

Programming Boot Camp Learning Phase #3

Bubble Advance

2023/11/11

table of contents

- External collaboration with Bubble
- Tips for collaboration in Bubble
- Team development exercises
- summary

Confirm what to do

- First, learn how to integrate applications created with Bubble with external services.
- Next, we will introduce some tips and things to keep in mind when working together using one application.
- Finally, the Development Phase team will practice and present app development

External collaboration with Bubble

.bubble

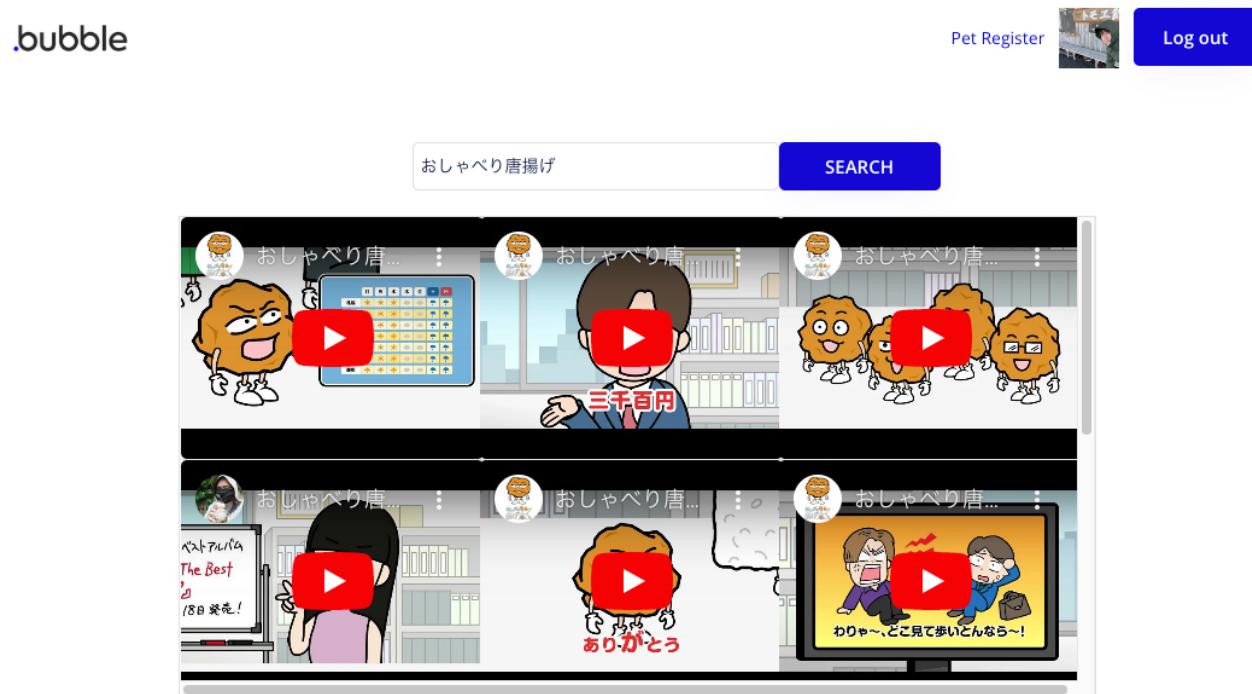
Pet Register

Log out



The screen includes the following functions:

- Display a list of YouTube videos
- Social login using your Google account
- Dynamically search YouTube video list

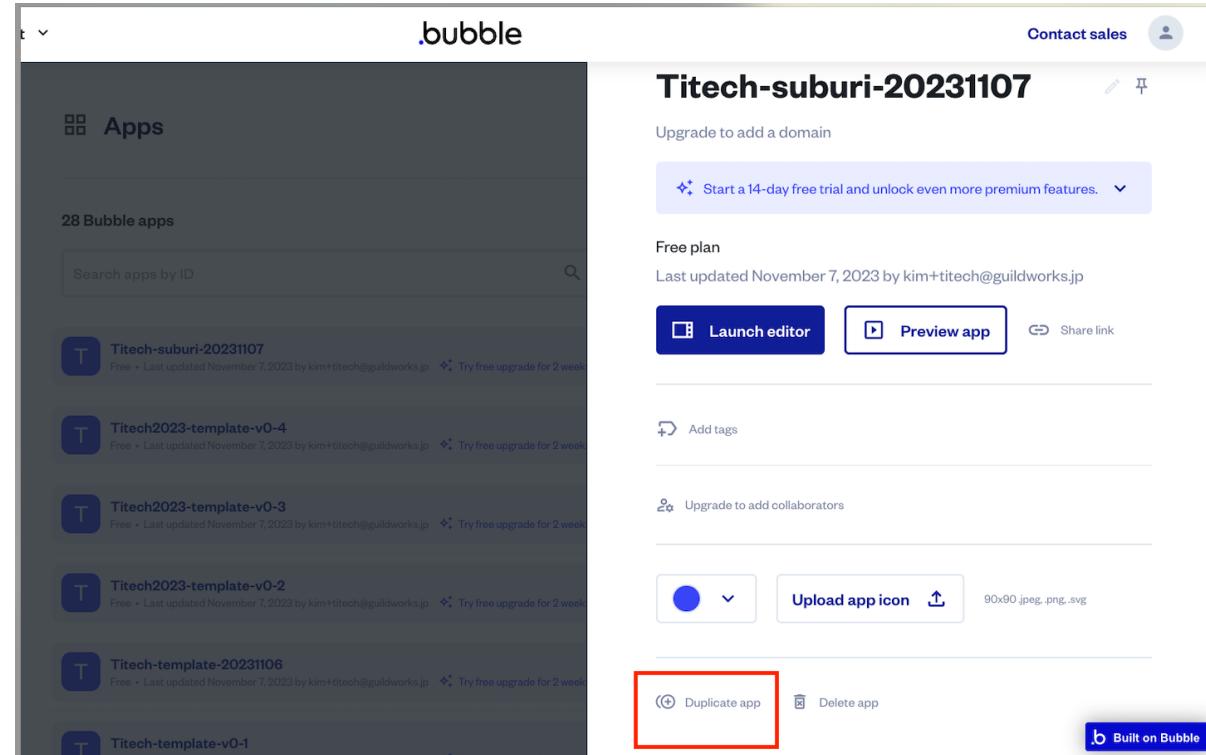


First, prepare a new application as the foundation

- Bubble I would like to copy and reuse the application used in the first lecture.
- Visit <https://bubble.io/home?tab=apps>

-A list of his Bubble apps created by each person's account will be displayed.

- From there, select the application you created in the basics section and click **Duplicate app** at the bottom of the popup



- An input dialog for the application name after copying will be displayed, so enter an appropriate name.
- Since the previous data will not be reused, leave the **Copy the application database content** check unchecked.
-Click COPY

[◀ 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

- When the copying is complete, it is OK if the editing screen (index) of the copied application is displayed.

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

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

Upgrade to deploy Preview

PetLog

Name

John Doe

Email

johndoe@gmail.com

Password

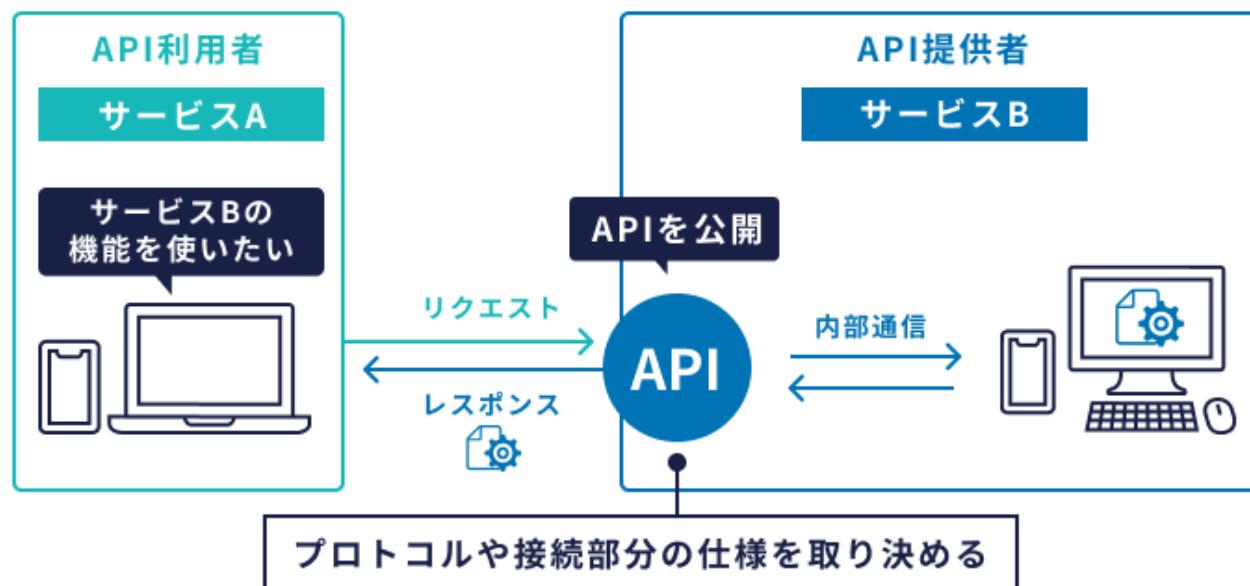
Re-enter Password

Role

Choose an role...

What is API?

API (Application Programming Interface) is a method for software and applications to communicate with other software and services. Through APIs, applications can share data and expose and leverage the functionality of other applications.



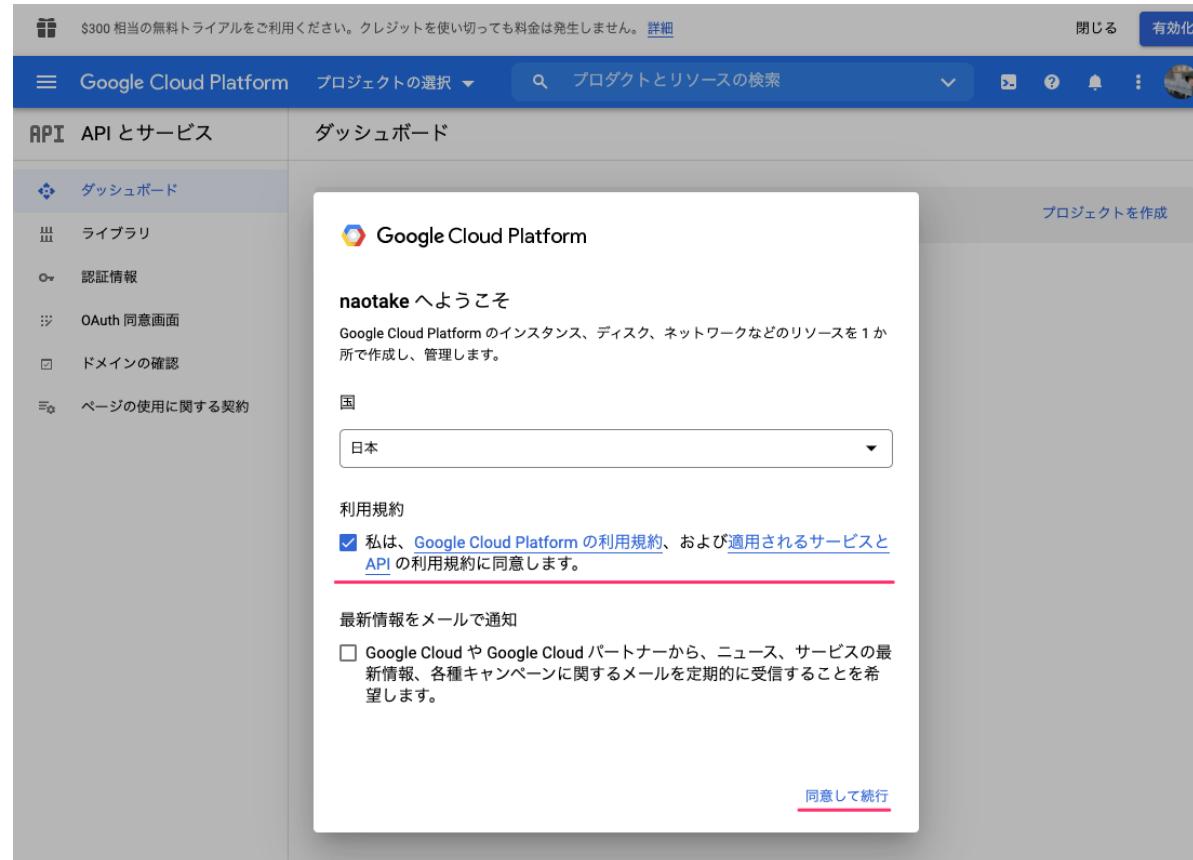
Preparation for YouTube linkage

- Next, make the necessary preparations to display the YouTube video list.
- In this lecture, we will use various APIs provided by the cloud service "Google Cloud Platform" provided by Google.
 - Think of it as Google's version of "Amazon Web Service" (AWS) provided by Amazon

- Access the URL below
 - <https://console.developers.google.com/>
- Log in with your Google account
 - If you do not have a Google account, please create one.



- When you access the Google Cloud Platform (GCP) screen for the first time after logging in, a dialog like this will be displayed, so select your country, check the terms of service, and click "Agree and continue"



- Once you agree, the API and service dashboard will be displayed, so click the Create project link to create the project (box) for this lecture.

The screenshot shows the Google Cloud Platform API and service dashboard. At the top, there is a banner with a gift icon and the text '\$300 相当の無料トライアルをご利用ください。クレジットを使い切っても料金は発生しません。' with a '詳細' link. To the right are '閉じる' and '有効化' buttons. The main header includes the Google Cloud Platform logo, a 'プロジェクトの選択' dropdown, a search bar with 'プロダクトとリソースの検索', and various navigation icons. On the left, a sidebar titled 'API API とサービス' lists options: 'ダッシュボード' (selected), 'ライブラリ', '認証情報', 'OAuth 同意画面', 'ドメインの確認', and 'ページの使用に関する契約'. The main content area is titled 'ダッシュボード' and displays 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.' with a red underline under the 'プロジェクトを作成' button.

- A new project creation screen will be displayed, so specify an appropriate name for the project name and create it.

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 * ?

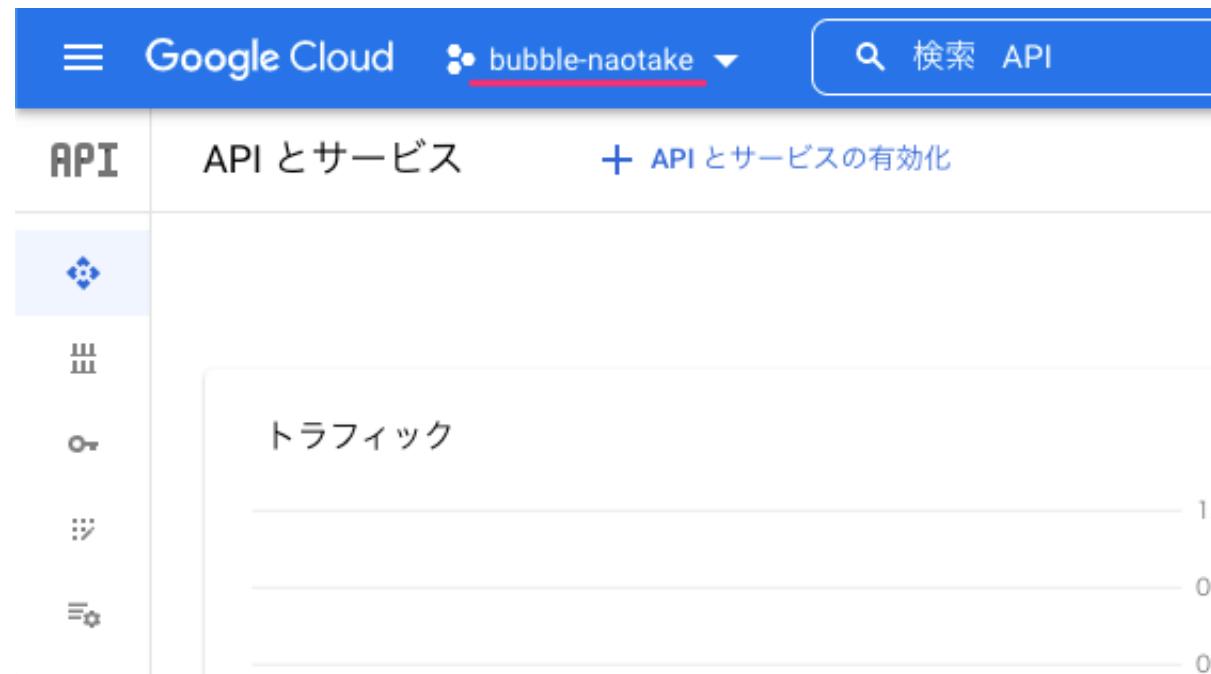
プロジェクト ID: **bubble-advance**。後で変更することはできません。 [編集](#)

場所 * 参照

親組織またはフォルダ

[作成](#) [キャンセル](#)

- Once creation is complete, the dashboard of the created project will be displayed
- It is OK if the created project name is displayed to the left of Google Cloud at the top of the screen.



- Next, we will issue the authentication key necessary to display the YouTube video list.
- Type "API" in the search box at the top of the screen to open the API and Services menu.
- Select "APIs and Services" in the "Projects and Pages" block from the suggestions



- When the API and services page opens, select "Credentials" from the left menu
- The right panel will change to authentication information, so click "+Create authentication information" at the top of the screen
- Select "API Key" from the popup

The screenshot shows the Google Cloud Platform's API & Services page. On the left, there's a sidebar with the following options:

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

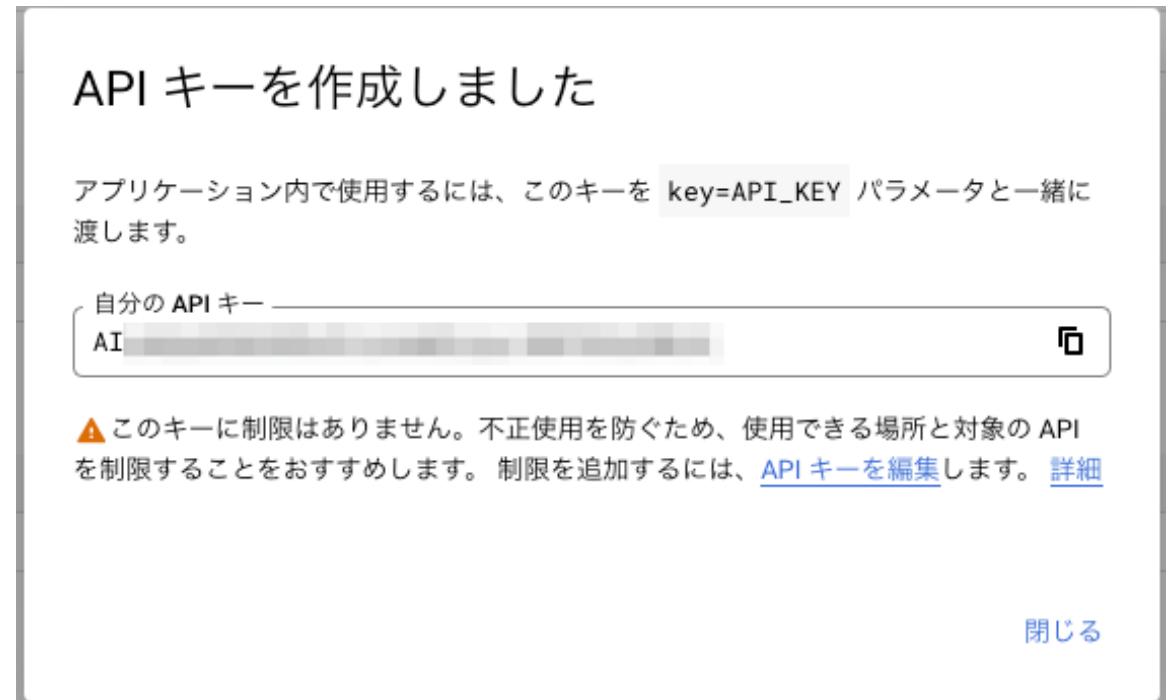
The main area is titled "認証情報" (Authentication Information). At the top, there are two buttons: "**+ 認証情報を作成**" (Create Authentication Information) and "削除" (Delete). A modal window is open over the main content, with its title bar also having the "**+ 認証情報を作成**" button.

The modal window contains the following sections:

- API キー** (API Key)
 - 有効な API にアクセス
 - 必ず、API キーを生成** (Warning: Always generate an API key)
- OAuth クライアント ID** (OAuth Client 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
- サービス アカウント** (Service Account)
 - 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
- ウィザードで選択** (Select via Wizard)
 - 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

Below the modal, there's a section titled "OAuth 2.0 クライアント ID".

- A pop-up will appear informing you that API key creation is complete, so make a note of the value.
- You will be asked to restrict keys, but we will delete these keys at the end of the lecture, so let's continue as is.



- Now that we have issued the key to get the YouTube list from Bubble, let's configure the settings on the Bubble side.
- First, prepare to retrieve the YouTube list from Bubble via API

Preparation for Bubble API integration

- Select Plugins from the left menu and click the Add plugins button
- Search for "API" and install the plugin called API Connector

The screenshot shows the 'Install New Plugins' interface. On the left, there are filters for 'API' selected under 'Types' (Action, API, Background Services, Element, Event, Login Service, Storage) and 'Categories' (Analytics, Blog, Compliance, Containers, Customer Support, Data (things), Ecommerce, Email, Energy, Health & Fitness, Input Forms, Internationalization, Location, Media, Mobile). The main area lists four plugins:

- API Connector** by Bubble (289,709 apps) - highlighted with a red border and an 'Install' button.
- Zapier (Legacy)** by Zapier (12,498 apps) - an 'Install' button is visible.
- Discord API (OUTDATED)** by Tristan Forum (10,372 apps) - labeled 'Free' and has an 'Install' button.
- Widgio API Connector** by twobrothers (6,170 apps) - labeled 'Free' and has an 'Install' button.

A 'DONE' button is at the bottom right, and a note about recurring subscriptions is at the bottom center.

- This plugin is a plugin that allows you to connect Bubble to information on the other side of the API using APIs published in the world from the Bubble app.
- Some Bubble plugins are specialized for specific APIs and may be simpler to configure than this API Connector.
- This time, let's use the YouTube API via the most basic API Connector plugin to display a list of videos.

- Now let's configure the settings to use the YouTube API.
- Select "API Connector" from Installed Plugins on the Plugins tab
- There is an overview of this plugin on the right panel, click "Add another API" below to start setting up a new API integration

The screenshot shows the 'API Connector' settings page. At the top left is a blue logo consisting of a lowercase 'b'. To its right, the text 'API Connector' is displayed. In the top right corner, there is a small grey box containing the text 'reference →' with a thin grey arrow pointing to the right. Below the logo, a paragraph of text reads: 'The API lets you define your own API calls directly in the Bubble Editor and use them in your app. A good practice is to get the calls to work in Postman first, and then add them here. You can also convert your connections into plugins that can be used in more than one app, and even shared with other Bubblers! See [your personal plugins page](#) for more details.' At the bottom left, there is a blue rectangular button with a white border and the text 'Add another API' inside. At the bottom right, there is another blue rectangular button with a white border and the text 'Uninstall this plugin' inside.

- When you click "Add another API", you will see API settings parts like this
- We will provide a brief explanation and settings for each item.

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: Group name for this API integration

-This time YouTube

The screenshot shows the configuration interface for a new API integration named "YouTube". The interface is divided into several sections:

- API Name:** A field containing "YouTube".
- Authentication:** A dropdown menu set to "None or self-handled".
- Shared headers for all calls:** A section with a button labeled "Add a shared header".
- Shared parameters for all calls:** A section with a button labeled "Add a shared parameter".
- API Call:** A collapsed section with a "Name" field containing "API Call".
- Import another call from cURL:** A button labeled "Add another call".
- Buttons:** "collapse" and "expand" buttons for the collapsed sections.
- Expand all calls:** A button labeled "expand all calls".

2. Authentication: Common authentication method performed by APIs under this group
- This time "None or self-handled"

The screenshot shows the configuration interface for an API named "YouTube".

- 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

Below the main configuration area, there is a section for importing API calls:

- Name:** API Call
- Import another call from cURL:** Add another call
- expand** (button)
- expand all calls** (button)

3. Shared headers for all calls: Common header information specified for all APIs under this group.

*Not specified this time

The screenshot shows the configuration interface for the YouTube API. At the top, the API Name is set to "YouTube" and the Authentication type is "None or self-handled". A "collapse" button is in the top right corner. Below this, there are two sections: "Shared headers for all calls" and "Shared parameters for all calls". Each section has an "Add a shared header" or "Add a shared parameter" button respectively. The "Shared headers for all calls" section is currently expanded, showing a table with one row where "Name" is "API Call" and "Value" is also "API Call". An "expand" button is in the top right corner of this table. At the bottom left, there is a link to "Import another call from cURL" and a blue "Add another call" button. On the right side, there is an "expand all calls" button.

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

4. Shared parameters for all calls: Common parameter information specified for all APIs under this group

*Not specified this time

The screenshot shows the configuration interface for an API named "YouTube". The top bar includes fields for "API Name" (set to "YouTube"), "Authentication" (set to "None or self-handled"), and a "collapse" button. Below this, there are two sections: "Shared headers for all calls" and "Shared parameters for all calls". Each section has an "Add a shared header" or "Add a shared parameter" button. The "Shared parameters for all calls" section is expanded, showing a table with columns for "Name" and "API Call". A "collapse" button is located in the top right corner of this expanded section. At the bottom, there is a link to "Import another call from cURL" and a blue "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

Name API Call expand

Import another call from cURL

Add another call

expand all calls

- Next, we will set the specific API contents for this group.
- On the previous screen, there is a link to the right of the block that says "API Call", click on it.

The screenshot shows a configuration interface for an API. At the top, there are fields for 'API Name' (set to 'YouTube'), 'Authentication' (set to 'None or self-handled'), and a 'collapse' button. Below these 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 central area is labeled 'expand' at the top right. At the bottom left, there is a field for 'Name' containing 'API Call', with an 'expand' button to its right. At the very bottom left is a button labeled 'Import another call from cURL' with a blue border, and at the bottom right is a button labeled 'expand all calls'.

API Name: YouTube

Authentication: None or self-handled

collapse

Shared headers for all calls

Add a shared header

Shared parameters for all calls

Add a shared parameter

Name: API Call

expand

Import another call from cURL

Add another call

expand all calls

- Then, the specific API setting items will be displayed, so I will briefly explain them.

collapse

Name API Call Use as Data Data type JSON (use [] for params)

GET 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

The screenshot shows the configuration panel for an 'API Call' action. At the top, there are fields for 'Name' (set to 'API Call'), 'Use as' (set to 'Data'), 'Data type' (set to 'JSON'), and a note '(use [] for params)'. Below these are dropdowns for 'Method' (set to 'GET') and 'Attempt to make the call from the browser' (unchecked). The 'Headers' section contains an 'Add header' button. The 'Parameters' section contains an 'Add parameter' button. There are two checkboxes at the bottom: 'Include errors in response and allow workflow actions to continue' (unchecked) and 'Capture response headers' (unchecked). A red warning message '⚠ You need to initialize this call before it will work.' is displayed. At the bottom, there are two buttons: 'Initialize call' (highlighted with a blue border) and 'Manually enter API response'.

1. Name: Specific API name

- This time enter "search"

The screenshot shows a configuration interface for an API call. The 'Name' field (1) contains 'search'. The 'Method' dropdown (2) is set to 'GET' and the URL is 'https://www.googleapis.com/youtube/v3/search'. The 'Data type' is set to 'JSON'. The 'Headers' section (3) has an 'Add header' button. The 'Parameters' section (4) contains two entries: 'key' with value 'AlzaSyAA3b34Zbc9rznw.' and 'q' with value 'おしゃべり唐揚げ'. Both parameters have the 'Private' checkbox checked. There are also 'Allow blank' and 'Optional' checkboxes. At the bottom, there are buttons for 'Initialize call' (highlighted in blue) and 'Manually enter API response'.

Name: search
Method: GET URL: https://www.googleapis.com/youtube/v3/search
Data type: JSON
Headers: Add header
Parameters:
Key: key Value: AlzaSyAA3b34Zbc9rznw. Private: Allow blank: Optional:
Key: q Value: おしゃべり唐揚げ Private: Allow blank: Optional:
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

2. Path: URL of specific API

- This time enter `https://www.googleapis.com/youtube/v3/search`

The screenshot shows a configuration interface for a workflow step. The top section is titled 'Path' with the value 'search'. Below it, 'Method' is set to 'GET' and the 'URL' is 'https://www.googleapis.com/youtube/v3/search'. A note '(use [] for params)' is present. The interface is divided into sections by numbered circles:

- ① Name: 'search'
- ② Method and URL: 'GET https://www.googleapis.com/youtube/v3/search'
- ③ Headers: 'Add header' button
- ④ Parameters:
 - Key: 'key', Value: 'AlzaSyAA3b34Zbc9rzxnw', 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

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 red warning message '⚠ You need to initialize this call before it will work.' is displayed. Two buttons at the bottom are 'Initialize call' (highlighted in blue) and 'Manually enter API response'.

3. Headers: Header information specific to this API

*Not specified this time

The screenshot shows a configuration interface for an API call named "search". The interface includes fields for Name, Method, URL, Headers, Parameters, and various workflow options.

Name: search (marked with ①)

Method: GET (marked with ②)

URL: https://www.googleapis.com/youtube/v3/search

Headers: (marked with ③) - An "Add header" button is present.

Parameters: (marked with ④) - Two parameters are defined: "key" with value "AlzaSyAA3b34Zbc9rznw" and "q" with value "おしゃべり唐揚げ". Both parameters have "Private" checked, "Allow blank" unchecked, and "Optional" unchecked.

Workflow Options:

- Include errors in response and allow workflow actions to continue:
- Capture response headers:

A red warning message at the bottom states: **⚠ You need to initialize this call before it will work.**

At the bottom, there are two buttons: **Initialize call** (highlighted in blue) and **Manually enter API response**.

4. Parameters: Parameter information specific to this API

- This time, specify the following two parameters. Private / Allow blank / Optional as attached
 - a. Key: `key` , Value: API key issued earlier on the Google account side
 - b. Key: `q` , Value: Keyword when searching for motion on YouTube (please specify freely)

- This is what it looks like when everything is set

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="Delete"/> | | | | |
| 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="Delete"/> |
| 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="Delete"/> |
| <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 | | | | | | |

- After completing the settings, check that the settings are correct.
- Click the "Initialize call" button at the bottom

- You probably got an error message with Status code 403.
- The content is as written, but it can be summarized as follows.

YouTube Data API v3 is disabled in your project.

Visit this URL to enable it and try again.

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": [
```

- It seems that the YouTube Data API used this time is not enabled on each person's girlfriend's Google account.
 - This is because from the beginning, Google has turned off features that everyone won't use, and only those who need them can turn them on themselves.
- So this time we want to use the YouTube Data API, so access the URL in the message and enable it.
[https://console.developers.google.com/apis/api/youtube.googleapis.com/overview?
project={Each person's project ID}](https://console.developers.google.com/apis/api/youtube.googleapis.com/overview?project={Each person's project ID})

- Then you should be kindly redirected to the "YouTube Data API v3" screen that you want to use this time.
- Click "Enable" to enable it as shown on the screen



- After a while, you should see a screen like this
- If the red line at the top of the screen says "Disable API", activation is successful.

The screenshot shows the Google Cloud Platform API library interface. On the left, there's a sidebar with navigation links: 'API & サービス' (selected), '有効な API とサービス' (selected), 'ライブラリ', '認証情報', 'OAuth 同意画面', and 'ページの使用に関する契約'. The main content area has a header 'API / サービスの詳細' with a red underline over the link 'API を無効にする'. Below the header, there's a large button with a play icon labeled 'YouTube Data API v3'. A description follows: 'The YouTube Data API v3 is an API that provides access to YouTube data, such as videos, playlists, and channels.' It's listed as owned by 'Google'. A table provides details: 'サービス名' (youtube.googleapis.com), 'タイプ' (公開 API), and 'ステータス' (有効). Buttons for 'LEARN MORE' and 'API EXPLORE' are shown, along with a circular arrow icon. At the bottom, tabs for '指標', '割り当て', and '認証情報' are visible, along with dropdown menus for 'グラフを選択' (4 個のグラフ) and '30 日'.

- Now that YouTube Data API has been enabled, return to the Bubble screen and click the "Initialize call" button.
- If successful, you should see a popup saying "Returned values - search"

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 | text |
| etag | bMKu0aiX56rekNwT5RVHZLOYr-o | text |
| nextPageToken | CAUQAA | text |
| regionCode | US | text |
| pageInfo totalResults | 1000000 | number |
| pageInfo resultsPerPage | 5 | number |
| items (list) | (see fields below) | search item |
| kind | youtube#searchResult | text |
| etag | WahfcvNgYipX-i3rjRIPUrHi0 | text |
| id kind | youtube#channel | text |
| id channelId | UCcRigzI_jBAZh_UySjv8xuw | text |
| id videoId | L2SaFPQI4_g | text |

SAVE Cancel

- This represents the information of the result (response) of executing the ["https://www.googleapis.com/youtube/v3/search"](https://www.googleapis.com/youtube/v3/search) API that we set this time.

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 | text |
| youtube#searchListResponse | |
| etag | text |
| mjrgGMgtIMWG6aVWolrNtjmgtG8 | |
| nextPageToken | text |
| CAUQAA | |
| regionCode | text |
| US | |
| pageInfo totalResults | number |
| 946629 | |
| pageInfo resultsPerPage | number |
| 5 | |
| items (list) | search item |
| (see fields below) | |
| kind | text |
| youtube#searchResult | |
| etag | text |
| WahficVNgYipX-l3rjRiPUrHi10 | |
| id kind | text |
| youtubee#channel | |
| id channelId | text |
| UCcRigzl_jBAZh_UySjv8xuw | |
| id videoId | text |
| S094aNshTSU | |

SAVE **Cancel**

I won't go into details, but there are three main points:

- The part marked `items(list)` is a list that stores the list of search results videos.
- The type is `search item`, which indicates that there are multiple types of data included in the search results.
 - The most important item in the settings that follows is `id videoId`

- As anyone who has watched YouTube will know, this Videoid is information that uniquely identifies each video, and this ID is also required when displaying on screen.
- Other items that are not particularly referenced may be confusing, so select `Ignore field` from the dropdown.
- By doing this, when handling it as Dynamic data, it will not be displayed in the options, so you will not have to worry about choosing a setting item.

- This is what it looks like when everything is set
- Once the settings are complete, click "SAVE" to save the settings.

Returned values - search

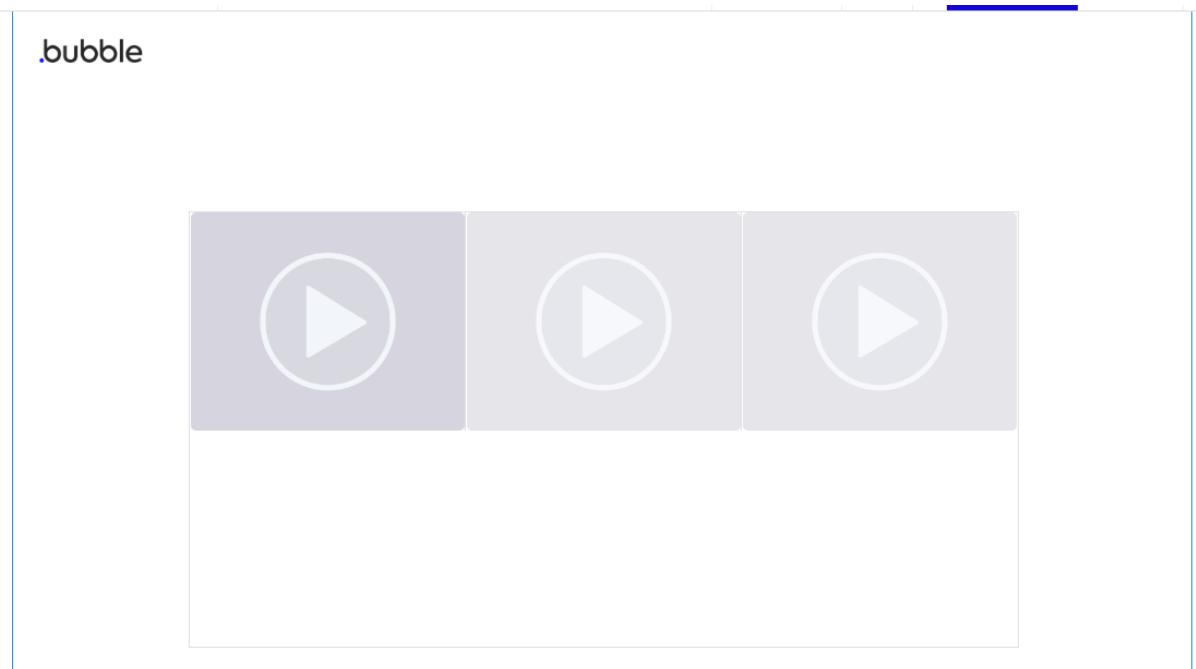
You can modify the data types that are returned by the call. This affects how you can use the data in Bubble. If you chose 'Ignore field', the fields won't be shown in the dropdowns.

| | |
|--|--------------|
| kind youtube#searchListResponse | Ignore field |
| etag mjrgGMgtIMWG6aVWolrNtjmgtG8 | 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 |

Preparing for screen display

- Now that we are ready to integrate Bubble's API, let's set up the screen.
- I think you have mastered the screen settings from the previous and this lecture, so please just give me an overview and try assembling the screen layout.

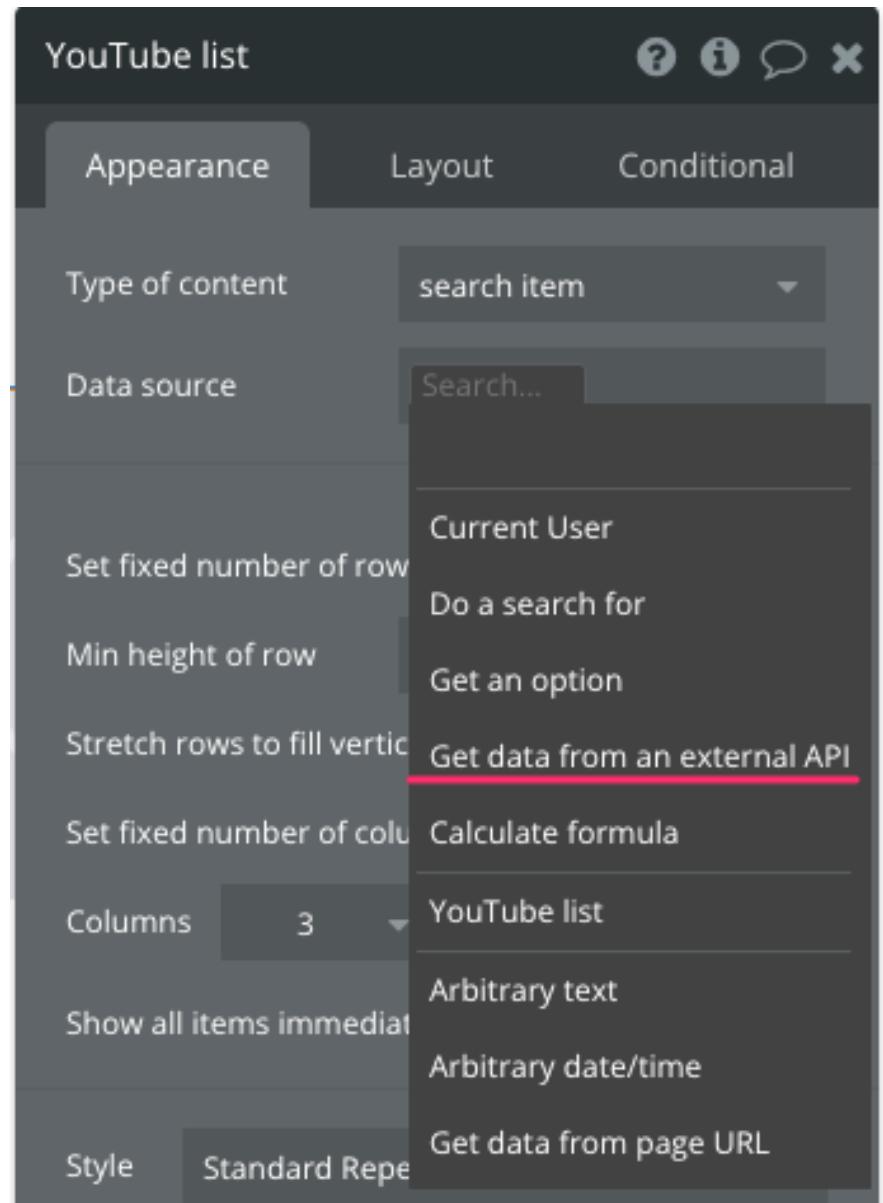
- Create a "video_list" screen by cloning the pet_list page
- Delete all contents of each cell of Repeating Group
- Instead, set the Visual elements Video to fill the cell.
- Something like this



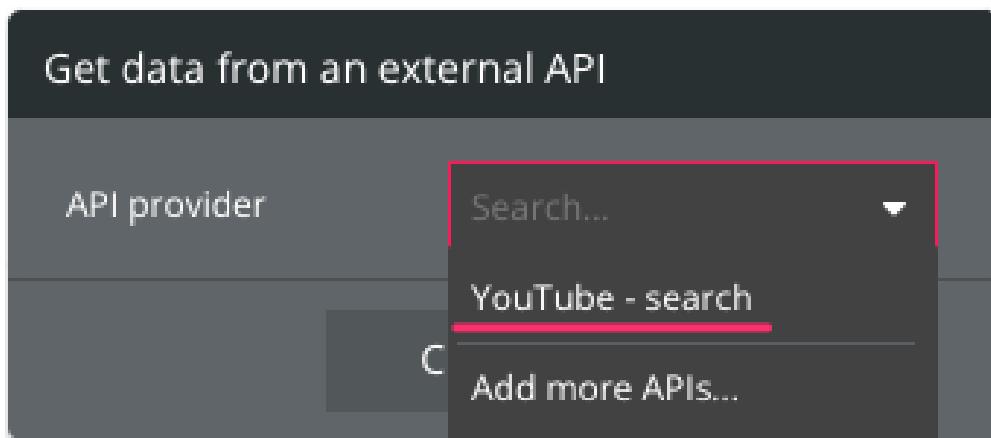
- Now that the layout is created, let's actually set up the YouTube list display.
- First, the Type of content of Repeating Group is the `search item` type that we checked earlier when executing the API.
- Now the type of data to repeat is for each YouTube video

The screenshot shows the configuration interface for a "YouTube list" component. At the top, there are four icons: a question mark, an information icon, a speech bubble, and a close button. Below the title "YouTube list" are three tabs: "Appearance" (selected), "Layout", and "Conditional". The "Layout" tab contains settings for "Type of content" (set to "search item"), "Data source" (set to "Click"), and "Set fixed number of rows" (unchecked). The "Min height of row" is set to 200 px. The "Stretch rows to fill vertical space" option is checked. The "Set fixed number of columns" option is also checked.

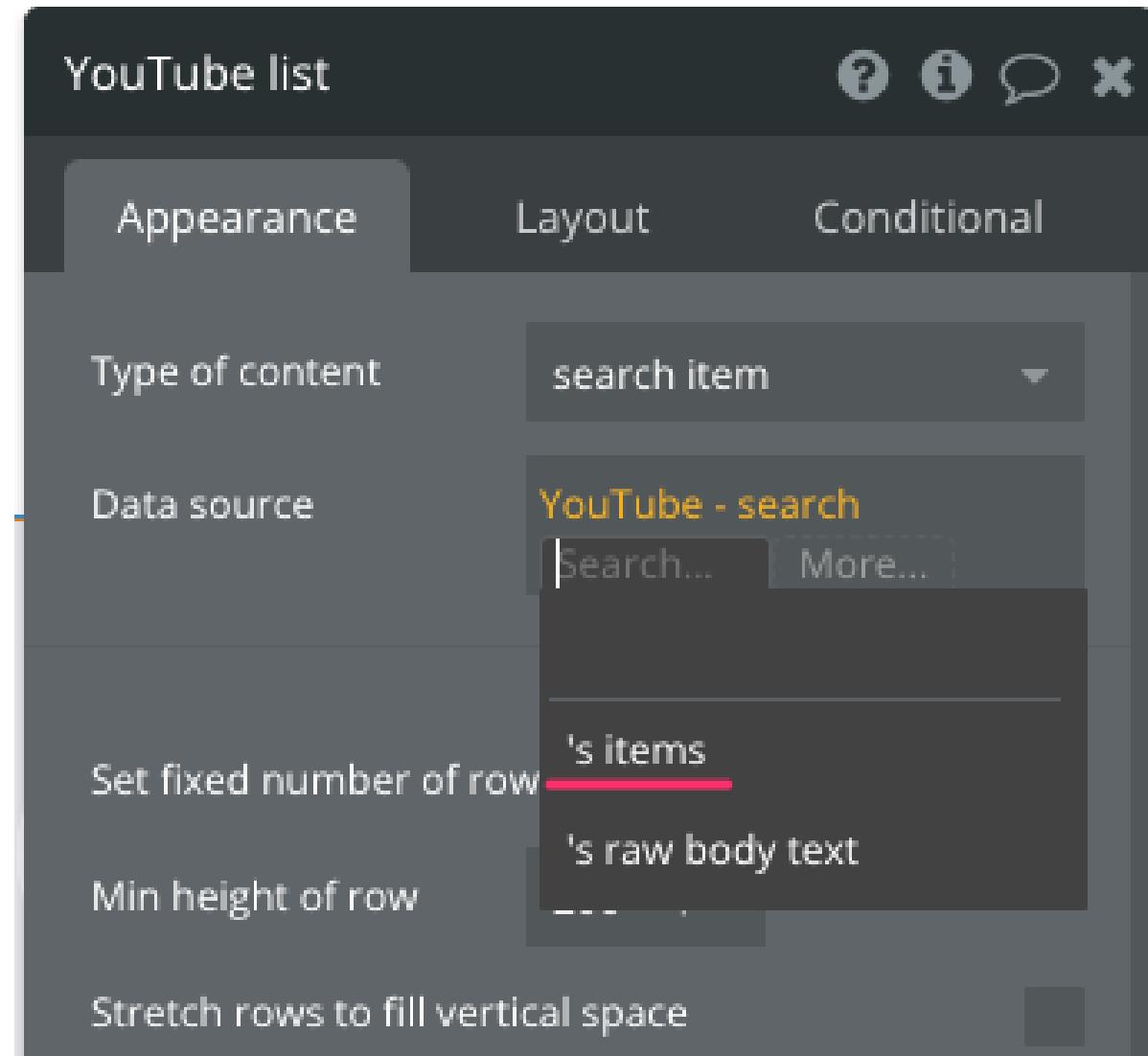
- Next is the data source, but this time we will target the YouTube video list obtained from the API.
- If you look at the options from Click, you will see an item called **Get data from an external API**, so select that.



- A popup will appear next to Get data from an external API, select the API provider dropdown.
- Then, there will be a candidate for `YouTube - search` that you set earlier in the dropdown, so please select it.
 - The name before the hyphen is the API group name, and the name after the hyphen is the specific API name.



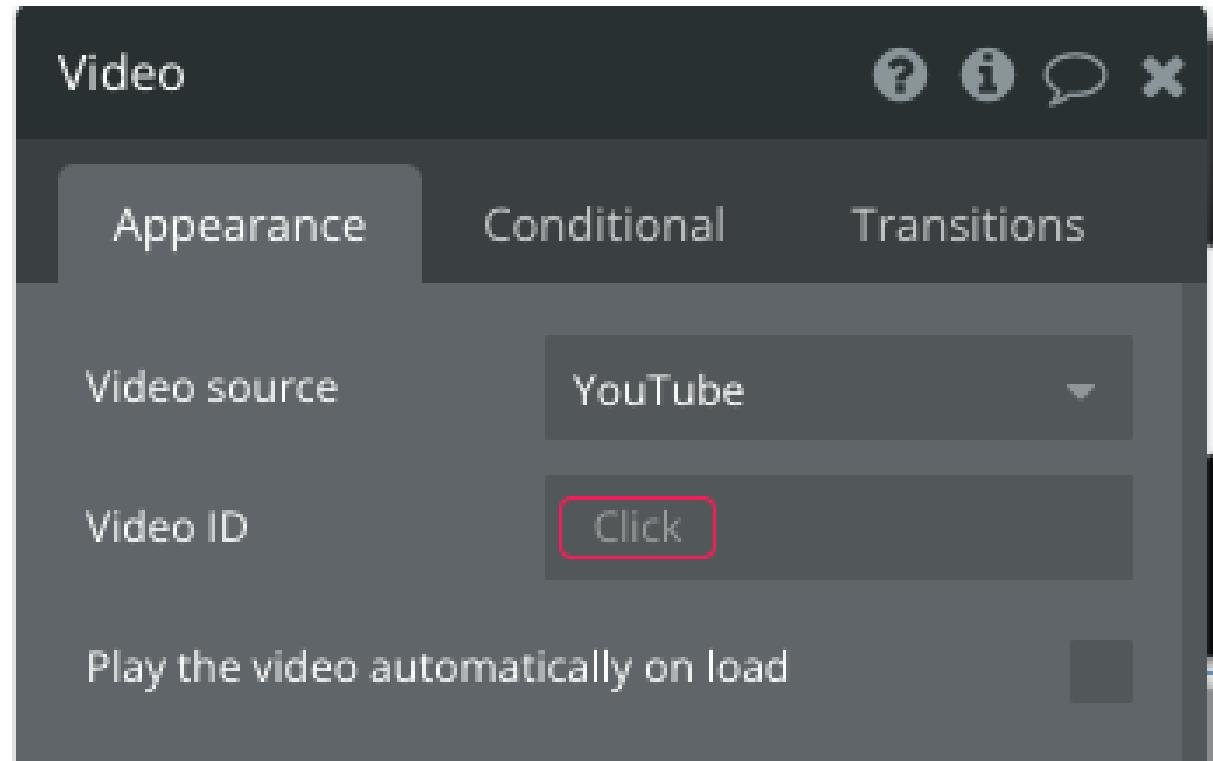
- Once the API Provider is configured, close it with CLOSE
- And further select 's items as the Data source selection of the original Repeating Group
- Now you can specify that the list of videos (items) included in the API results will be displayed repeatedly.



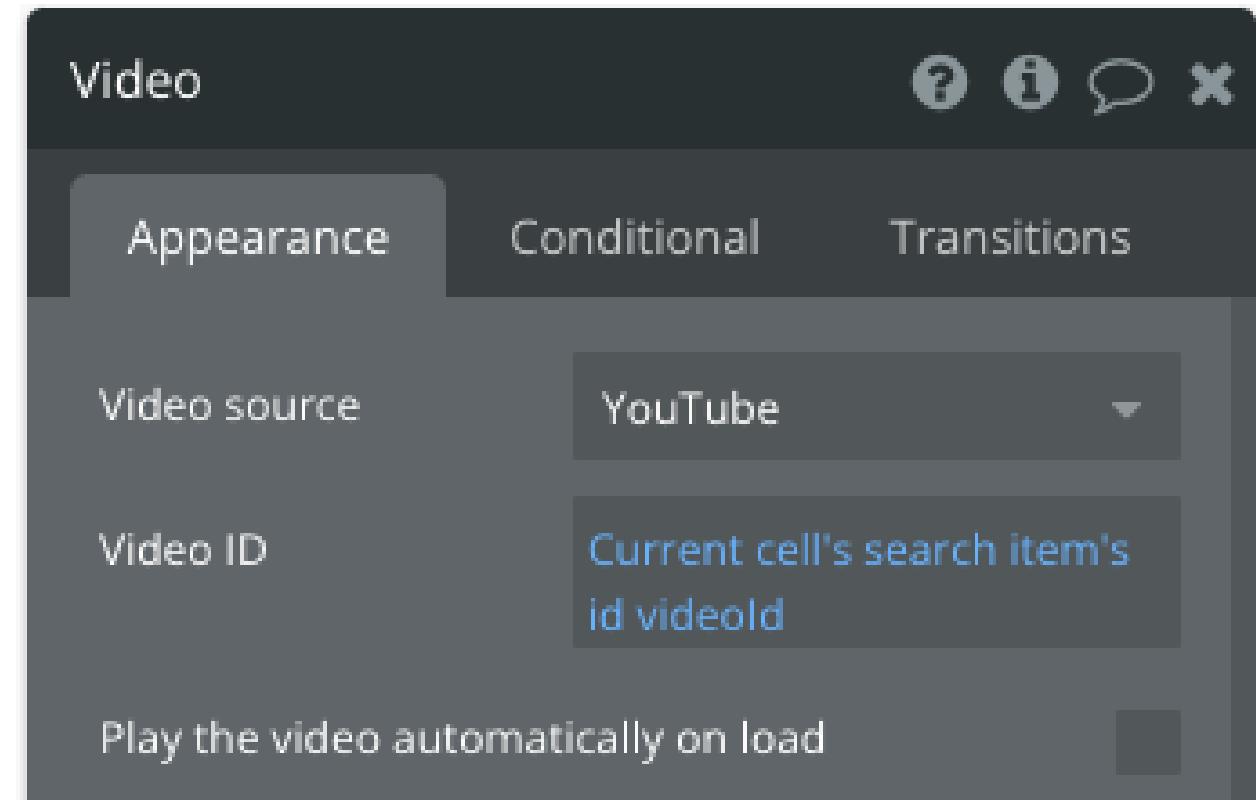
- Next, we will set up each cell, but first let's set it up.

- Hint 

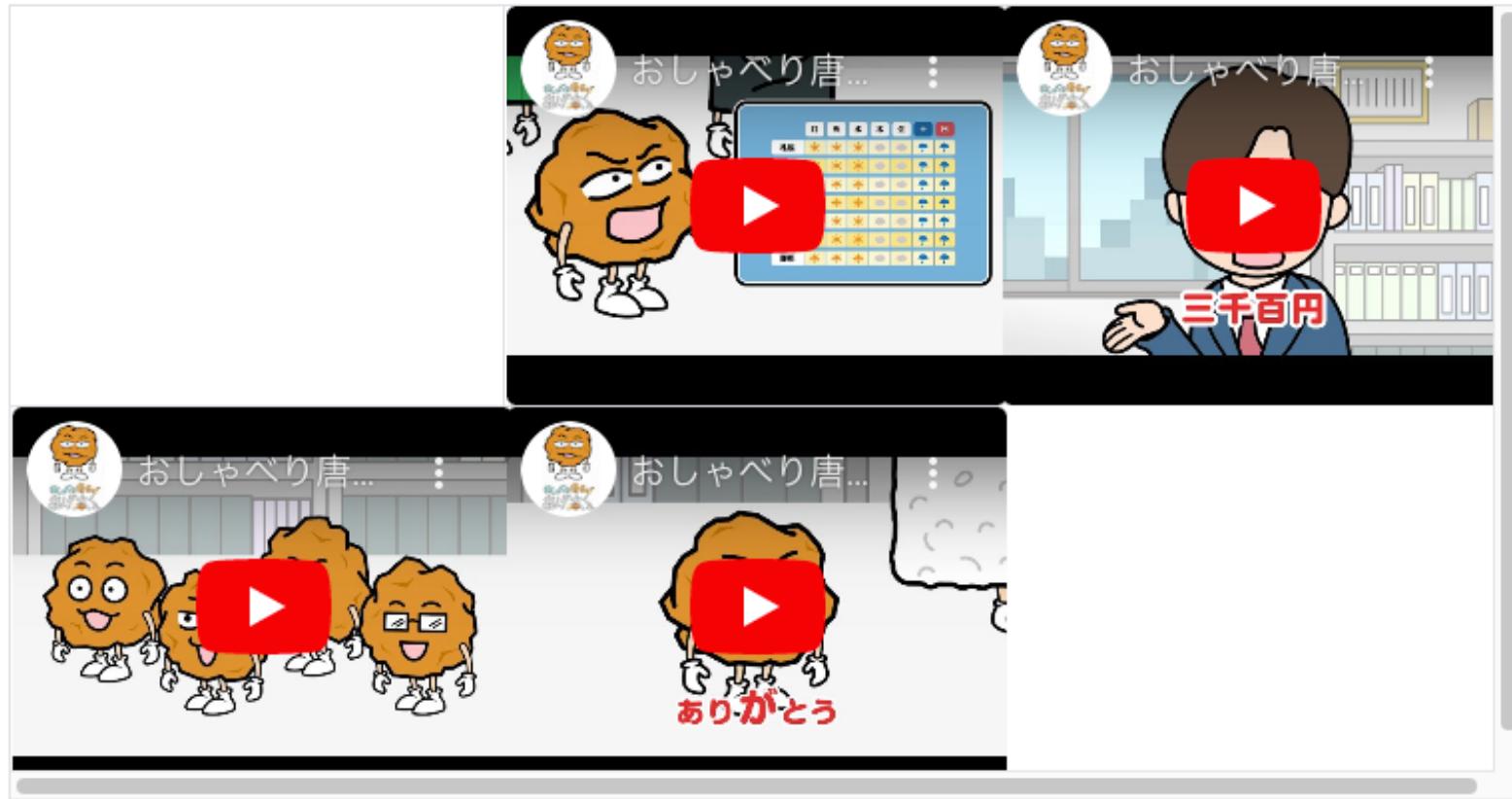
- Video source is YouTube
 - Then an item called Video ID will be displayed, so it seems like you should refer to the **Video Id** that you saw earlier when setting up the API.



- Something like this



- The setup is now complete! Let's preview it now!
- Did you see a list of videos for the keywords you specified in advance using the API?



Try Google Login

- Next, log in to the Bubble app using your Google account!
- First, install the Google plugin

The screenshot shows the 'Install New Plugins' section of the Bubble app. A search bar at the top contains the text 'Google'. On the left, there are two filter sections: 'Types' and 'Categories'. Under 'Types', 'Action' is checked. 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'. The main list displays four plugins:

- Google Material Icons**: Material design system library is a free library of icons built by Google, using Material Design principles. By Bubble. **Install**
- Google Analytics**: Google Analytics lets you track visits on your website and get some insights on your audience for free. By Bubble. **Install**
- Google**: Let users log in with their Gmail account, and fetch their profile data. By Bubble. **Install** (This plugin is highlighted with a red border)
- Google Places**: Fetch restaurant, schools, etc. from the Google Places API. By Bubble. **Install**

A 'DONE' button is located at the bottom right, and a warning message at the bottom states: **⚠ Recurring subscriptions to paid plugins are on paid plans only. Click here to adjust your application plan.**

- Select installed Google plugins
- Check the `Use a generic redirect URL` box displayed on the right panel
 - Make a note of this URL as well, but since you cannot select it on the screen, please replace the app name part of the URL below with your own app name and write it down 😊
 - `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

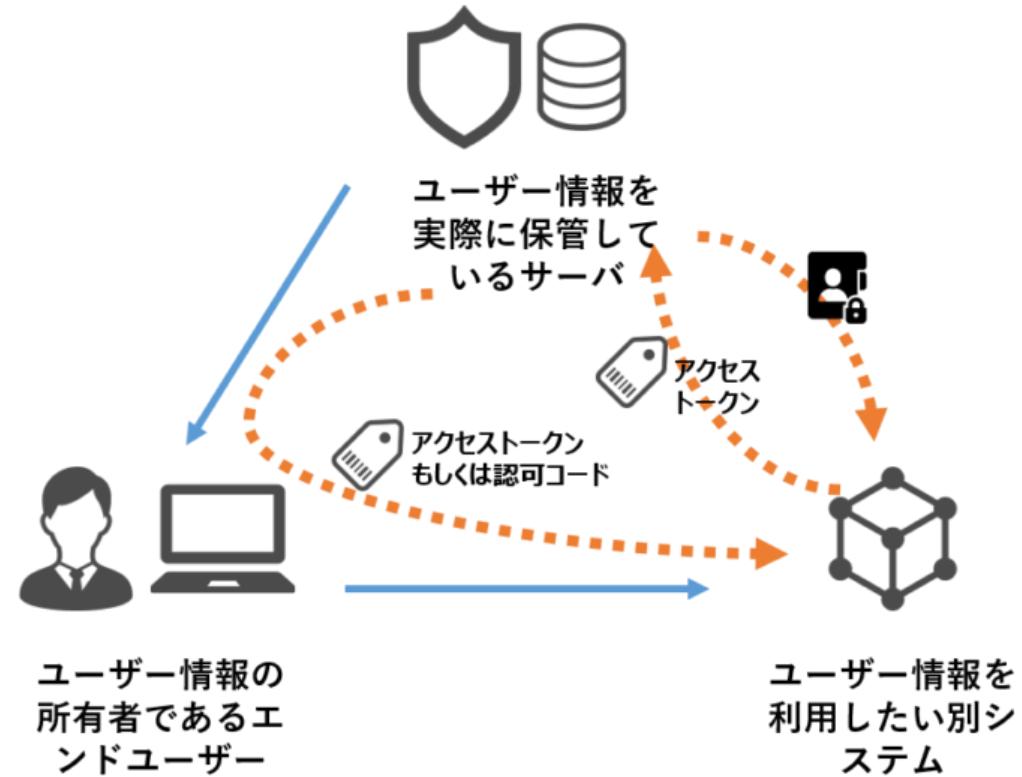
This plugin is an authentication provider.

What is OAuth?

OAuth is a mechanism on the Internet that allows one service to authenticate and access resources without revealing its password to another service. If you want to use another service (e.g. a pet management service) but don't want to create a new account to do so, you can use OAuth to log in with your existing account (e.g. Google or Facebook account).

How it works

- Login request: Select "Login with Google Account" for the pet management service.
- Permission confirmation: Google asks, "Can I use some of your information for this service?"
- Passing a "key": If you give permission, Google will pass a special "key" to the service. That service then has access to limited information, such as your name and email address.



Preparing for authentication to return to the app after logging in to Google

- Access the Google Cloud Platform screen

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

- Display the same API and service screen as before

- Select the project and select "OAuth consent screen" from the left menu

The screenshot shows the Google Cloud Platform interface. The top navigation bar includes the Google Cloud logo, a user dropdown for 'bubble-naotake', a search bar with placeholder text '検索 プロダクト、リソース、ドキュメント...', and various navigation icons. On the left, a sidebar titled 'API & サービス' lists several options: '有効な API とサービス' (selected), 'ライブラリ', '認証情報', and 'OAuth 同意画面' (selected). The main content area is titled 'OAuth 同意画面'. It contains a note about configuring and registering the app (selecting a target user type) and a section for 'User Type'. Two radio button options are shown: '内部' (selected) and '外部'. Below each option is a detailed description. A large blue '作成' (Create) button is at the bottom.

API & 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。](#)

- Select "External" for User Type and click Create

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 [?](#)

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。](#)



- Enter the required information on the next page

- App information
 - App name: App name created with bubble
 - User support email: your own Google account email address

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。

- Approved domains
 - bubbleapps.io
- Developer contact information
 - Your Google Account email address
- "Save and Next"

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*

[+ 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 *

 (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](#)

- Do not change the scope settings and click "Save and Next"

-Enter your girlfriend's Google email address as the test user

- After setting, save and continue

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 | trash |

保存して次へ キャンセル



- Registration is now complete, so we will issue authentication information next.
- Select credentials from the left menu and click "+ Create Credentials"
- Select OAuth Client ID from the submenu

| 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 | 有効な API にアクセス API キー OAuth クライアント ID サービス アカウント OAuth 2.0 クラ 表示する OAuth クライアントがありません | + 認証情報を作成 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 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 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 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 |

- The screen for creating an OAuth client ID will be displayed, so enter it as shown below and click "Save".
 - Application type: Web application
 - Name: 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

- Approved redirect URI: URL you recorded when installing the Google plugin on Bubble side

```
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

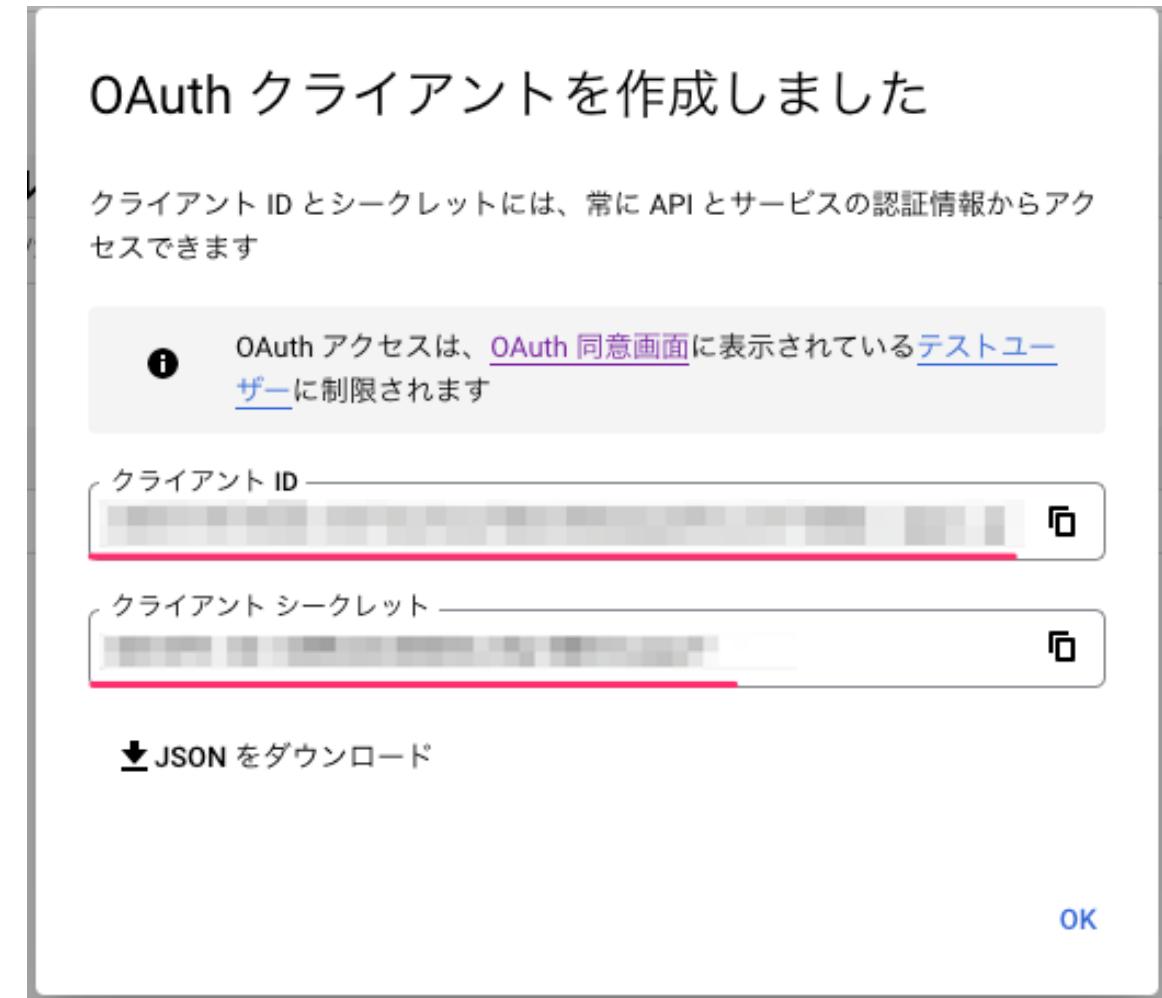
URI 1* — **https://titech2022-advance-naotake.bubbleapps.io/api/1.1/oauth_redirect**

+ 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

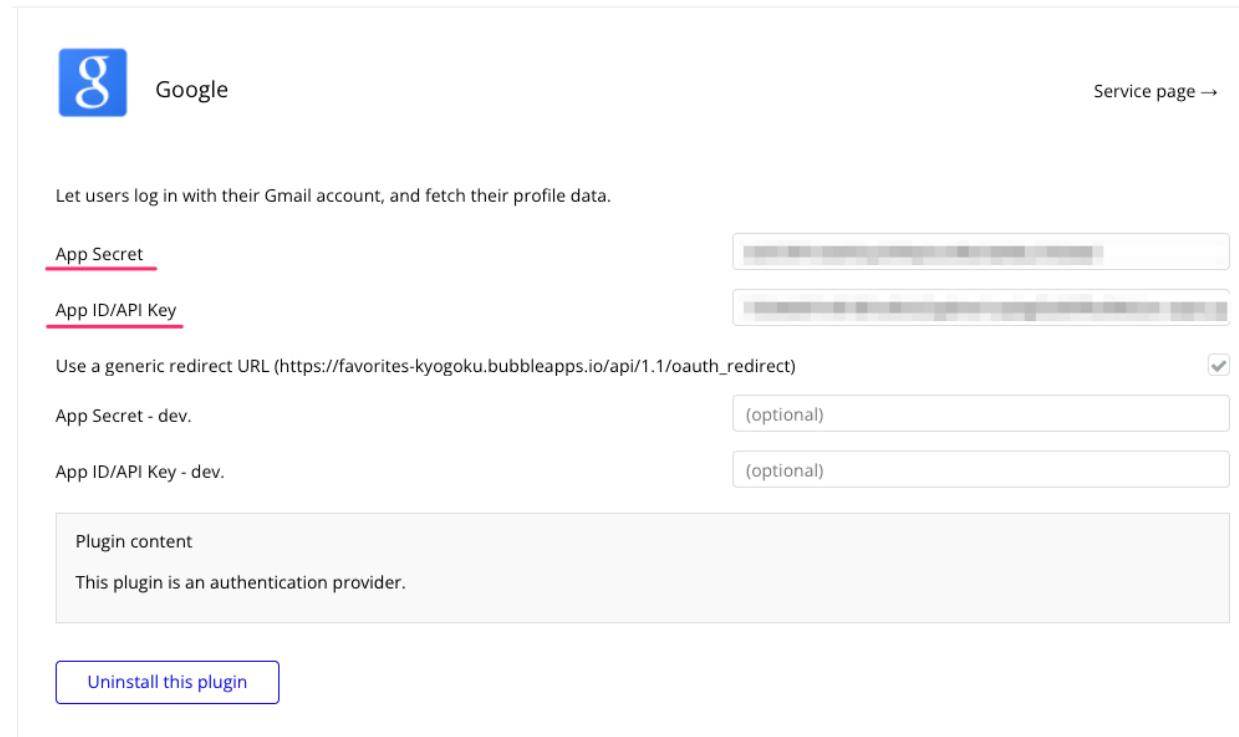
作成 キャンセル

- Click "Create"
- Then a popup will appear saying "OAuth client has been created" and the client ID and client secret will be displayed, so make a note of them.
 - If you have a memo and you might forget it, you can download the JSON.
- Make a note and close with OK



Set the issued authentication information on Bubble side

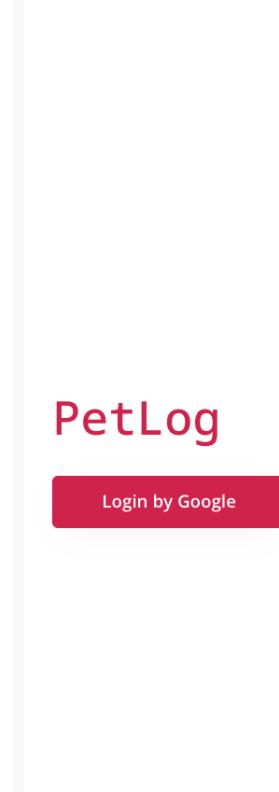
- Return to Bubble's Google plugin screen and enter the key you obtained earlier into each field.
 - Client ID → AppID/API Key
 - Client Secret → App Secret



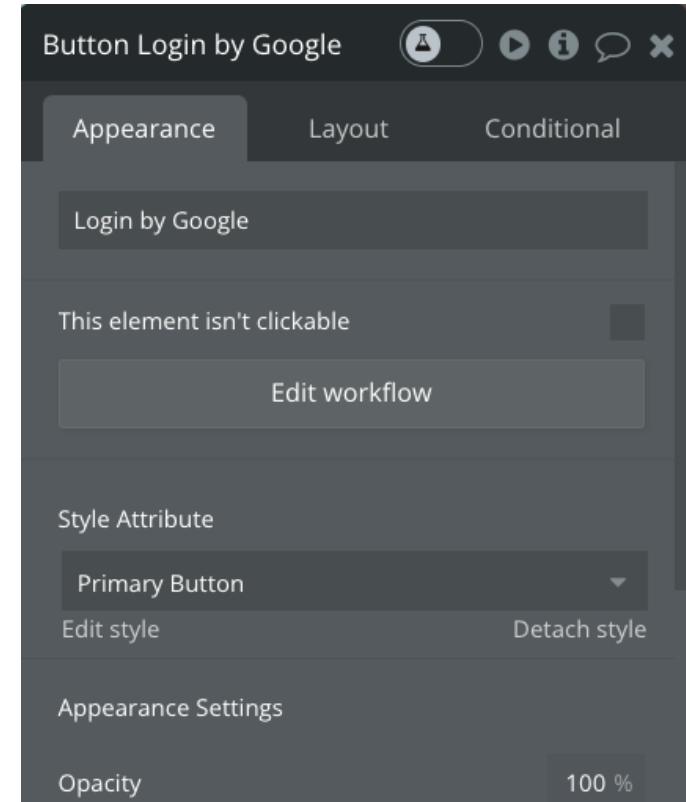
- Pre-configuration is now complete
- Next, we will create a login function

Replace login mechanism with Google

- Open the index screen and remove all signup text boxes and buttons
- Place a new button and change the label to `Login by Google`



- Click **Edit workflow** in the settings window of the button you placed to start creating the button click workflow



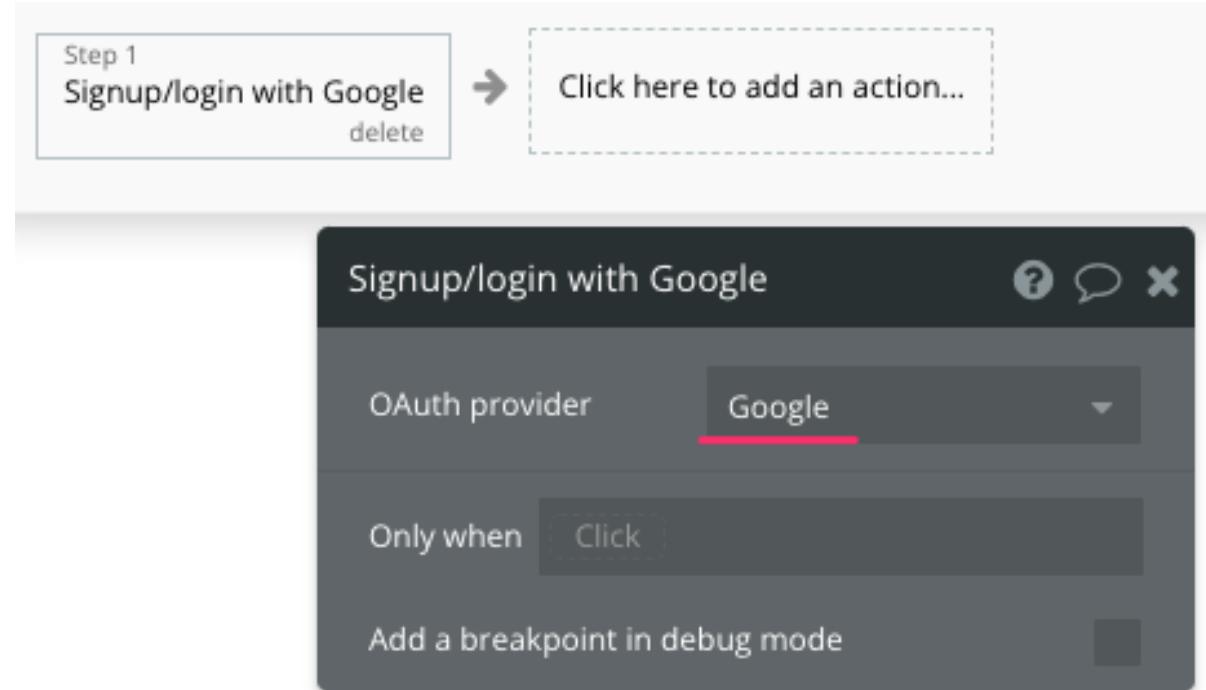
- Set up the Google Login workflow
- Select Click here to add an action...
- Select Signup/login with a social network from Account

Click here to add an action...

Search for an action...

| | |
|---|------------------------------------|
|  Account | Sign the user up |
|  Navigation | Log the user in |
|  Data (Things) | Signup/login with a social network |
|  Email | Log the user out |
|  Payment | Update the user's credentials |
|  Analytics | Make changes to current user |
|  Element Actions | Send confirmation email |
|  Plugins | Send password reset email |
|  Custom Events | Send magic login link |

- A Signup/login with Google popup will appear
- Select Google as OAuth provider



Specify that you will be redirected to the pet list after logging in.

- Select Click here to add an action...
- Specify Navigation > Go to page
- Specify pet_list as Destination

Now you can log in with your Google account. It's easy.

- As a precaution, let's display the account name and image at the top of the header once you have successfully logged in.

Pet Register



Log out

- Open the header of Reusable elements

The screenshot shows a user interface for managing website content. At the top, there are search and selection fields: "Search a page or a reusable..." and "Pick an element...". Below these are buttons for "Edit" and "Saved".

The main area is divided into two sections: "Pages" and "Reusable elements".

Pages Section:

- index
- pet_list
- pet_weight_register
- reset_pw
- Add a new page...

Reusable elements Section:

- pet_detail
- pet_register
- video_list
- 404

Bottom Left:

- header
- Add a new reusable element...

- Switch to the Design tab and display the following three components that are currently hidden from the Elements tree.
 - If you do not do this, it will be difficult to understand the image after adding the image.

The screenshot shows the UI Builder interface with the 'Design' tab selected. The sidebar on the left lists categories: Workflow, Data, Styles, and Plugins. The main area displays the 'Elements tree' with the following items:

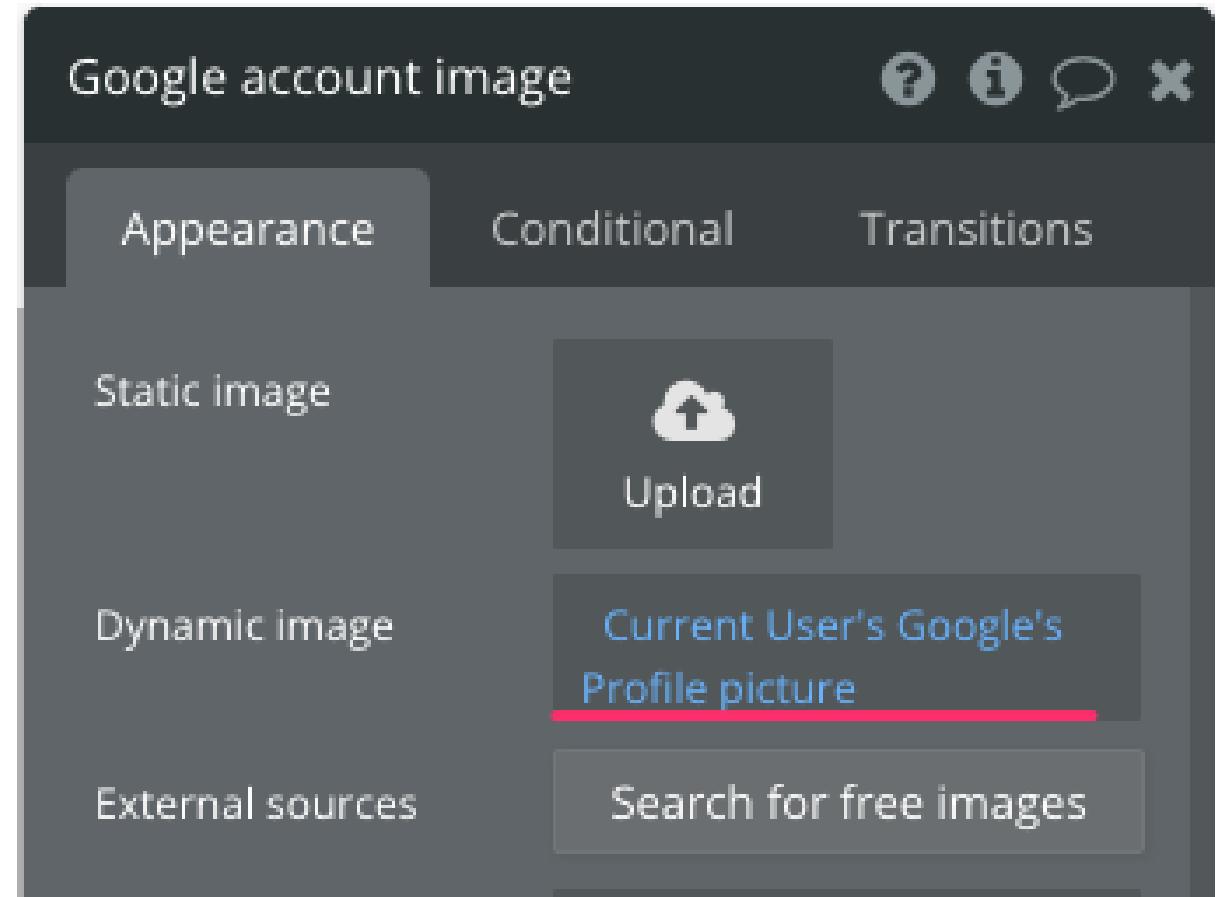
- Only show hideable
- Reusable header
- Group Header Standard wi...
- + GroupFocus Menu
- Group Header Standard
- Image Logo
- Group Menu
- Icon Menu
- Group Buttons
- Link Pet Register
- Button Log In (Head...
- Button Log out (Hea...

- Select Image from Visual elements and place the image between the "Pet Register" link and the "Log in" button
 - Make it a square with dimensions 50 x 50.



- and set the image source
- Bubbler users will have an idea of how to set it up.
- Dynamic display settings are Dynamic image
 - What you want to display is the profile image of the currently logged in user's girlfriend's Google account.

- Something like this



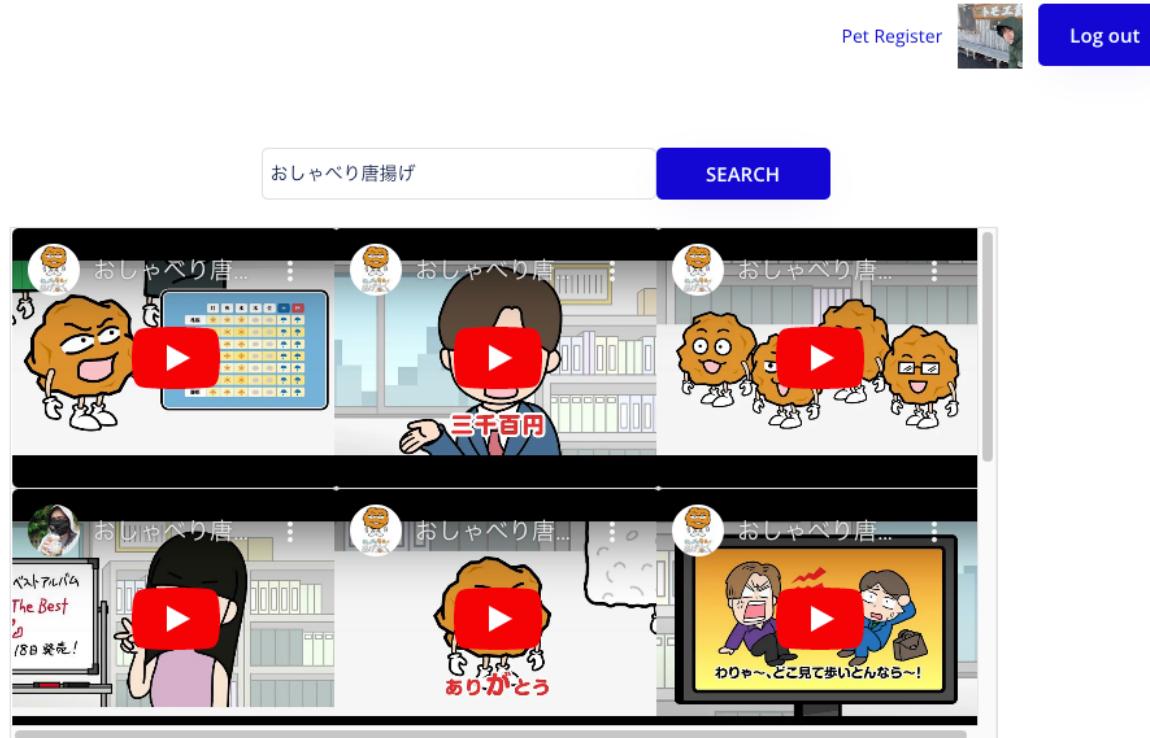
- Let's preview it!
 - In advance, if there is any data in the Data User with the same email address as the Google email address that you will use to check the operation, please delete it.
- When you press the login button, you are taken to the Google login screen, and when you log in there, you are returned to Bubble, right?
- By the way, when you log in, the email address information of the Google account you logged into will be registered in the User data on Bubble.
 - Of course, your password is not saved, so don't worry 😊



<Advanced>

Finally, let's incorporate the search function using the YouTube API

.bubble



<Advanced>

- Currently, YouTube API authentication keys and search keywords are fixed.
- So, let's change this to the following
 - Obtain the authentication key from the logged in user information (add a new field)
 - Search keywords can be entered and searched from the screen

<Advanced>

I will write some hints on the next page, so let's try it first!

<Advanced>

- First, add a field called `key` to User and set it in advance.
 - Normally, it would be natural to have the user enter the information themselves, but due to time constraints, we will omit the API key setting screen this time.
- If you want to dynamically set API parameters during processing, first remove the fixed value and uncheck `Private`, then you can specify the parameters when calling the API with DataSource etc.
- Place new search elements (text box and search button) on the video_list page, and when the search button is clicked, search via YouTube API and set the results to Repeating Group.



<Advanced>

- Explain important points
- First, add a field called `key` to the User type

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>

- Then, set the authentication key (API key) you obtained this time to the registered User in advance.

Data types Privacy App data Option sets File manager

Database views Application data - All Users - Development version

New view Primary fields Search Search view names or data types...

1 entries (displaying 1) New entry

| | | Email | key |
|--------------------------|---|--------|--------|
| <input type="checkbox"/> | <input type="checkbox"/> Run as → kyougoku2bubble@gmail.com | AlzaSy | AlzaSy |

All Users

<Advanced>

- Next, the settings on the YouTube API side are like this.

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' with value 'key', 'q' with value 'q', 'maxResults' with value '10', and 'type' with value 'video'. Each parameter has checkboxes for 'Private', 'Allow blank', and 'Optional'. Below the parameters, there are buttons for 'Add parameter' and 'Add header'. At the bottom, there are two checkboxes: 'Include errors in response and allow workflow actions to continue' and 'Capture response headers'. A message states 'This call has been modified since you last initialized; consider reinitializing.' At the very bottom, there are buttons for 'Reinitialize call' and 'Manually enter API response'.

Name: search
Use as: Data
Data type: JSON
GET https://www.googleapis.com/youtube/v3/search
(use [] for params)

Headers

Add header

Parameters

| Key | Value | Private | Allow blank | Optional |
|------------|-------|-------------------------------------|--------------------------|--------------------------|
| key | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| q | | <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"/> |

Add parameter

Include errors in response and allow workflow actions to continue

Capture response headers

This call has been modified since you last initialized; consider reinitializing.

Reinitialize call Manually enter API response

<Advanced>

- The key point is that for items that you want to specify dynamically, delete the Value value and uncheck Private.
- This allows you to specify values for these parameters when using the API

The screenshot shows a configuration interface for a workflow step. 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' with 'Value' and 'Private' checkboxes, 'q' with 'Value' and 'Private' checkboxes, 'maxResults' with 'Value' set to '10' and checked 'Private' and 'Allow blank' checkboxes, and 'type' with 'Value' set to 'video' and checked 'Private' and 'Allow blank' checkboxes. There are also buttons for 'Add parameter' and 'Reinitialize call'. At the bottom, there are checkboxes for 'Include errors in response and allow workflow actions to continue' and 'Capture response headers', along with a note about modified calls and a 'Manually enter API response' button.

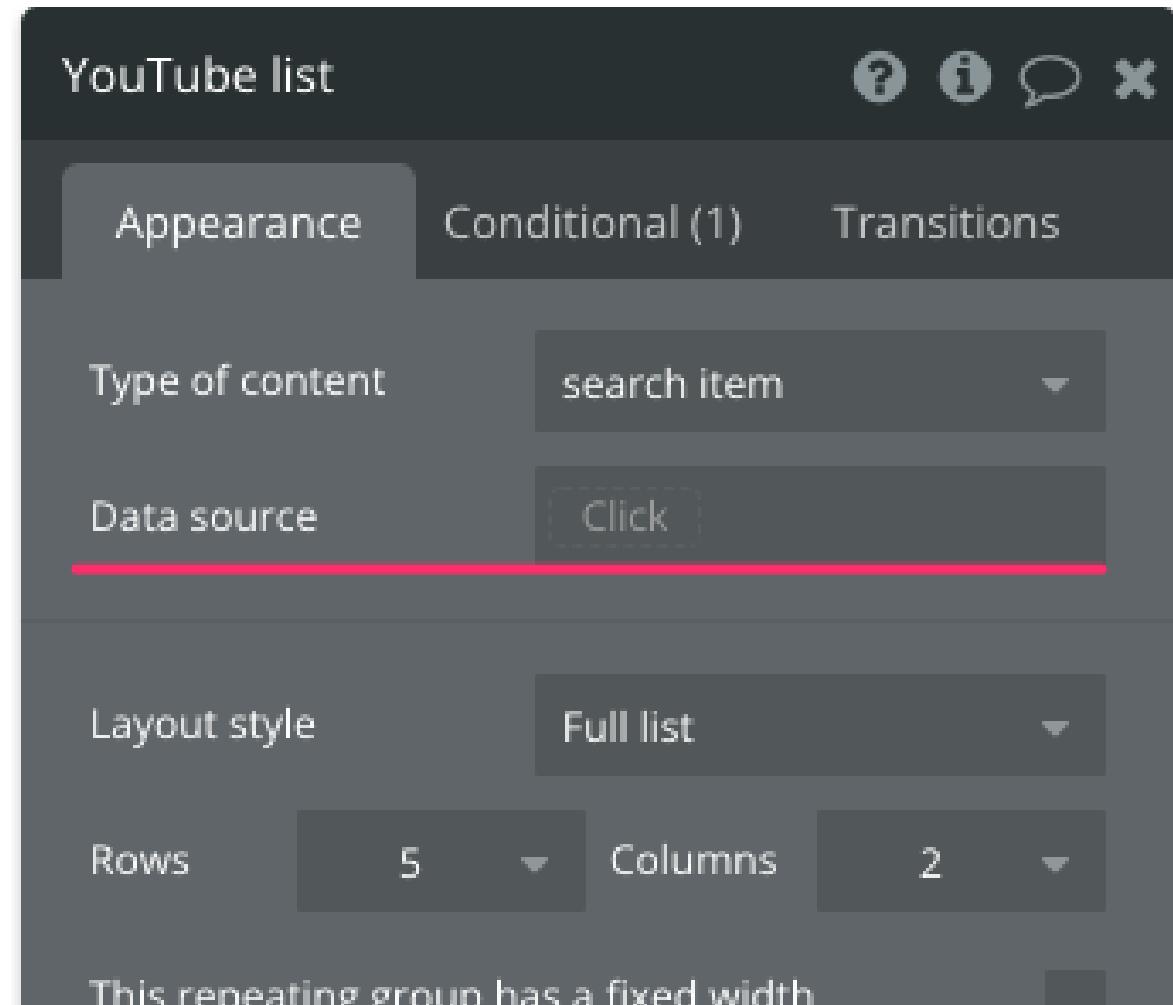
<Advanced>

- While we're at it, let's add two search options.
- key: maxResults , number of data to retrieve in one search
 - Default is 5
- key: type , type of data to search for
 - You can search only videos by specifying video

The screenshot shows a configuration interface for a search API call. At the top, there are fields for 'Name' (search), 'Use as' (Data), 'Data type' (JSON), and a URL field containing 'https://www.googleapis.com/youtube/v3/search'. Below this, under 'Headers', there is a button 'Add header'. Under 'Parameters', there are four entries: 'key' with value 'key', 'q' with value ' ', 'maxResults' with value '10' (which is highlighted with a red border), and 'type' with value 'video' (also highlighted with a red border). Each parameter entry includes checkboxes for 'Private', 'Allow blank', and 'Optional'. Below the parameters, there are buttons for 'Add parameter', '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.' with buttons for 'Reinitialize call' and 'Manually enter API response'.

<Advanced>

- Then empty the Repeating Group's Data source
- The target data to be displayed will be specified in the workflow of the new search function, so there is no problem even if this field is empty.



<Advanced>

- This will be the workflow setting for the next keyword search, so set the workflow for the newly prepared search button.
- This time, select "Display list" in "Element Actions" since the action will be for an element (Repeating Group).

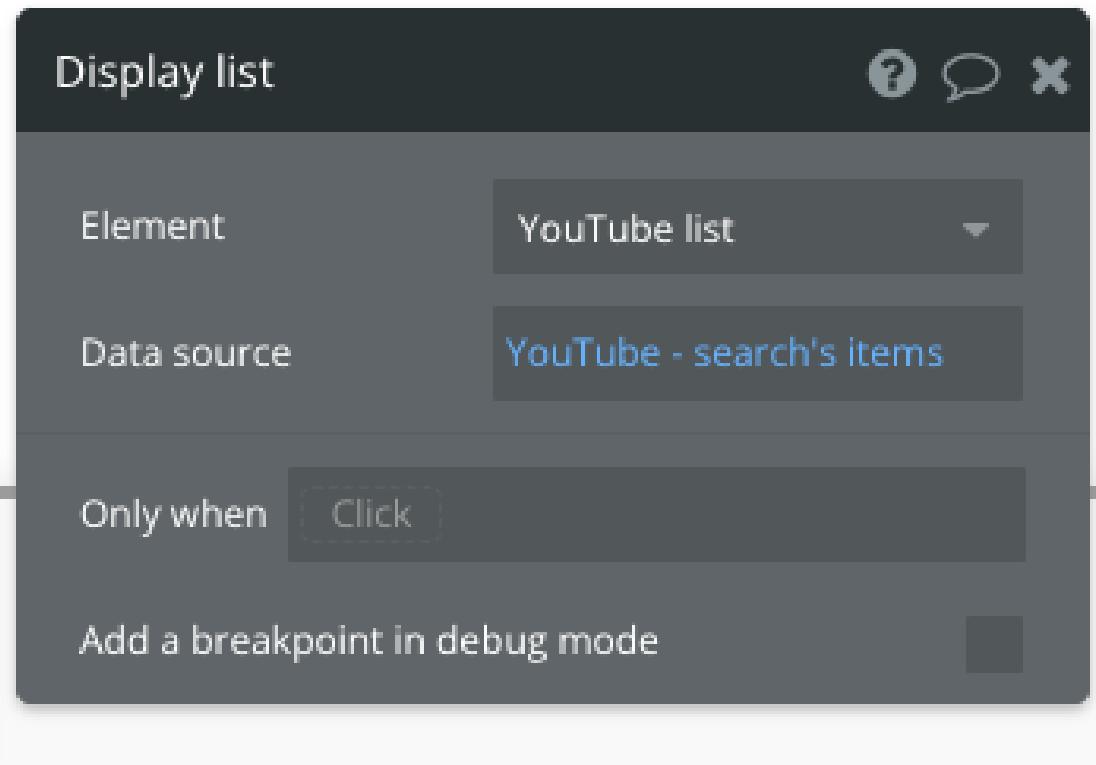
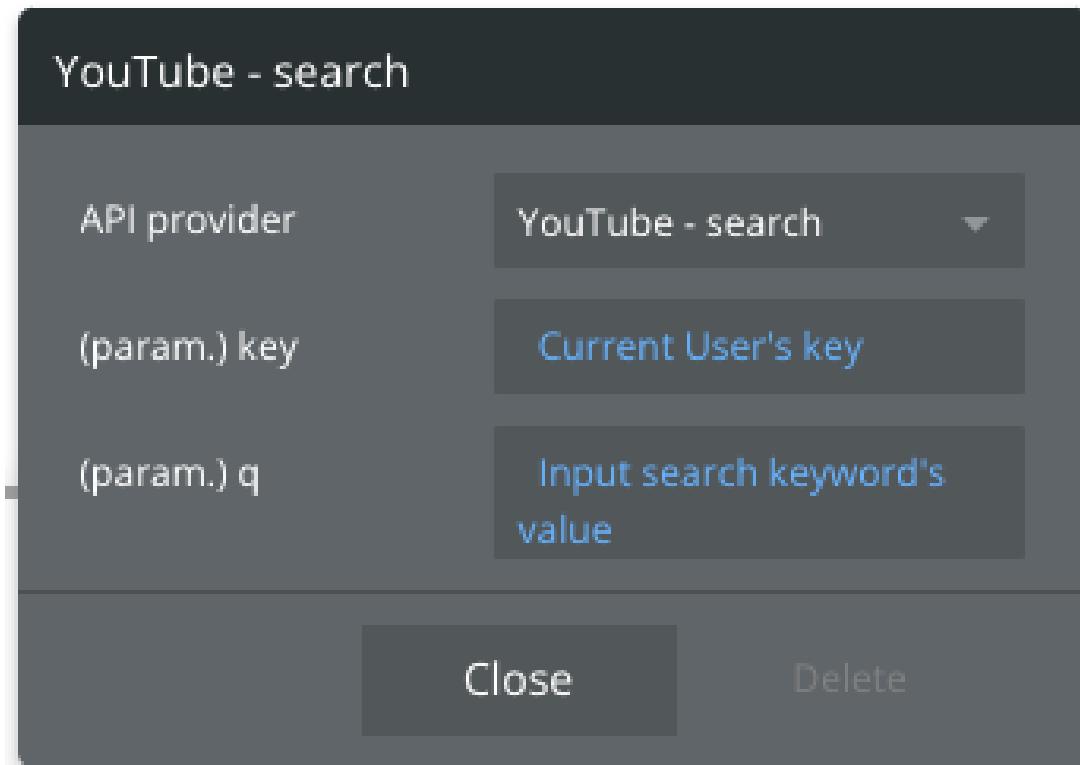
| Search for an action... | | |
|---|------------------------|-----------------|
|  Account | All elements | |
| Show | Hide | |
|  Navigation | Toggle | Animate |
| Scroll to | Set state | |
|  Email | <i>Input</i> | |
| Set focus | Reset inputs | |
|  Payment | <i>Group</i> | |
| Display data | Reset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

<Advanced>

- Select Get data from an external API for Data source to display data via API.
- And when you select YouTube – search as the API provider, you should see two params that weren't there before.
- This is because you previously unchecked Private for the parameters you want to specify dynamically in the settings on the YouTube API side.
- The value specified here is the usual Dynamic data.

<Advanced>

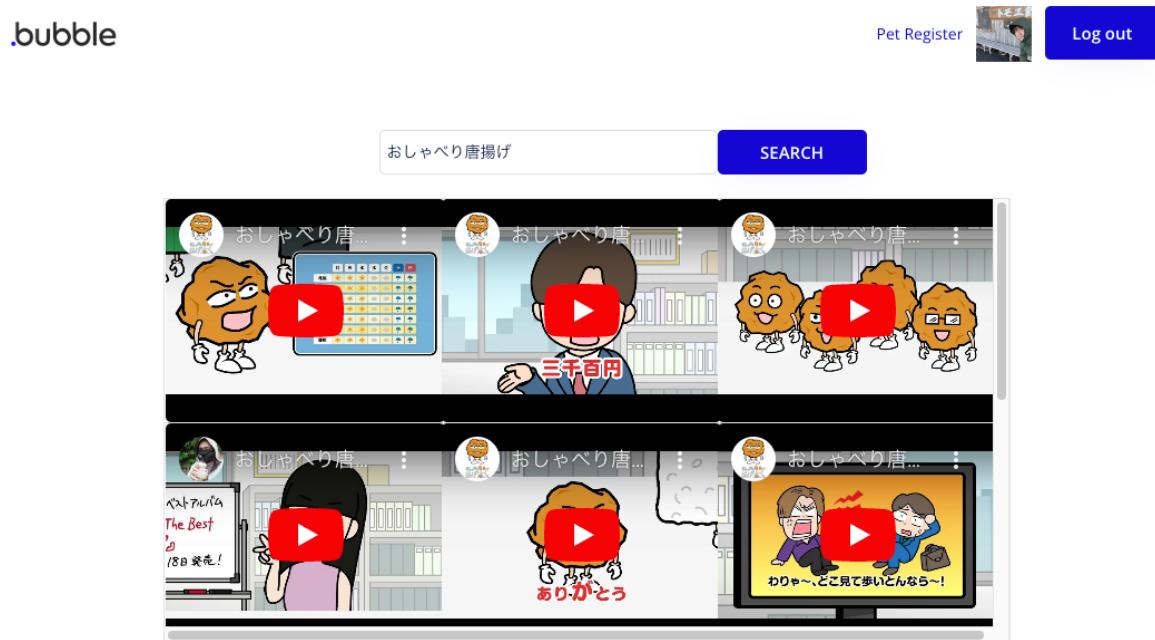
- This is what it looks like when you set it up



<Advanced>

Let's preview

- Did you see a list of videos you entered in the keyword field while logged in?



<Advanced>

- What did you think?
- I think the range of things that can be done with Bubble has been further expanded by linking with API!
- Some teams may use external APIs at the training camp, so let's take advantage of what we learned today and put API collaboration into practice!

Tips for collaborating with Bubble

Here's some useful information when collaborating with your team using Bubble.

How to invite team members on Bubble

To co-author an app with your team members...

- The team will prepare one account and everyone will log in and use that account.

- Bubble has a collaborative editing function using a regular method... 💰💰💰

| 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 ⓘ |

Notes on simultaneous editing with Bubble

- Avoid simultaneous editing on the same element (later edits will overwrite it)
 - To this end, we recommend assigning a person to be in charge of each screen and avoid editing the same screen at the same time.
- There are no other special considerations.
- Edits are reflected on other people's screens in real time.
 - Not only screen edits, but also database and Workflow edits are reflected in real time.
- If you divide the work between screens, development will be easy.

Team development exercise

- Try developing a new app with the DevelopmentPhase team
 - Make the app different from the one you plan to develop in DevelopmentPhase.
- You are free to use Adalo or Bubble
 - You can choose the one you plan to use in DevelopmentPhase, or you can choose both.
- Let everyone move their hands

Example of how to divide work

- Let's all work together to identify screens and functions and design the database first.
- Decide who will be in charge of each screen (registration screen person, list screen person, detail screen person, update screen person, etc.)
- Another option is for multiple people to work on the same PC. In that case, the person who operates the PC and the person who gives instructions should be alternated regularly.



Announcement of exercise results

Let's have each team present about the app they created during the exercise!

summary

- That's all for the learning phase lecture.
 - Have you mastered Adalo or Bubble?
 - Do you have an idea of how to develop a team?
- From the next session onwards, we will be in the Development Phase, where each team will brush up and develop the ideas they presented.
 - In preparation for that, let's reconfirm our ideas and select the tools (Adalo, Bubble, or something else?) to use for development.

lastly

- Delete the API key and OAuth information issued by Google Cloud Platform.

that's all!

Thank you for your hard work!