managing application data. At the top, there are tabs: Data types, Privacy, App data (which is selected), Option sets, and File manager. Below the tabs, the title is "Database views" and the subtitle is "Application data - All Users - Development version". There are buttons for New view, Primary fields, Search, and New entry. A search bar contains the placeholder "Search view names or data types...". The main area displays a single user entry in a table:

		Email	key
<input type="checkbox"/>	<input type="checkbox"/>	kyougoku2bubble@gmail.com	AlzaSy [REDACTED]

Below the table, there are buttons for All Users and a edit icon.

<Advanced>

- 次に YouTube API 側の設定はこんな感じですね

The screenshot shows a configuration interface for a workflow action. At the top, there are fields for 'Name' (set to 'search'), 'Use as' (set to 'Data'), 'Data type' (set to 'JSON'), and a URL field containing 'https://www.googleapis.com/youtube/v3/search'. Below this, there are sections for 'Headers' (with an 'Add header' button) and 'Parameters'. The 'Parameters' section contains four entries:

Key	Value	Private	Allow blank	Optional
key	(redacted)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
q	(redacted)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
maxResults	10	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
type	video	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Below the parameters are two checkboxes: 'Include errors in response and allow workflow actions to continue' and 'Capture response headers'. A message at the bottom states 'This call has been modified since you last initialized; consider reinitializing.' At the very bottom are buttons for 'Reinitialize call' and 'Manually enter API response'.

<Advanced>

- ポイントとしては、動的に指定したい項目については Value の値を削除し、Private のチェックを外しました
- これにより、API を使う場面でこれらのパラメータの値を指定することができるようになります

The screenshot shows a configuration interface for an API call. At the top, there are fields for 'Name' (set to 'search'), 'Use as' (set to 'Data'), 'Data type' (set to 'JSON'), and a URL field containing 'https://www.googleapis.com/youtube/v3/search'. Below this, under 'Headers', there is a button labeled 'Add header'. Under 'Parameters', there are four entries: 'key' (Value field is redacted), 'q' (Value field is redacted), 'maxResults' (Value: 10, Private checked, Allow blank unchecked, Optional unchecked), and 'type' (Value: video, Private checked, Allow blank unchecked, Optional unchecked). There are also buttons for 'Add parameter', 'Include errors in response and allow workflow actions to continue' (unchecked), and 'Capture response headers' (unchecked). A message at the bottom states 'This call has been modified since you last initialized; consider reinitializing.' with buttons for 'Reinitialize call' and 'Manually enter API response'.

<Advanced>

- ついでに、検索のオプションを2つ追加しておきましょう
- key: maxResults、1回の検索で取得するデータの件数
 - デフォルトは5
- key: type、検索対象のデータ種類
 - video を指定することで、動画だけの検索が可能

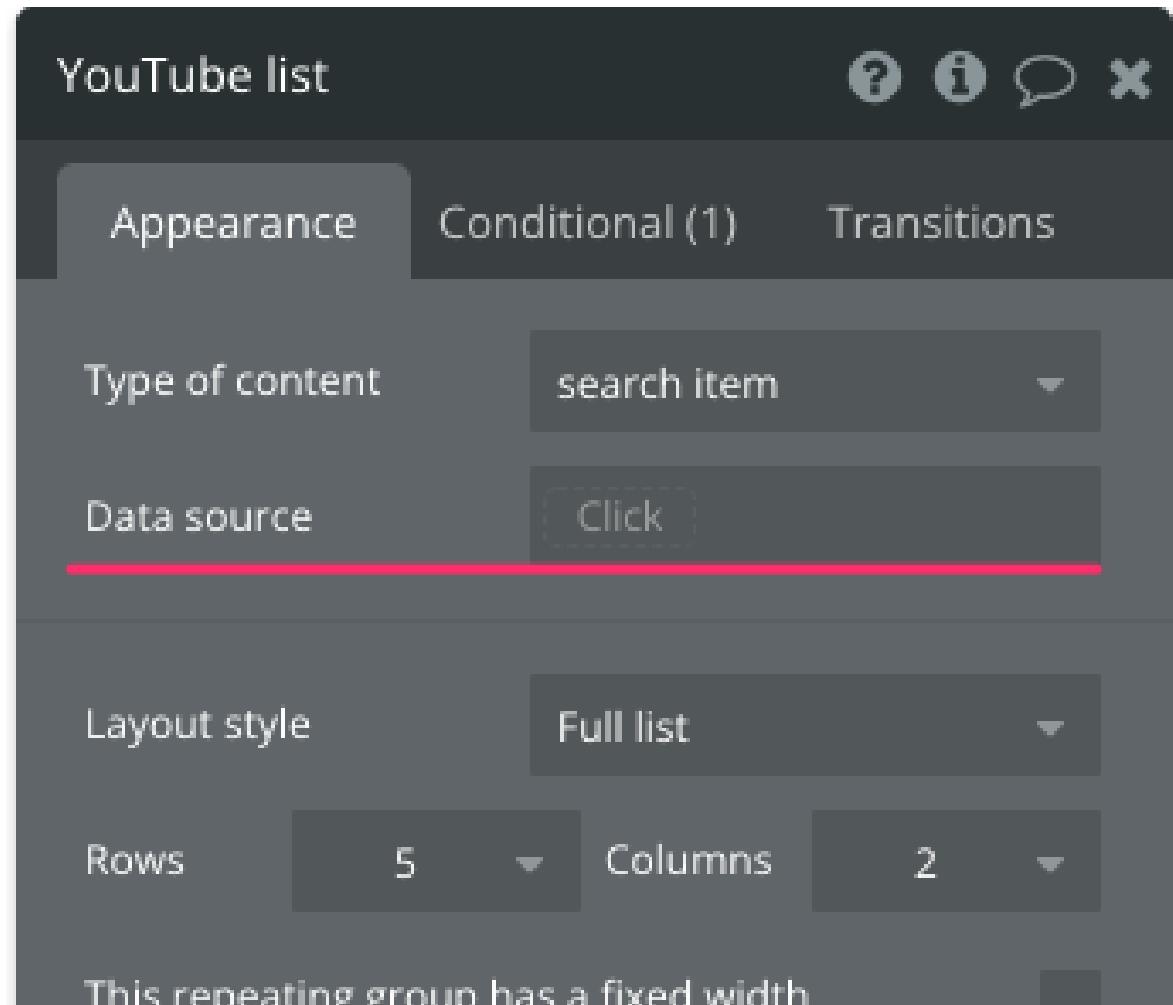
The screenshot shows a configuration interface for an API call named 'search'. The 'Data type' is set to 'JSON'. The method is 'GET' and the URL is 'https://www.googleapis.com/youtube/v3/search'. The 'Parameters' section contains four entries:

- Key: key, Value: (empty), Private: Allow blank: Optional:
- Key: q, Value: (empty), Private: Allow blank: Optional:
- Key: maxResults, Value: 10, Private: Allow blank: Optional:
- Key: type, Value: video, Private: Allow blank: Optional:

Below the parameters are two checkboxes: 'Include errors in response and allow workflow actions to continue' and 'Capture response headers'. A message at the bottom states 'This call has been modified since you last initialized; consider reinitializing.' At the bottom are buttons for 'Reinitialize call' and 'Manually enter API response'.

<Advanced>

- 続いて Repeating Group の Data source を空にします
- 表示する対象データの指定は、新たに設ける検索機能のワークフローの中で設定するため、ここが空でも問題ありません



<Advanced>

- 次にキーワード検索が行われた時のワークフローの設定となるため、新たに用意した検索ボタンに対してワークフローを設定します
- 今回は要素 (Repeating Group) に対するアクションとなるため "Element Actions" の中にある "Display list" を選択します

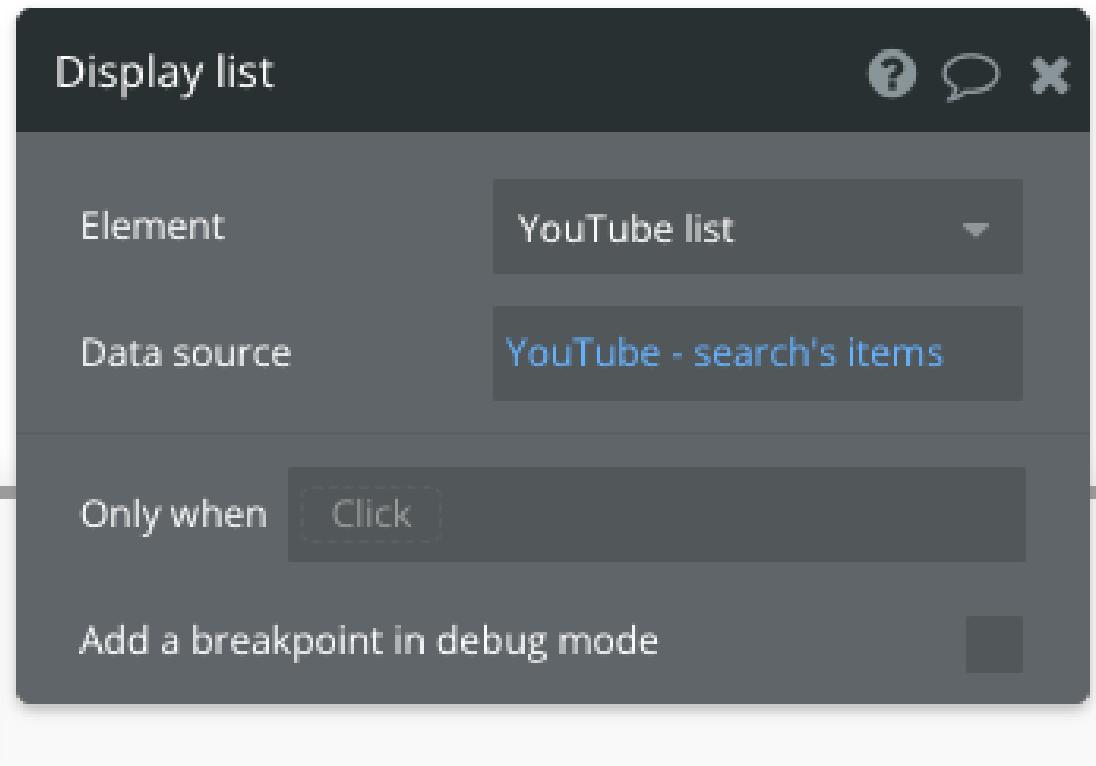
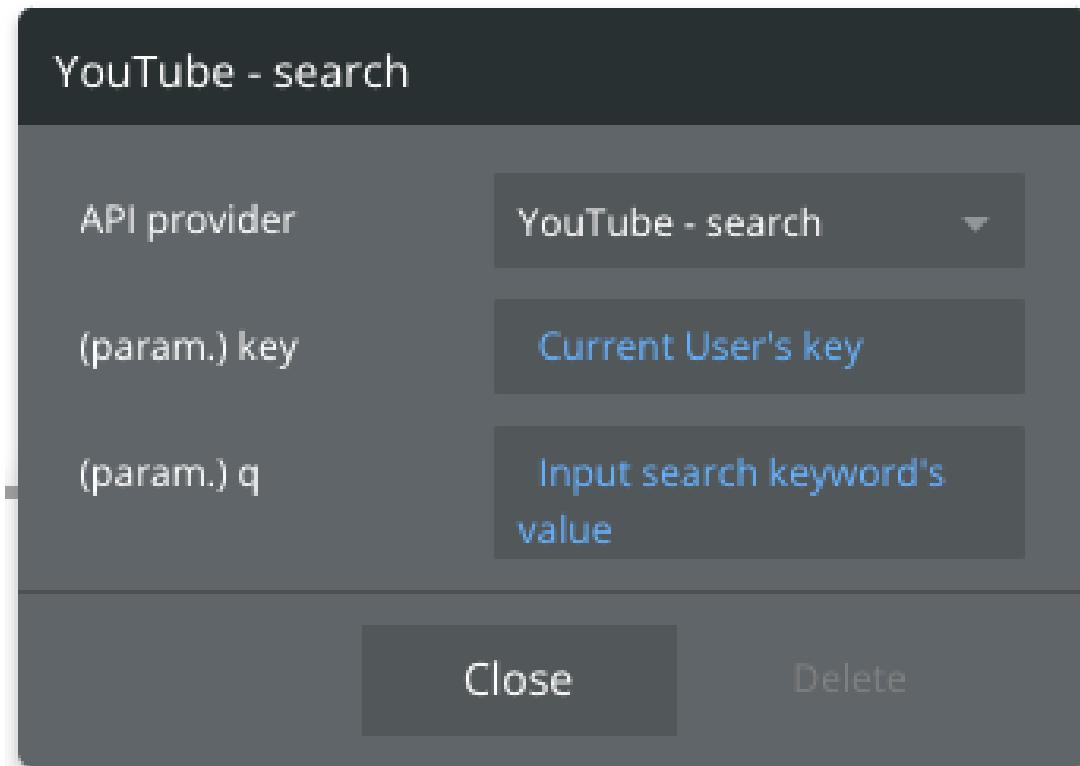
Search for an action...		
 Account	All elements	
 Navigation	Show	Hide
 Data (Things)	Toggle	Animate
 Email	Scroll to	Set state
 Payment	<i>Input</i>	
 Analytics	Set focus	Reset inputs
 Element Actions	<i>Group</i>	
 Plugins	Display data	Reset data
 Custom Events	<i>Repeating Group</i>	
	<u>Display list</u>	Show previous
	Show next	Clear list
	Go to page	Scroll to entry

<Advanced>

- Data source には Get data from an external API を選択し、API 経由のデータ表示とします
- そして API provider に YouTube – search を選択すると、先ほどはなかった 2 つの param が表示されるはずです
- これは先ほど YouTube API 側の設定で動的に指定したいパラメータについて Private のチェックを外したためとなります
- ここで指定する値はいつもの Dynamic data ですね

<Advanced>

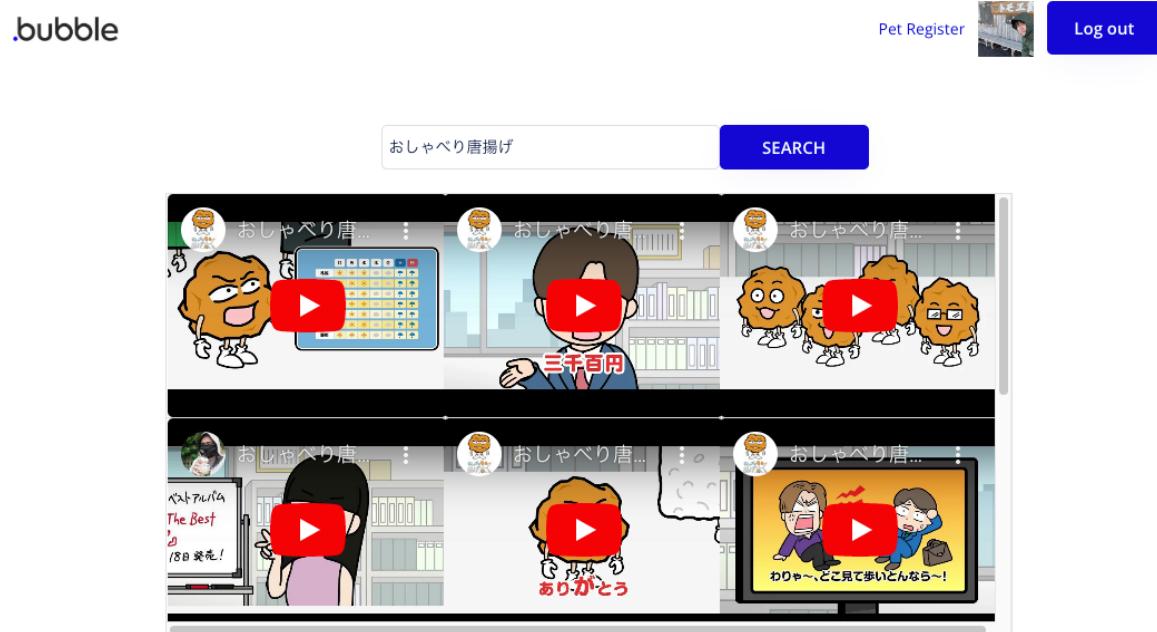
- 設定するとこんな感じです



<Advanced>

プレビューしてみましょう

- ・ログインしている状態で、キーワード欄に入力した動画一覧が表示されましたか？



<Advanced>

- いかがでしたでしょうか？
- API 連携を行うことで、Bubble で出来る幅がさらに広がったと思います！
- 合宿でも外部の API を使うチームがあるかもしれません、そのときは今日学んだことを活かして、API 連携を実践してみましょう！

Bubble での共同作業の Tips

チームで Bubble を使って共同作業をする時に役立つ情報をお伝えします。

Bubbleでのチームメンバーの招待方法

チームメンバーと一つのアプリを共同編集するには...

- チームで一つのアカウントを用意し、そのアカウントをみんなでログインして使いましょうことになります

- Bubble には正規の方法による共同編集機能はついているのですが... 💰💰💰

RECOMMENDED				
Free	Starter	Growth	Team	Enterprise
Best for learning how to use Bubble	Best for launching your app and testing	Best for growing your user base	Best for scaling your team and business	Best for maximum security and scale
\$0 / month	\$32 / month	\$134 / month	\$399 / month	Contact us
Try Free	Get started	Get started	Get started	Talk to sales
Great for: Projects that are under construction	Great for: MVPs and simple tools with small to moderate user bases	Great for: Consumer projects with complex functionality	Great for: Scaling projects with high usage	Great for: Internal tools and customer-facing apps
Free plan features:	Everything in Free, plus:	Everything in Starter, plus:	Everything in Growth, plus:	Everything in Team, plus:
<ul style="list-style-type: none"> ✓ Development version ⓘ ✓ API connector ⓘ ✓ Component library ⓘ ✓ 1 app editor ⓘ ✓ 50k workload units/mo ⓘ ✓ 6 hours of server logs ⓘ 	<ul style="list-style-type: none"> ✓ Live app ⓘ ✓ Custom domain ⓘ ✓ Recurring workflows ⓘ ✓ Basic version control ⓘ ✓ 175k workload units/mo ⓘ ✓ 2 days of server logs ⓘ 	<ul style="list-style-type: none"> ✓ 2 app editors ⓘ ✓ Premium version control ⓘ ✓ Two-factor authentication ⓘ ✓ 10 custom branches ⓘ ✓ 250k workload units/mo ⓘ ✓ 14 days of server logs ⓘ 	<ul style="list-style-type: none"> ✓ 5 app editors ⓘ ✓ Sub apps ⓘ ✓ 25 custom branches ⓘ ✓ 500k workload units/mo ⓘ ✓ 20 days of server logs ⓘ 	<ul style="list-style-type: none"> ✓ Choice of hosting location ⓘ ✓ Centralized admin ⓘ ✓ Dedicated server ⓘ ✓ Priority support ⓘ ✓ Enhanced security ⓘ ✓ Custom workload units ⓘ

Bubbleでの同時編集の注意点

- 同じエレメントに対する同時編集は避けましょう(後から行われた編集で上書きされます)
 - そのためには、画面ごとに担当者を決めて同じ画面の同時編集を避けるのがおすすめです。
- それ以外の注意点は特にありません。
- 編集は他の人の画面にリアルタイムで反映されます。
 - 画面の編集だけでなく、データベースやWorkflowの編集もリアルタイムで反映されます。
- 画面ごとに作業分担すれば、難なく開発できそうです。

チーム開発演習

- DevelopmentPhase のチームで、新たに一つアプリを開発してみてください
 - DevelopmentPhase で開発予定のアプリとは異なるアプリにしましょう
- AdaloかBubble のどちらを使うかは自由です
 - DevelopmentPhase で使う予定の方を選んでも良いですし、両方を選んでも良いです
- 全員が手を動かすようにしましょう

作業分担のやり方の例

- 画面や機能の洗い出しとデータベース設計は最初にみんなで一緒にやりましょう
- 画面ごとに担当者を決めましょう(登録画面担当、一覧画面担当、詳細画面担当、更新画面担当など)
- 複数人で一つのPCを使って作業をするというやり方もあります。その場合、PC操作をする人と指示をする人は定期的に交代しましょう



演習結果の発表

チームごとに演習で作ったアプリについて発表してみましょう！

まとめ

- Learning Phase のレクチャーは以上になります
 - Adalo や Bubble をマスターできましたか？
 - チーム開発のやり方のイメージがつきましたか？
- 次回からは Development Phase として、各チームがプレゼンをしたアイデアのブラッシュアップや開発をしていきます
 - それに向けて、自分たちのアイデアの再確認や、開発に使用するツール(Adalo? Bubble? それ以外?)の選定をしておきましょう

最後に

- Google Cloud Platform で発行した API キーや OAuth の情報を削除しておきましょう

以上です！

お疲れさまでした！