

Programming Boot Camp

# Bubble Basics

Tokyo Institute of Technology 2021/11/13

Naotake KYOGOKU

# Table of Contents

- What is Bubble?
- Registering with Bubble
- Overview of Application Development with Bubble
- Let's Start with Pet Registration and List Display
- Let's Make a Pet Detail Screen
- Let's manage your pet's weight
- Summary

## What is ## Bubble?

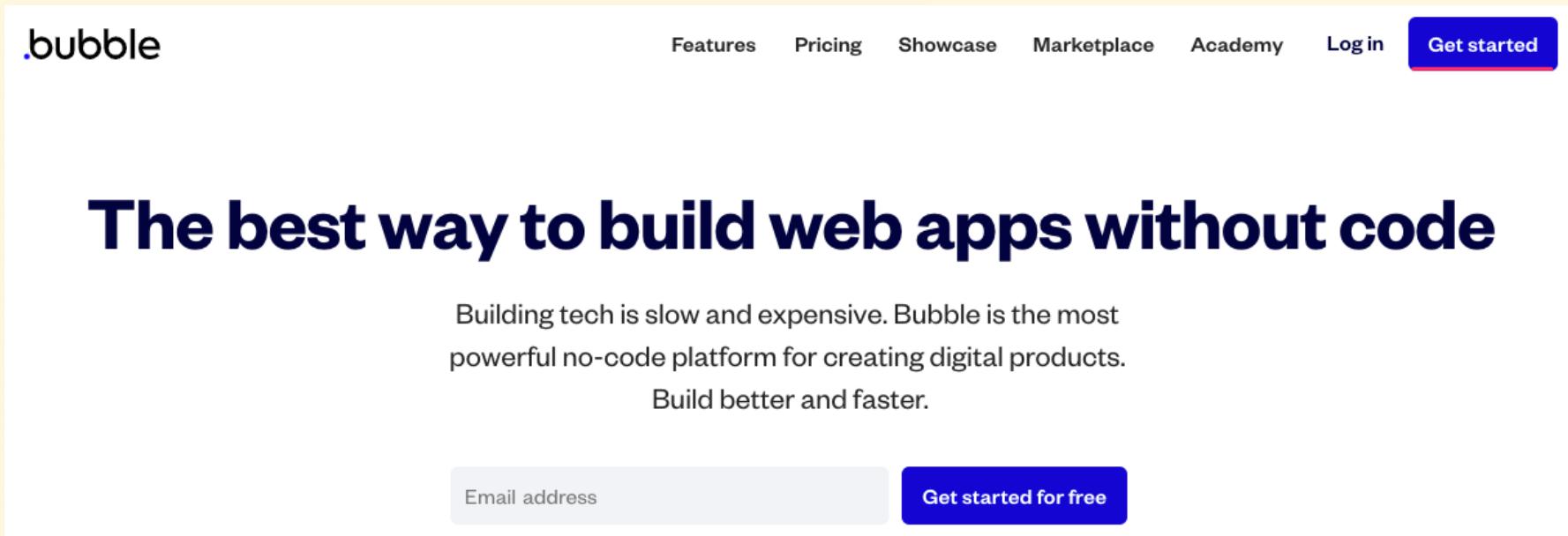
- Bubble is a visual programming tool, which means it is a no-code tool, but it requires a programming mindset.
- It is a no-code tool, but it requires programming ideas.
- However, the part where you can add movement to the placed parts requires programming ideas.
- Also, unlike Adalo, this application is developed based on the premise of a web application, and then adapted to be displayed on a smartphone.

## Examples of apps built with Adalo

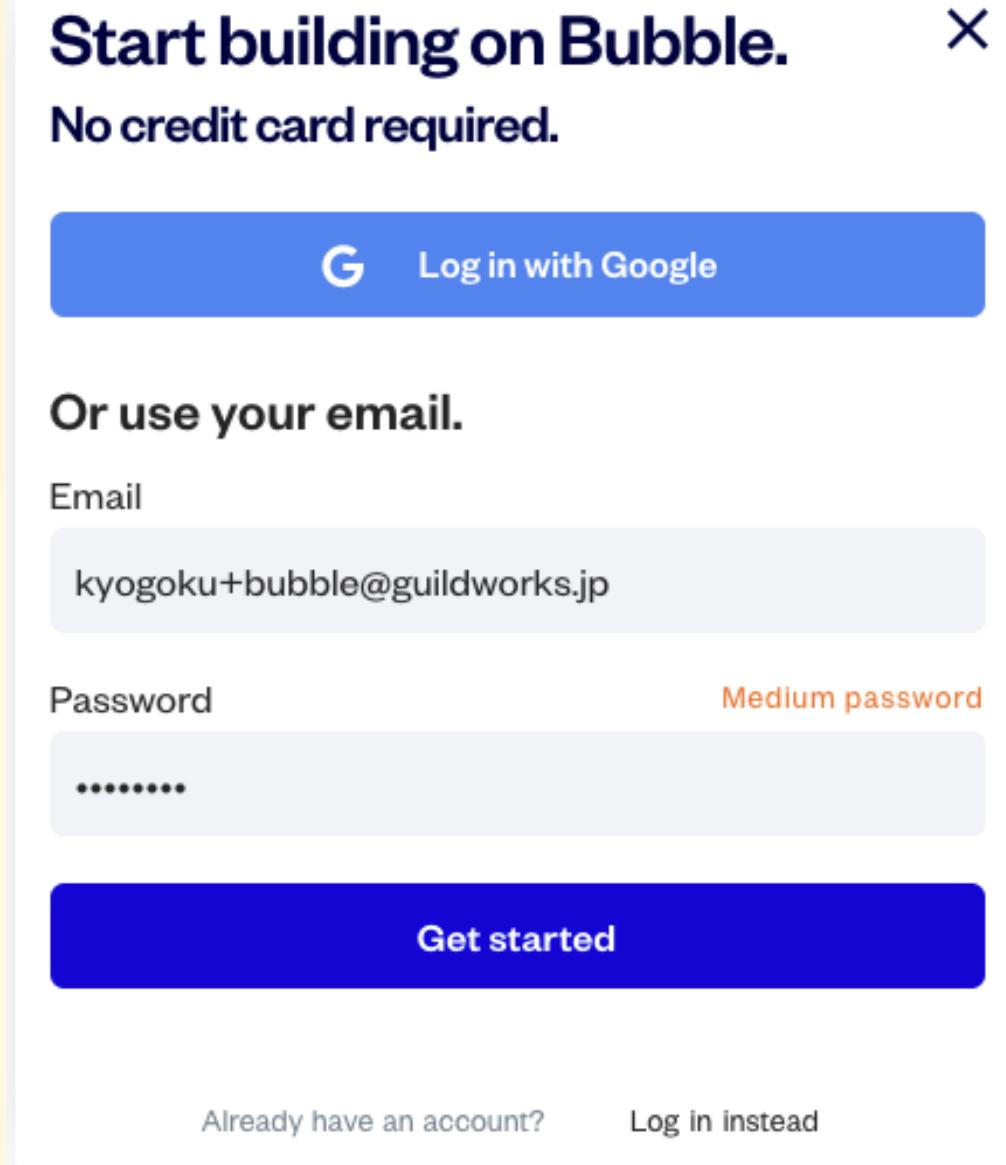
- LIBRIS: <https://libris.ne.jp/>
  - Released in March 2021
  - A search site to find the bookstore you want to visit
  - You can search for bookstores all over the country, including privately owned stores, book cafes, libraries, etc.
  - The developer is said to be a non-engineer who loves books
- More
  - <https://nocode-freaks.com/?p=200>
  - There was also a game...

# Register with Bubble

- Go to the top page of Bubble and enter your email address to register.
  - <https://bubble.io/>
  - Get started Click



- You can register for a free account
- Enter your email address and password and click "Get started"!



- I agree to the terms of use and "I agree..." Click!

## Agreeing to terms

Before you can use Bubble, you need to accept Bubble's Terms of Service. In particular, you must agree to Bubble's

- [Terms](#) - our Terms of Service are a legal agreement between you and Bubble and govern your use of our services.
- [Privacy Policy](#) - your privacy is important to us. We will only process your personal data as described in our Privacy Policy.
- [Cookies](#) - you consent to the placing of cookies on your computer or device to make this website better in accordance with our Privacy Policy.

Please read these documents carefully and click "I agree to Bubble's terms, privacy policy, and use of cookies" to use Bubble. If you have not read these documents, or if you do not agree to be legally bound by them, click "I do not agree" and do not use the platform.

[I agree to Bubble's terms, privacy policy, and use of cookies](#)

[I do not agree](#)

- Answer a few questionnaires for use.
- Finally, "Submit (thanks!)"  
Click !

## Welcome to Bubble!

We have a couple of questions to help you get started.

### Who do you want to build apps for?

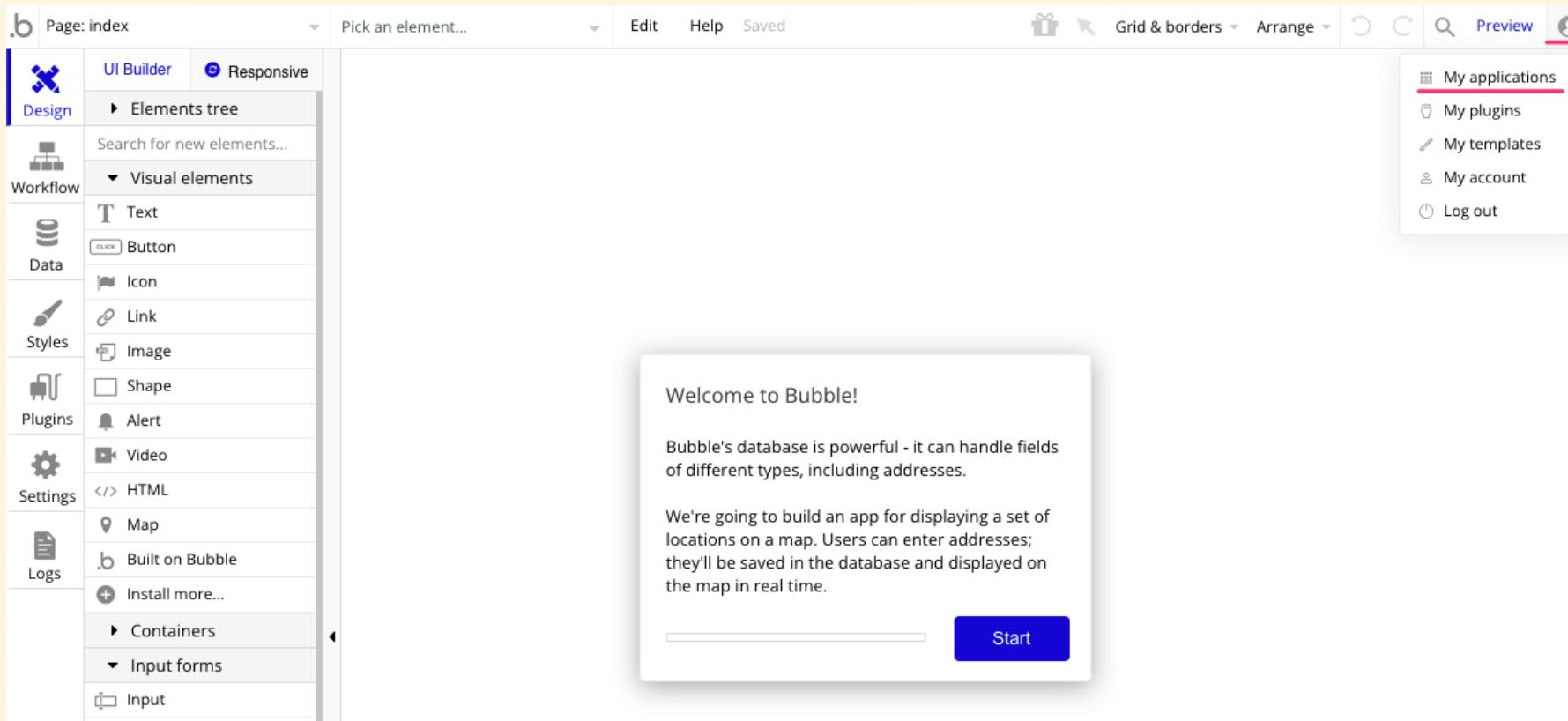
- Startup
- Small business
- Large, enterprise-size company
- Clients, as part of an agency
- Clients, as a freelancer
- School / university
- Non-profit or government agency
- Myself

Skip

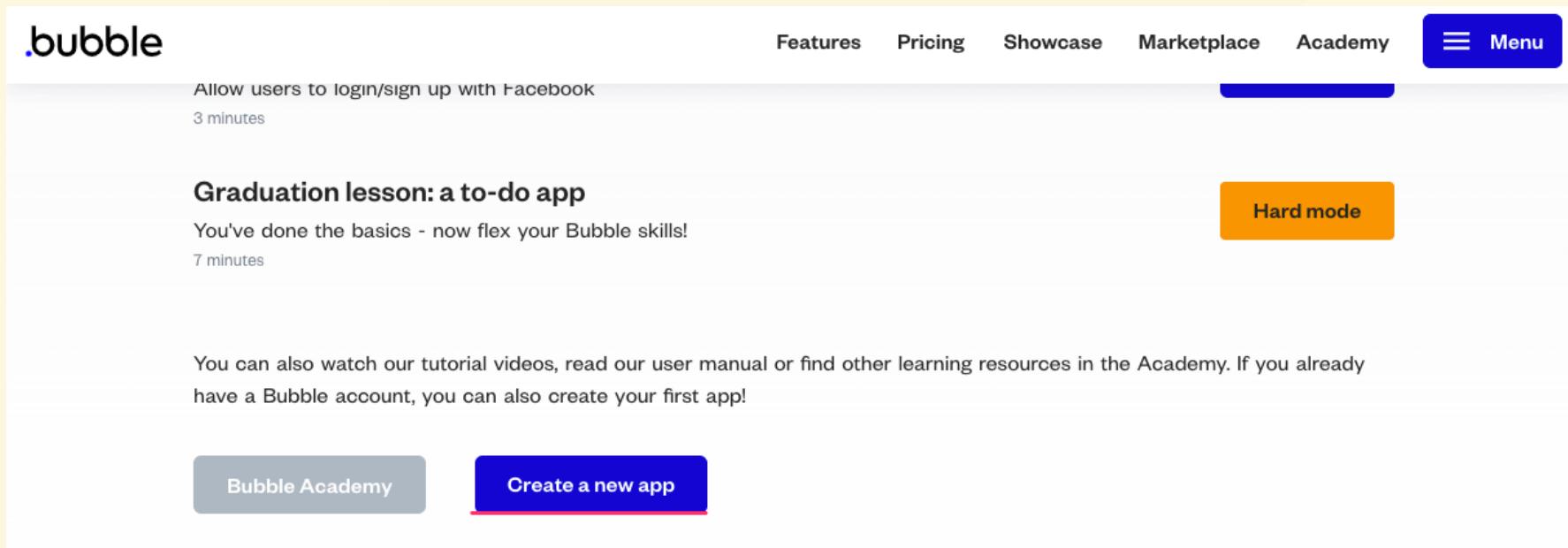


Submit (thanks!)

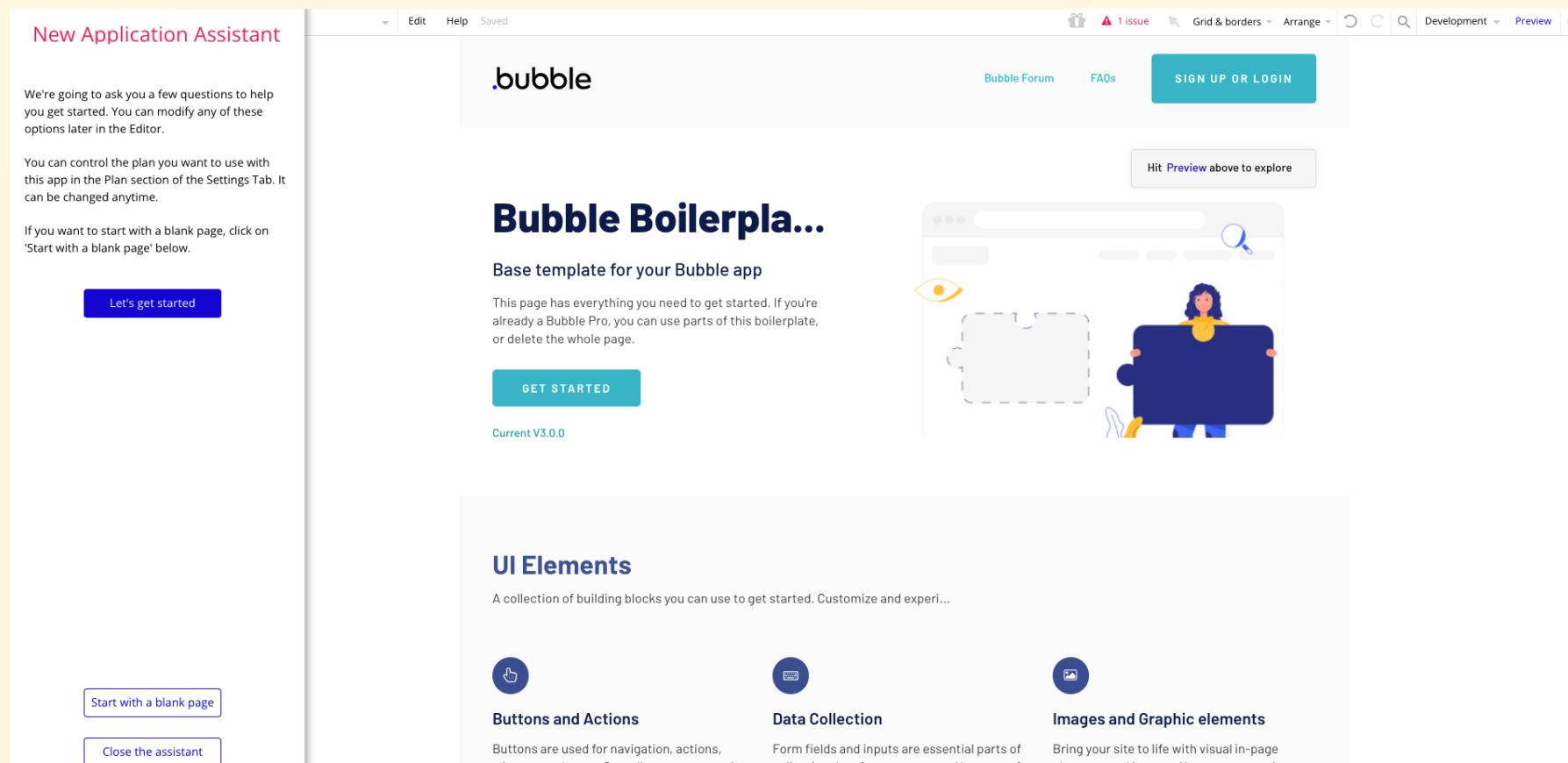
- This will bring up the editor screen and start the tutorial, which can be closed later.
- Click on the icon in the upper right corner of the screen and select "My applications".



- When the Welcome page appears, scroll to the bottom of the screen and click "Create a new app".

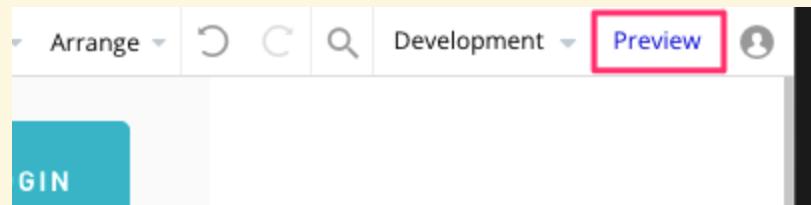


- If you see a screen like this, you are good to go
- This is the initial state of the template generated by Bubble.
- If you see the New Application Assistant in the left panel, select "Close the assistant" at the bottom



# Let's try to run the Bubble template first.

- Click the Preview button in the upper right corner of the screen.



- The preview screen will be launched.

.bubble

Bubble Forum   FAQs   SIGN UP OR LOGIN

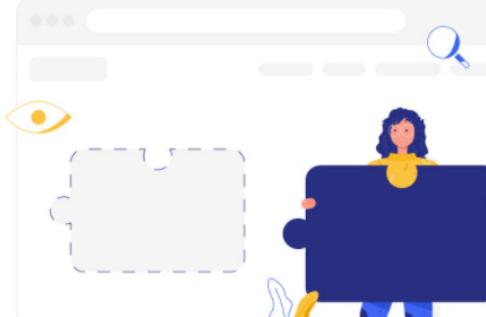
## Bubble Boilerplate

Base template for your Bubble app

This page has everything you need to get started. If you're already a Bubble Pro, you can use parts of this boilerplate, or delete the whole page.

[GET STARTED](#)

Current V3.0.0



### UI Elements

A collection of building blocks you can use to get started. Customize and experiment!

 <b>Buttons and Actions</b> Buttons are used for navigation, actions, triggers, and more. Set <code>rollover_states</code> , <code>active_states</code> , and <code>start/edit_workflows</code> .	 <b>Data Collection</b> Form fields and inputs are essential parts of collecting data from your users. Here are a few examples to get you started.	 <b>Images and Graphic elements</b> Bring your site to life with visual in-page elements and images. You can use static uploaded visuals or dynamic data driven ones.
---	--	---

Debugger   [Normal](#)   [Slow](#)   [Step-by-step](#)

index   [Inspect](#)

# Try to Sign up

- Click on "SIGN UP OR LOGIN" at the top right of the screen.
- Enter your Email / Password / Confirm password and click "SIGN UP".
  - Make a note of the Email and Password you entered (you will use them later).

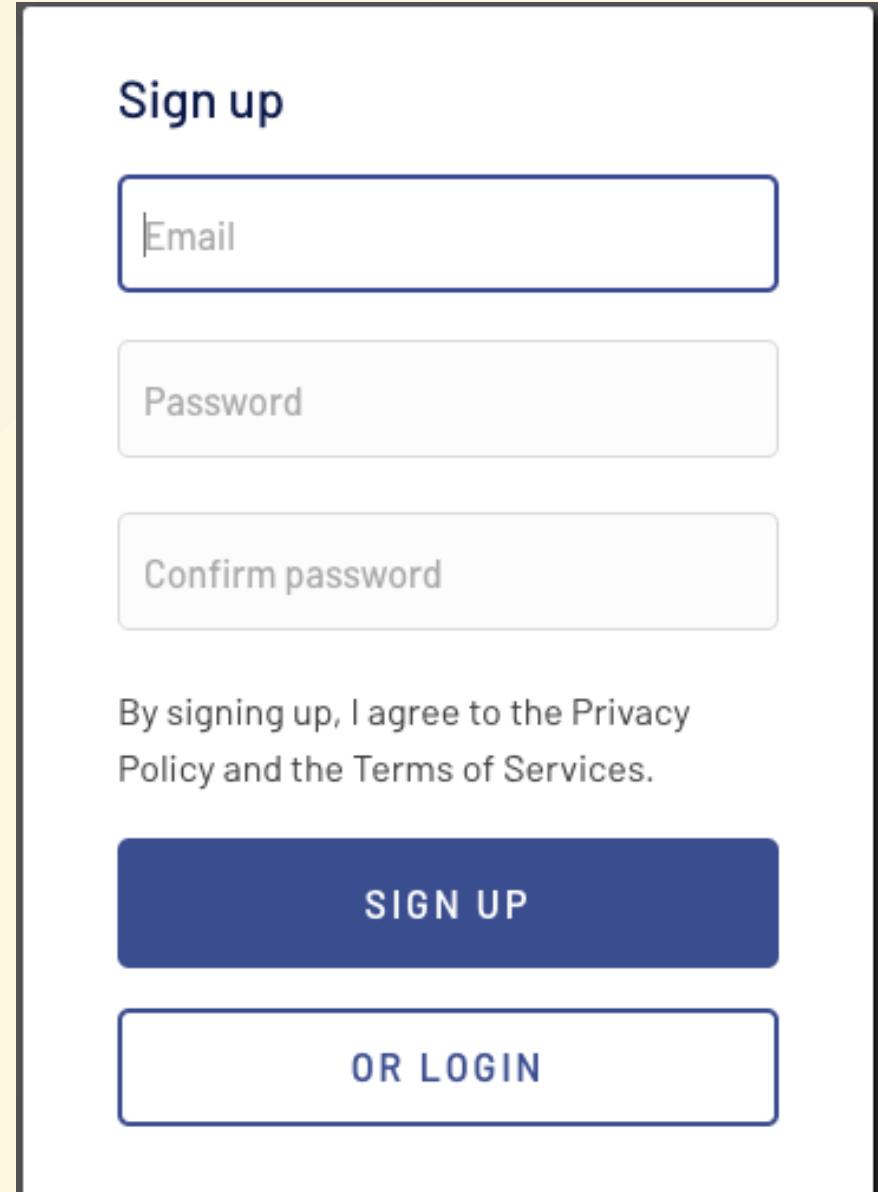
Sign up

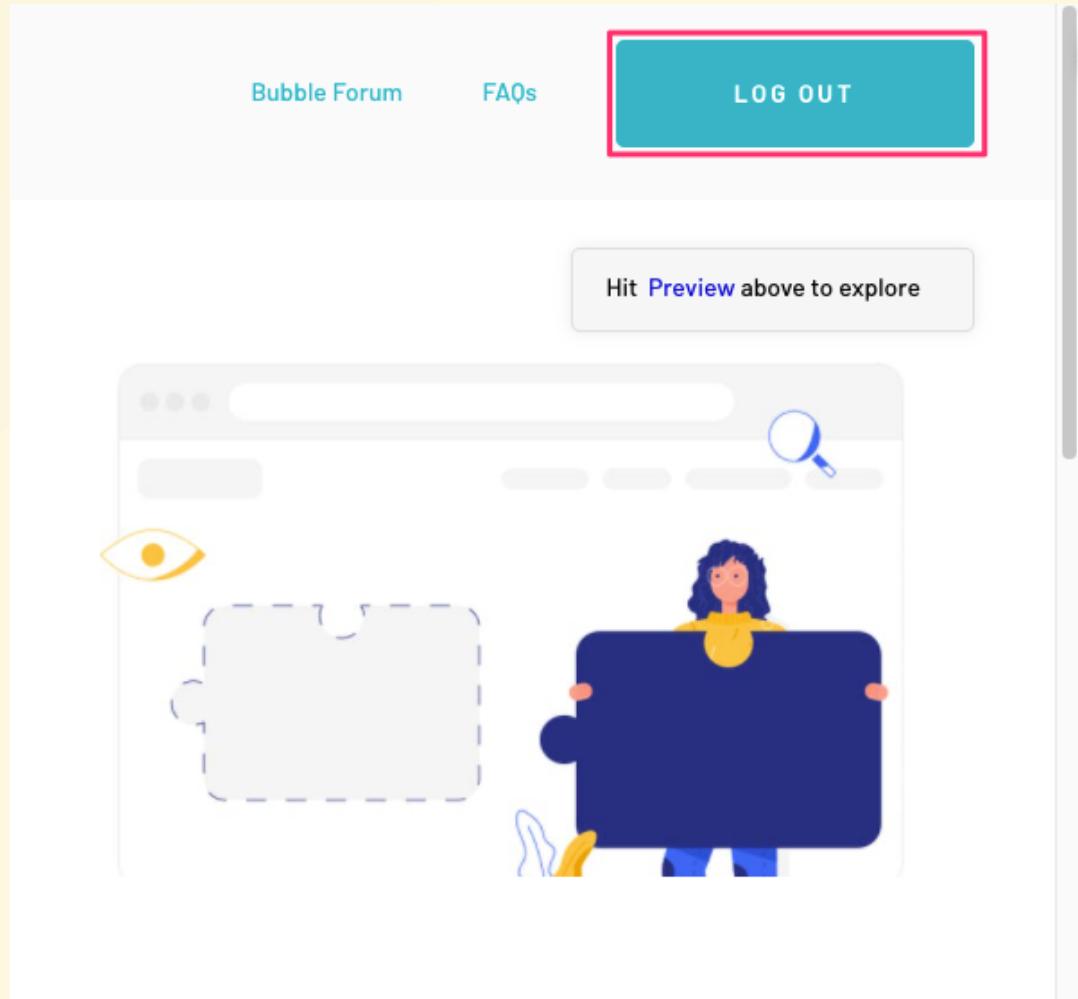
By signing up, I agree to the Privacy Policy and the Terms of Services.

**SIGN UP**

**OR LOGIN**

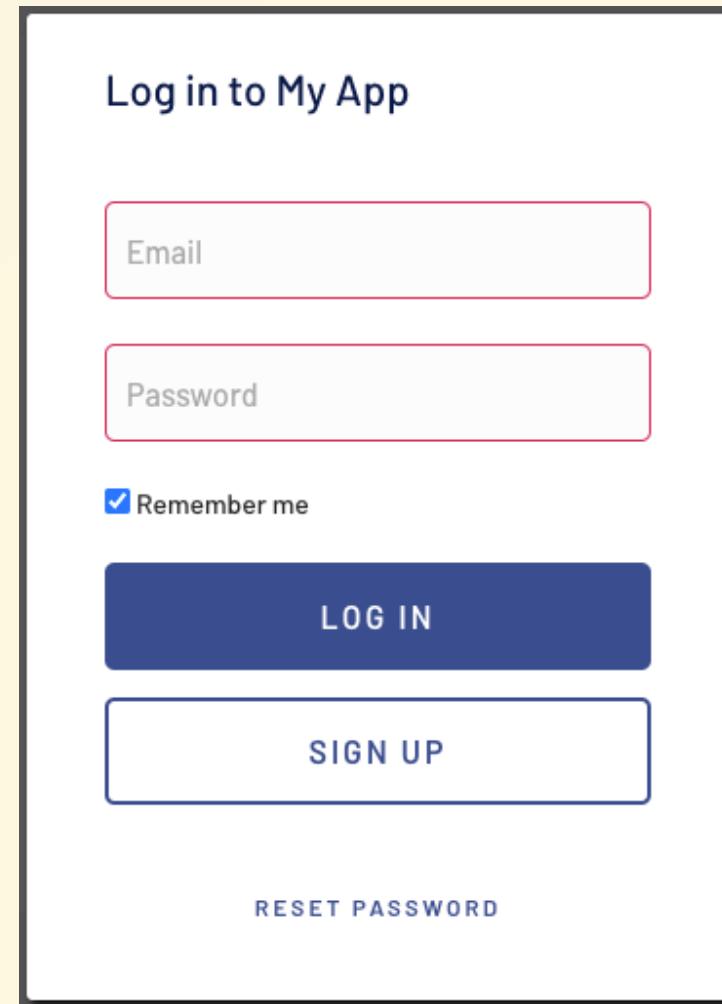


- The button on the upper right corner of the screen should now read "LOG OUT".
- This means that your account has been registered and you are now logged in.



# Try to Login

- Click "LOG OUT" in the upper right corner of the screen to log out.
- Click "SIGN UP OR LOGIN" in the upper right corner again.
- Click "OR LOGIN" at the bottom of the modal.



The image shows a login form titled "Log in to My App". It features two input fields: "Email" and "Password", both outlined in red. Below the password field is a checked checkbox labeled "Remember me". At the bottom of the form are three buttons: a dark blue "LOG IN" button, a white "SIGN UP" button with blue text, and a small link "RESET PASSWORD" in the bottom right corner.

- When the login modal appears, enter the email address and password that you just entered.
- Finally, click "LOG IN".
- If successful, the button in the upper right corner of the screen should change to "LOG OUT".
- Now you can see that the sign-up and sign-in process is working.

The image shows a login interface titled "Log in to My App". It features two input fields: "Email" and "Password", both outlined in red. Below the fields is a "Remember me" checkbox with a checked mark. A large blue "LOG IN" button is centered below the checkbox. To the right of the "LOG IN" button is a white "SIGN UP" button with blue text. At the bottom of the modal is a "RESET PASSWORD" link.

Log in to My App

Email

Password

Remember me

LOG IN

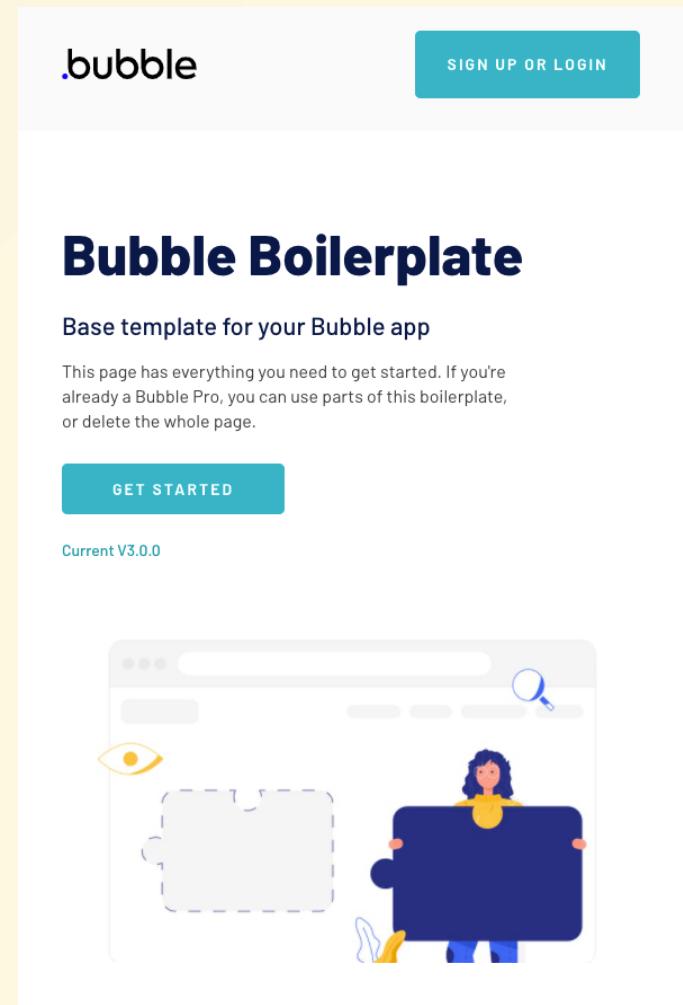
SIGN UP

RESET PASSWORD

## About mobile phone support

- As we mentioned at the beginning, the Bubble app is based on a web application.
- To make the Bubble app compatible with smartphones, we will use the concept of **responsive design**.

- Let's see it in action first.
- Try shrinking the width of your browser.



- The layout is not broken, and the elements are automatically rearranged to fit the phone size.
- This is called responsive design.
- Google explains it like this

Regardless of the user's device type (computer, tablet, mobile, non-visual browser), the It delivers the same HTML code at the same URL, but renders it differently based on screen size.

- The points are as follows
  1. the URL of the page being displayed is the same
  2. automatically change the rendering method according to the screen size

- To break it down a little more, it controls whether elements should stretch/shrink, wrap/unwrap, and display/not display depending on the screen size.
- By configuring these settings at each item level, you can automatically support responsive design.
  - During actual development, it is a good idea to set up the settings and preview them to make sure they work.
- We'll see how to set up responsive design in the screen operations later.

# An overview of application development in Bubble

The following is an overview of how to develop an app in Bubble.

# The three main operations

- In Bubble, there are three main operations that you can use
  1. design
  2. Workflow
  3. Data
- You can go back and forth between them from the left panel



# Design

- A mode for placing components on the screen to create a user interface.
- UI components in the left panel, actual screen editing area in the right panel

Page: index Pick an element... Edit Help Saved

UI Builder Responsive

Design

Workflow

Data

Styles

Plugins

UI コンポーネント群

Settings

Logs

UI Builder Responsive

Elements tree

Search for new elements...

Visual elements

Text

Button

Icon

Link

Image

Shape

Alert

HTML

Map

Built on Bubble

Install more...

Containers

Group

Repeating Group

Popup

Floating Group

Group Focus

Install more...

Input forms

Input

Multiline Input

Checkbox

.bubble

Bubble Forum FAQs SIGN UP OR LOGIN

Hit [Preview](#) above to explore

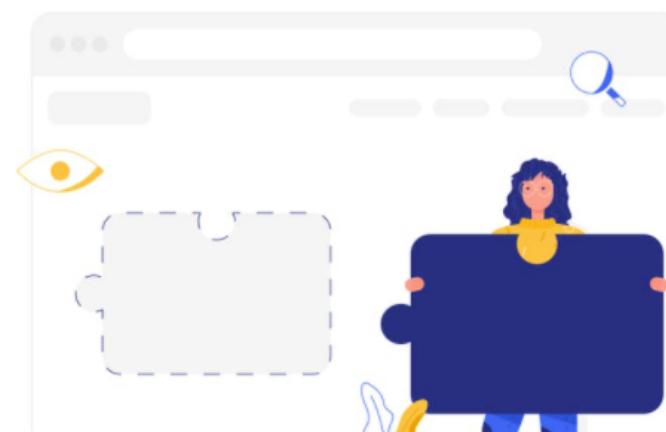
# Bubble Boilerpla...

Base template for your Bubble app

This page has everything you need to get started. If you're not already a Bubble Pro, you can use parts of this boilerplate, or delete the whole page.

[GET STARTED](#)

Current V3.0.0



## UI Elements

A collection of building blocks you can use to get started. Customize and experi...

# Workflow

- A mode for adding movement to the screen.
  - Example: Screen transition when a button is pressed
- This is the part where you need to think a little bit about programming.

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

- Header:** .b Page: index, Signup / Login Popup is clicked, Edit, Help, Saved, 1 issue, Development, Preview, User icon.
- Sidebar:** Design, **Workflow** (selected), Data, Styles, Plugins, Settings.
- Main Area:**
  - A grey box labeled "When Signup / Login Popup is clicked" is selected.
  - To the right of the trigger, a dashed box contains the text "Click here to add an event...".
  - Below the trigger, a large dashed box contains the text "Click here to add an action...".

# Data

- Mode for defining and manipulating data

The screenshot shows the Bubble Data interface. On the left is a vertical sidebar with icons for Design, Workflow, Data (which is selected), Styles, Plugins, Settings, and Logs. The main area has a header with tabs for Data types, Privacy, App data, Option sets, and File manager. Below this is a sub-header for Database views, Application data - All Users - Development version, with buttons for Copy and restore database and Switch to live database. A search bar and a 'New entry' button are also present. A table displays one entry with columns for Email, Created Date, Modified Date, and Slug. The table includes a 'Run as' button with the email address kyogoku+bubble\_test@guildwo. At the bottom, there's a 'All Users' section with a table and edit icons.

		Email	Created Date	Modified Date	Slug
<input type="checkbox"/>	<input type="checkbox"/>	kyogoku+bubble_test@guildwo	Oct 31, 2021 8:51 pm	Oct 31, 2021 8:51 pm	

- Select the "App data" tab and click the "All Users" link on the right side.
- You will see the list of registered users in the right panel, and the user you just "signed up" should be there.
- Now you can see that the screen and the database are connected!

The screenshot shows the Bubble app interface. On the left, a sidebar menu includes options like Design, Workflow, Data (which is selected), Styles, Plugins, Settings, and Logs. The main area has a header with 'Page: pet\_weight\_register', 'Edit', 'Help', 'Saved', and various status icons. Below the header, tabs include 'Data types', 'Privacy', 'App data' (selected), 'Option sets', and 'File manager'. Under 'App data', the 'Database views' section shows 'Application data - All Users - Development version'. It features buttons for 'New view', 'Primary fields', 'Search' (with a count of 1 entry), 'New entry', 'Delete (0)', 'Upload', 'Modify', 'Export', and 'Bulk'. A search bar below the buttons allows searching for view names or data types. A table displays user data with columns: Email, Created Date, Modified Date, and Slug. A single row is shown, with the 'Email' field containing 'kyogoku+bubble\_test@guildwo' and the 'Created Date' field showing 'Oct 31, 2021 8:51 pm'. At the bottom, links for 'All Pets', 'All PetWeightLogs', and 'All Users' are visible, with 'All Users' being the active tab.

		Email	Created Date	Modified Date	Slug
<input type="checkbox"/>	<input checked="" type="checkbox"/>	kyogoku+bubble_test@guildwo	Oct 31, 2021 8:51 pm	Oct 31, 2021 8:51 pm	

# Other Operations

- Styles
  - By naming styles, you can use them universally.
  - You can also specify individual styles for each part.
  - In this lecture, we will basically use common styles, so we will not cover them in this lecture.

The screenshot shows the Bonita BPM Studio interface with the 'Styles' tab selected in the left sidebar. The main area displays a list of styles:

- Alert - Standard Alert (Default)
- Alert - Warning Alert
- Button - Outline Button
- Button - Primary Button

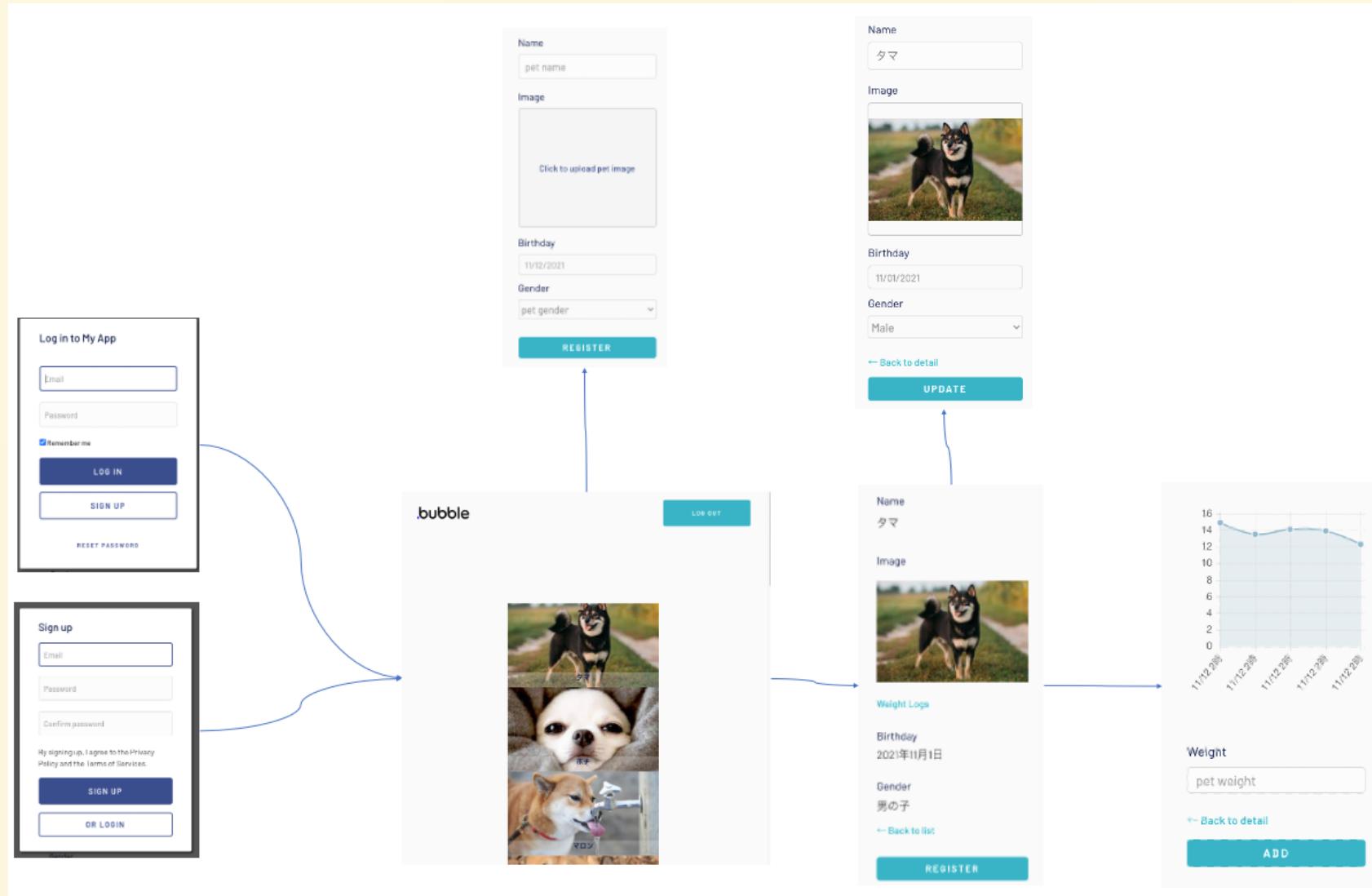
For the 'Alert - Standard Alert (Default)' style, there is a preview window showing a standard alert message. Below the preview are buttons for 'Find all elements using this style' and 'Define another condition'. The top navigation bar includes links for 'Edit', 'Help', 'Saved', 'Development', 'Preview', and a user profile icon. There are also icons for a gift, 8 issues, and search.

- Plugins
  - Just as we added a line chart component to Adalo, Bubble has a variety of components available to extend the application
  - In Bubble, these are called plugins.
  - There are both free and paid plugins for Bubble, so check them out before you use them!

- Settings
  - Change your Bubble plan and account settings
  - Centralize the messages provided by Bubble
    - Change the locale (language setting) for the main user of the application
    - Change this message to unify message management across the system
    - You can change the messages that can be used in common throughout the application, and you can also change the default messages.
    - It is also possible to add app-specific messages.
  - Not covered in this lecture.

- Logs
  - You can view the logs of the application as it runs.
  - However, since it is possible to view the logs while previewing the application, we will not cover this in this lecture.

# Overall view of the screen we are going to create.



# First, let's make something from pet registration to list display.

- We will create the same screen elements and data structure that we created in the Adalo meeting.
- Therefore, we will create the screen and database operations together in the Bubble meeting.
- We'll also use the default authentication methods provided in Bubble.
- First of all, let's create a screen to register pets and display them in the list!

## Prepare only the frame for the pet list page

- First of all, we will use the page that we used to check the operation as the pet list page.
- To do so, we will delete all the parts that Bubble generated automatically.
- Select one of the elements in the index page and press "Ctrl + A" to select all the elements.
- Then click on the Delete button and you should see the screen clear up.

# before element trimming

The screenshot shows a web application template from Bubble. At the top, there's a navigation bar with the '.bubble' logo, 'Bubble Forum', 'FAQs', and a 'SIGN UP OR LOGIN' button. Below the navigation, a callout box says 'Hit [Preview](#) above to explore'. The main heading is 'Bubble Boilerplate' in large, bold, dark blue text. Below it is a sub-headline 'Base template for your Bubble app'. A paragraph explains that the page contains everything needed to get started, mentioning Bubble Pro users can customize it. A 'GET STARTED' button is present. The page footer includes a 'Current V3.0.0' link. The bottom section features a 'UI Elements' heading with a sub-section for 'Buttons and Actions' (with a hand icon), 'Data Collection' (with a database icon), and 'Images and Graphic elements' (with a camera icon). Each section has a brief description.

.bubble

Bubble Forum FAQs SIGN UP OR LOGIN

Hit [Preview](#) above to explore

## Bubble Boilerplate

Base template for your Bubble app

This page has everything you need to get started. If you're already a Bubble Pro, you can use parts of this boilerplate, or delete the whole page.

[GET STARTED](#)

Current V3.0.0

### UI Elements

A collection of building blocks you can use to get started. Customize and experiment!

Buttons and Actions

Buttons are used for navigation, actions, triggers, and more. Set rollover states, active

Data Collection

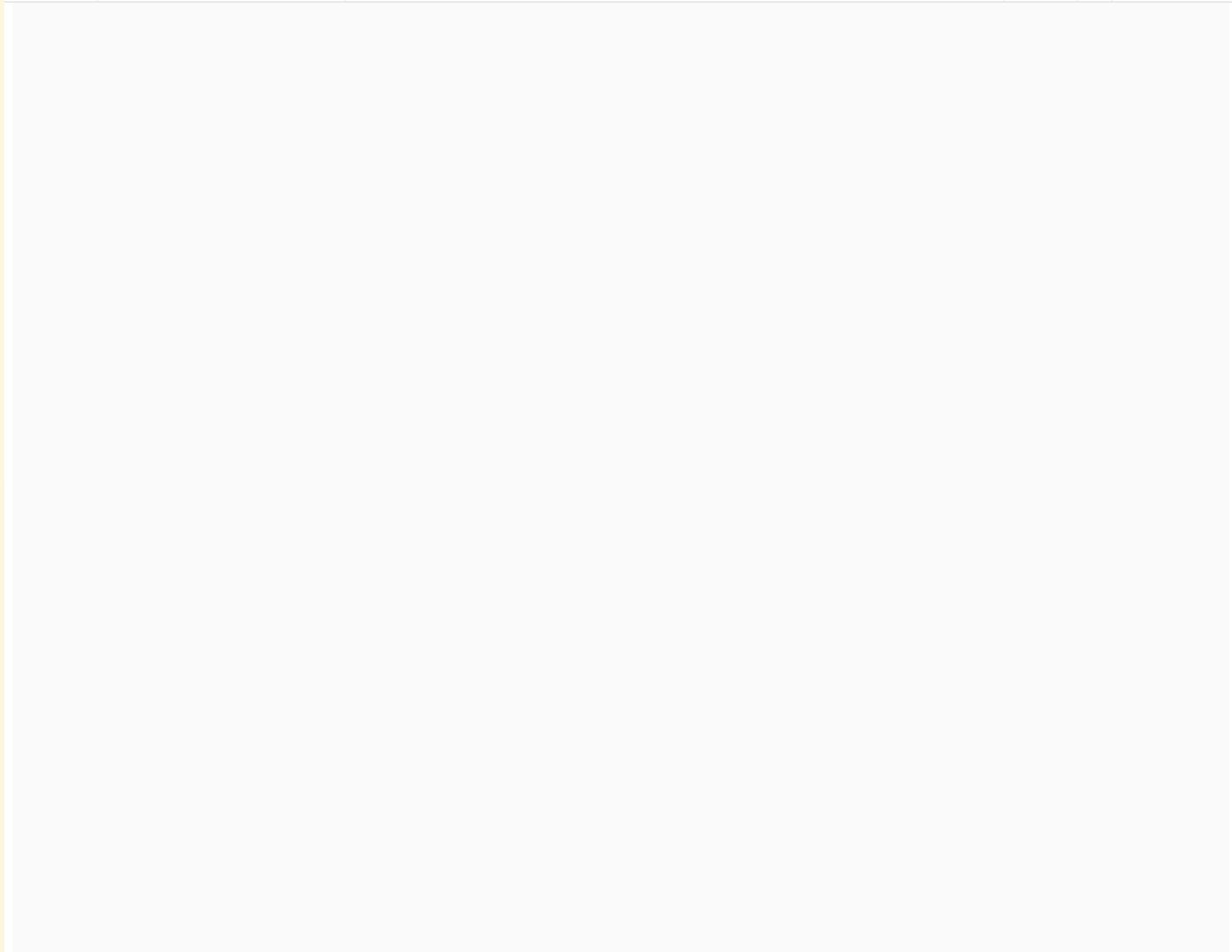
Form fields and inputs are essential parts of collecting data from your users. Here are a few

Images and Graphic elements

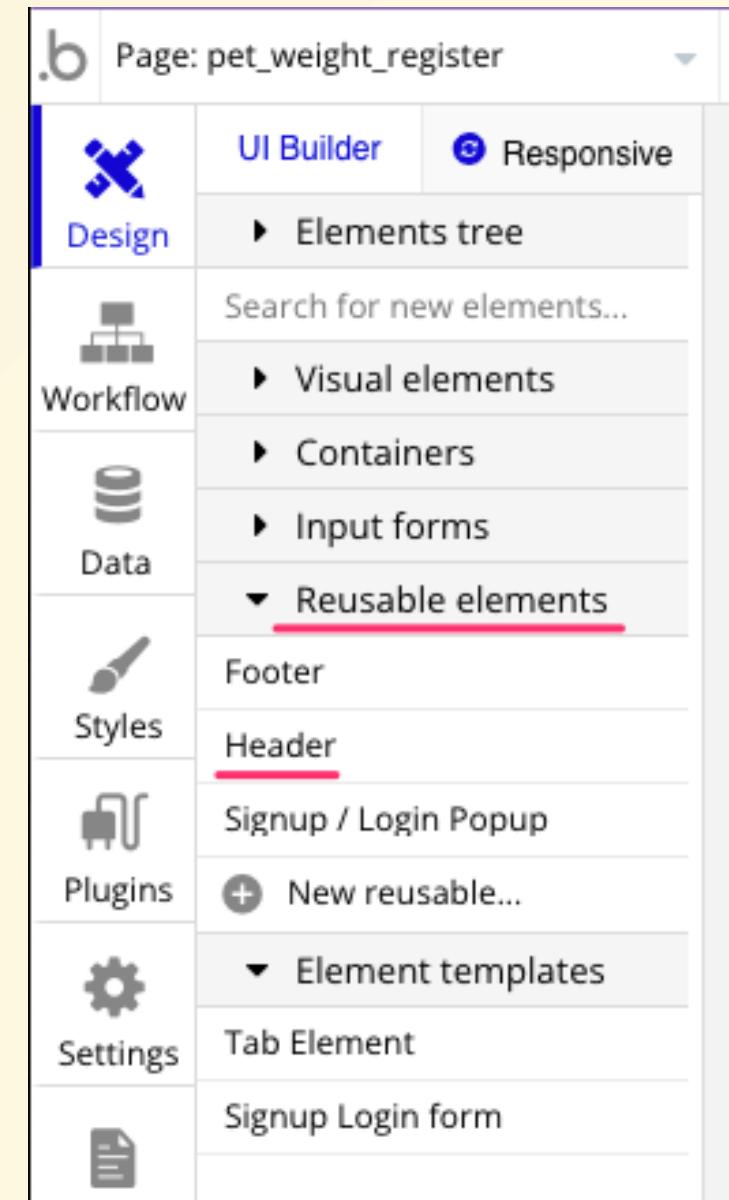
Bring your site to life with visual in-page elements and images. You can use static

37

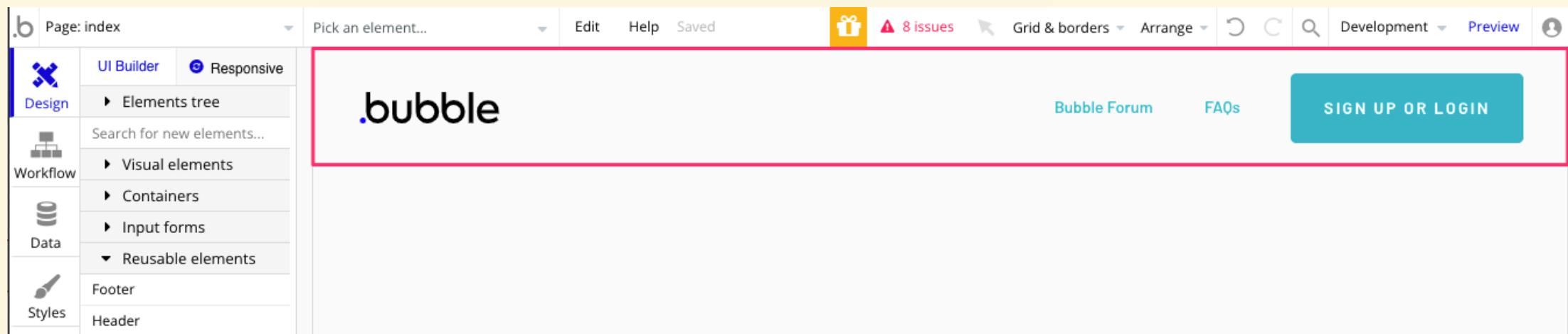
## after element trimming



- We'll use the header from the Bubble template
- Select the Design tab, scroll down to the "UI Builder" section, and in the **Reusable elements** section, you will see a Header, click on it



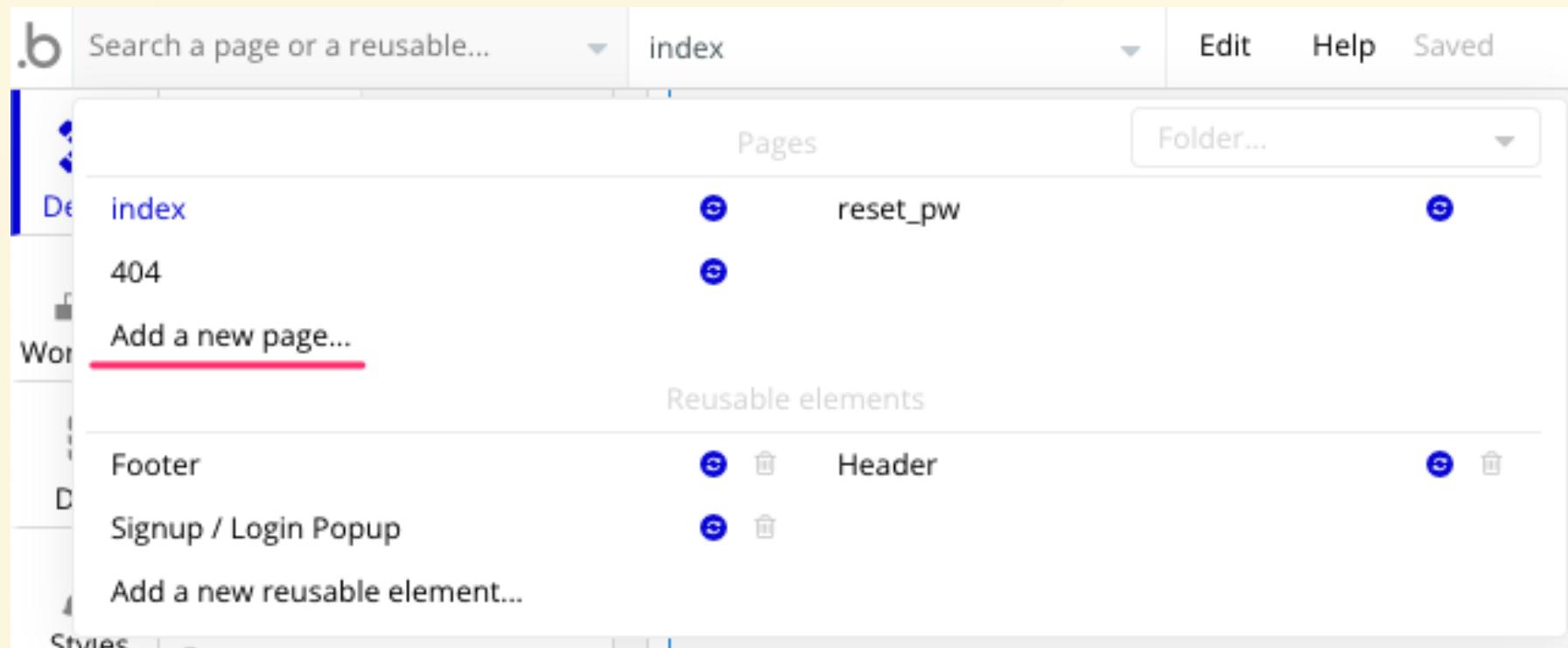
- Then, select (drag) a range at the top of the right side drawing area to add a common Header element to that area



- If the location cannot be placed exactly at the top, select Header and drag it to the top center of the screen.
  - Right-click on an element and select **Center horizontally** to center the horizontal position of the element
  - If you select **Center vertically**, the vertical position of the element will be centered.
  - Use this if you want to center the element in the future.
- The contents of the header will be changed later.

# Let's create the pet registration page.

- Click on the **Search a page or ~** part in the upper right corner of the screen, and a popup will appear.
- Click **Add a new page...** in the popup Click on `Add a new page...



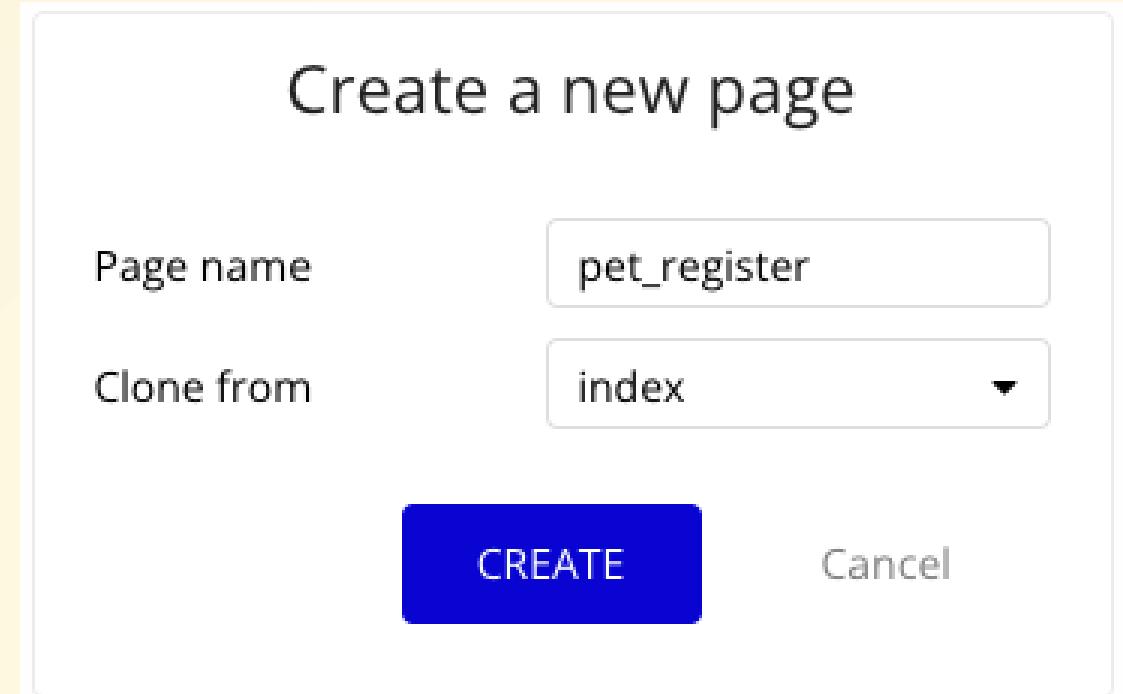
## By the way...

- Here you can see the list of "Pages" and "Common Components" in the application you are currently creating.
- In Adalo, all the screens were displayed on one canvas, but in Bubble, you will be manipulating one screen on one canvas.
- In Bubble, you will be manipulating a single screen on a single canvas, so keep in mind that you will need to change the movement between screens from here 

- Click

Add a new page... Click  
Add a new page...` to  
display a popup for the new  
page.

- The Page name will be the name of the screen, so enter only alphanumeric characters, no spaces.
  - This time, let's use pet\_register .



- The **Clone from** function allows you to create a copy of a similar screen by selecting the screen to be copied from.

Create a new page

Page name

Clone from  ▾

CREATE Cancel

- For example, the elements of the registration screen and the editing screen are almost the same, so if you want to make the editing screen after making the registration screen, selecting the registration screen as "Clone from" will reduce the development time.

Create a new page

Page name	<input type="text" value="pet_register"/>
Clone from	<input style="width: 150px;" type="text" value="index"/> ▼

CREATE Cancel

- This time, let's select and copy the index for which we just prepared the header.
- After entering the necessary information, click the "CREATE" button.

Create a new page

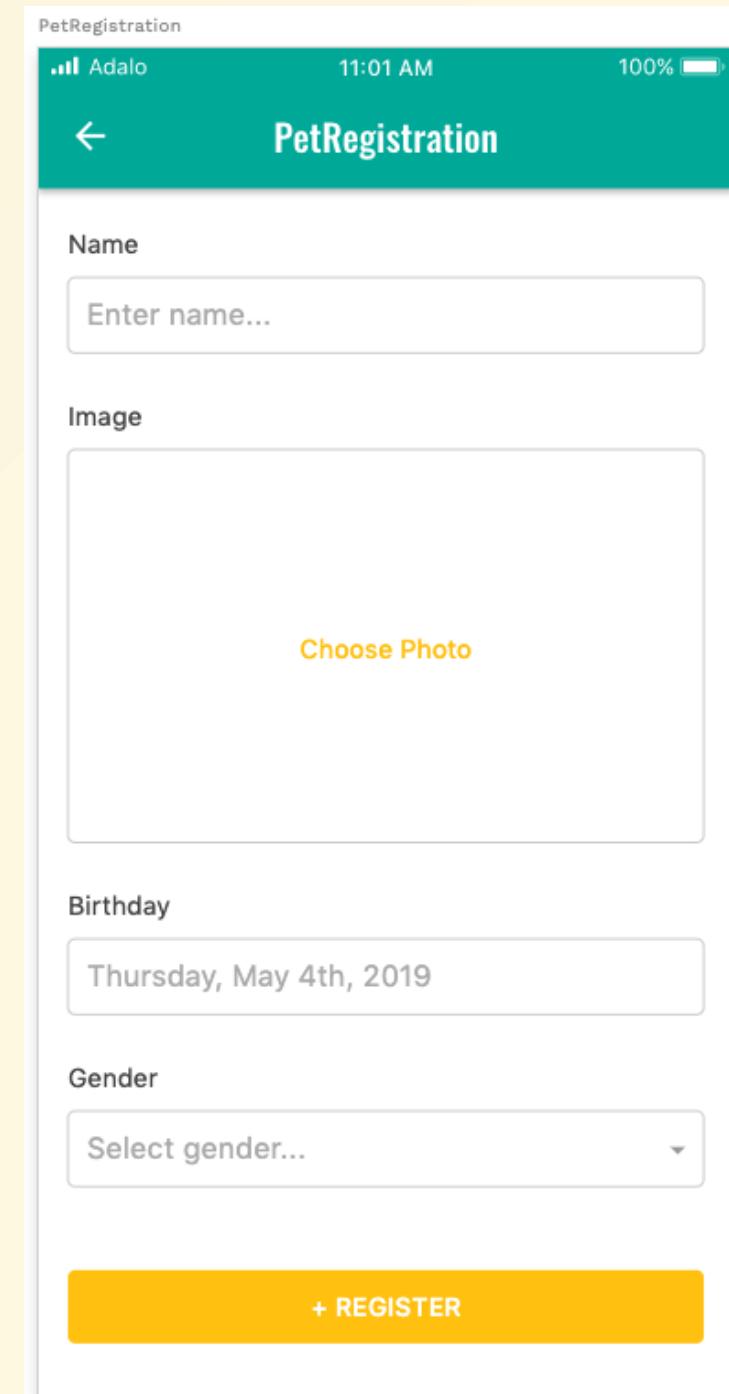
Page name

Clone from  ▾

CREATE Cancel

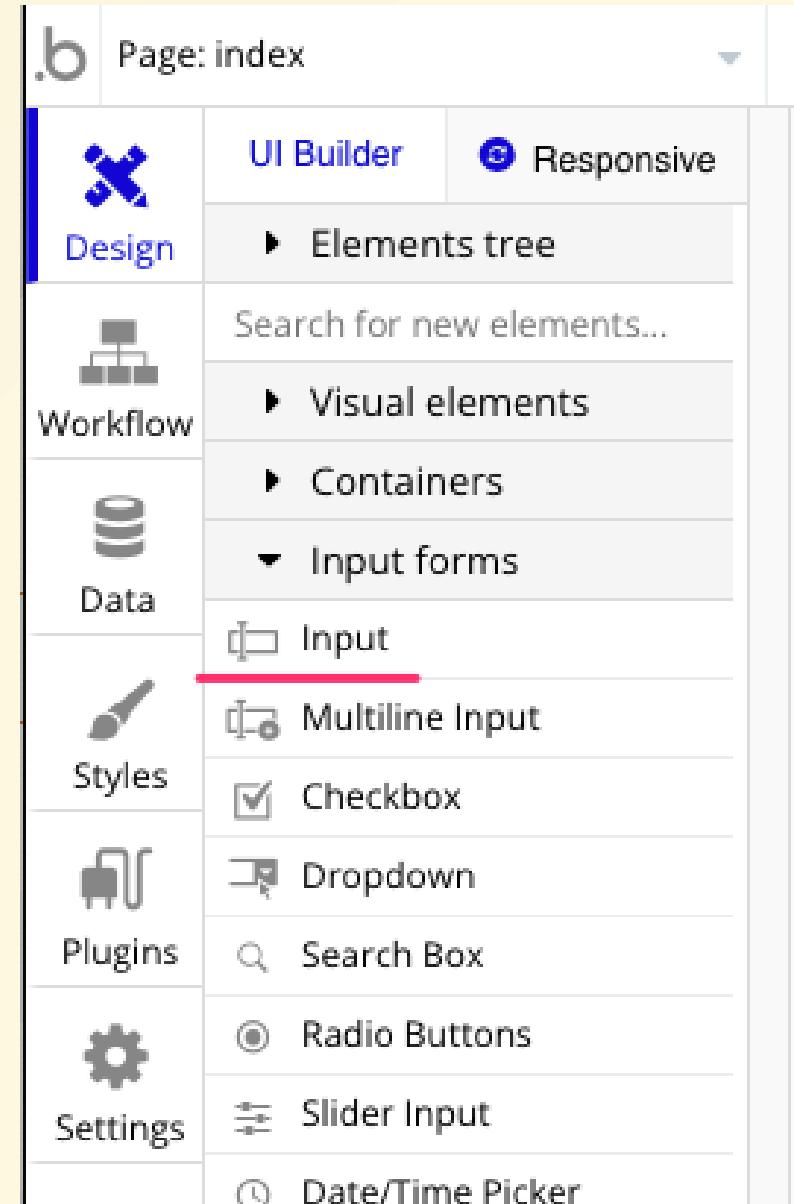
# Assemble the Pet Registration Screen

- Let's make a screen based on the image of the pet registration screen created by Adalo.

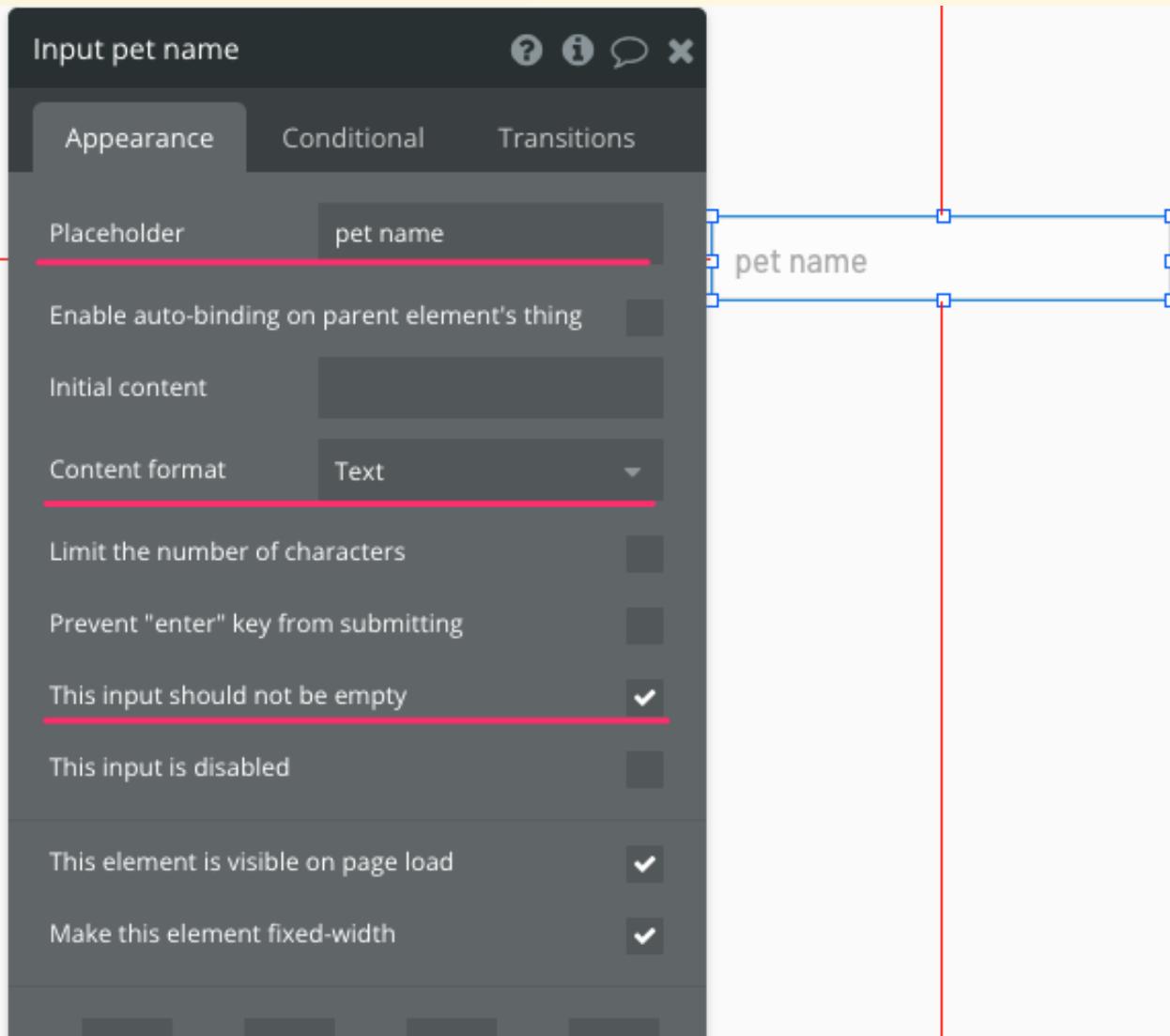


## Pet's name

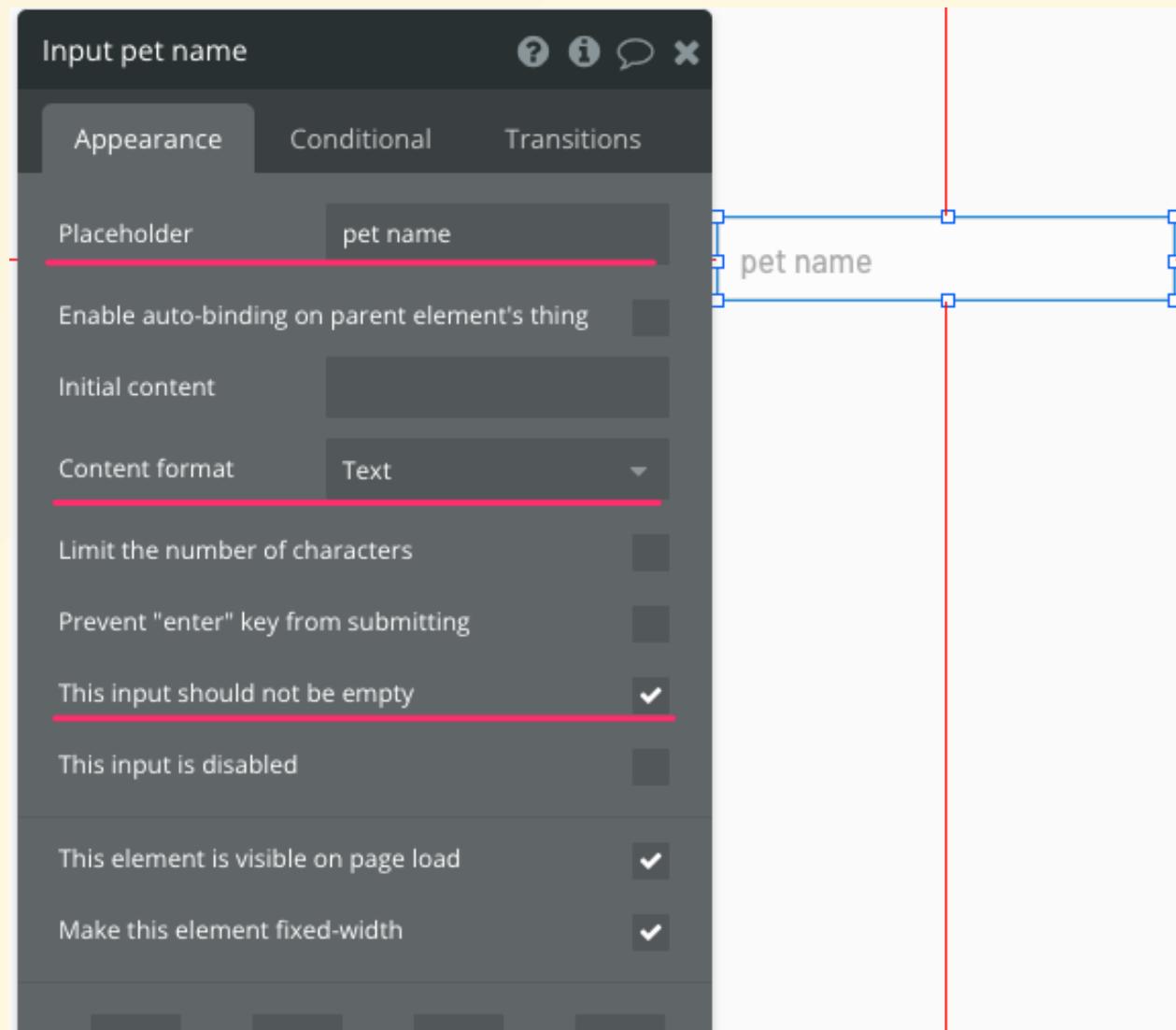
- First, we'll place a text box for the pet's name.
- Go to "UI Builder" in the left panel and select **Input** in the **Input forms** section.
- Then, drag the text box on the right panel where you want to place it, and the input box will appear



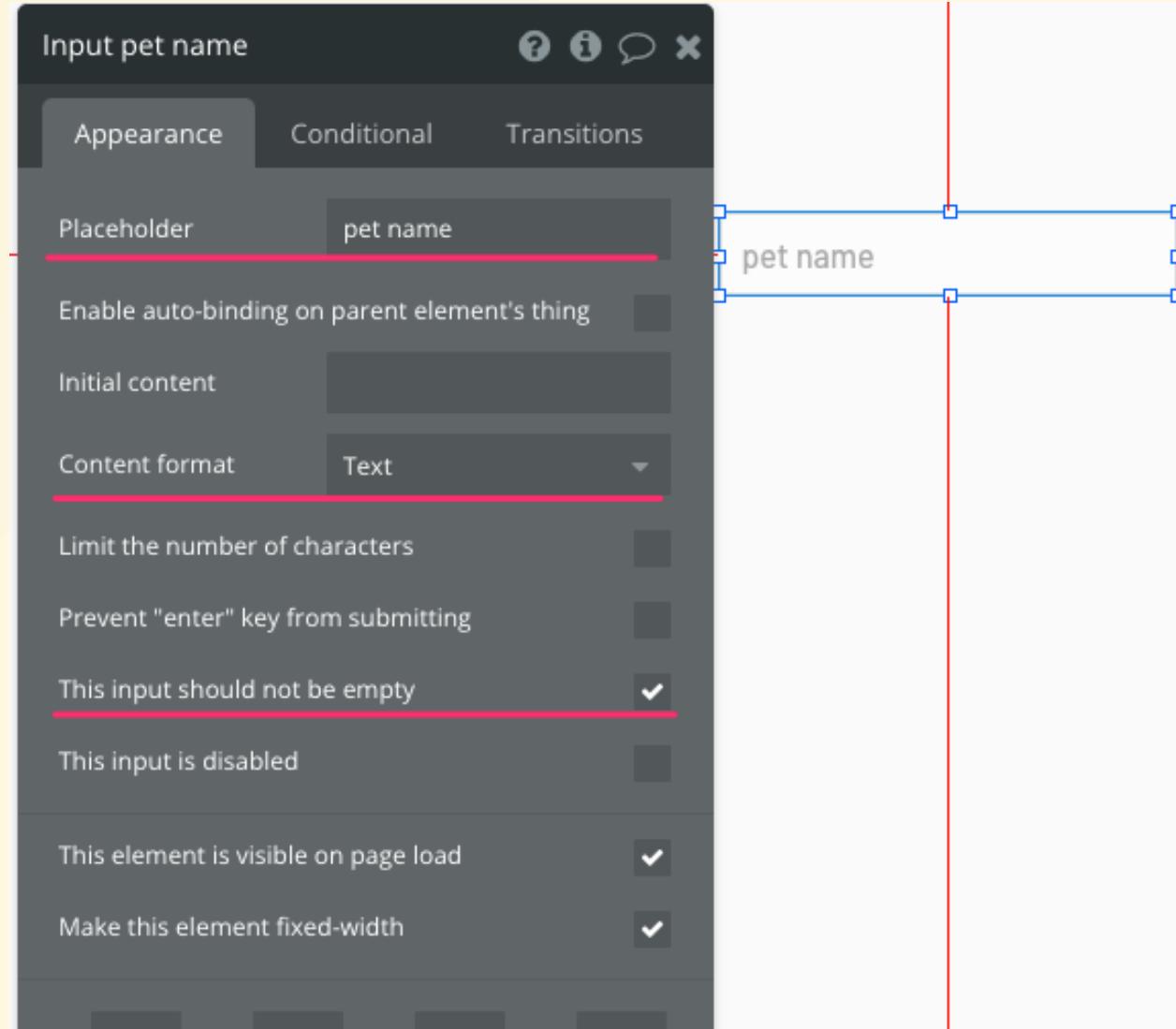
- Double-clicking on an element will bring up a dialog where you can set various information about the element, and set the necessary items.
- The **Placeholder** is the auxiliary text that will be displayed when the text box is not filled in.
  - In this case, we will use "pet name".



- The **Content format** allows you to specify the format of the value that can be entered into the text box.
  - In this case, we'll leave it as "Text" since we'll be entering a string.

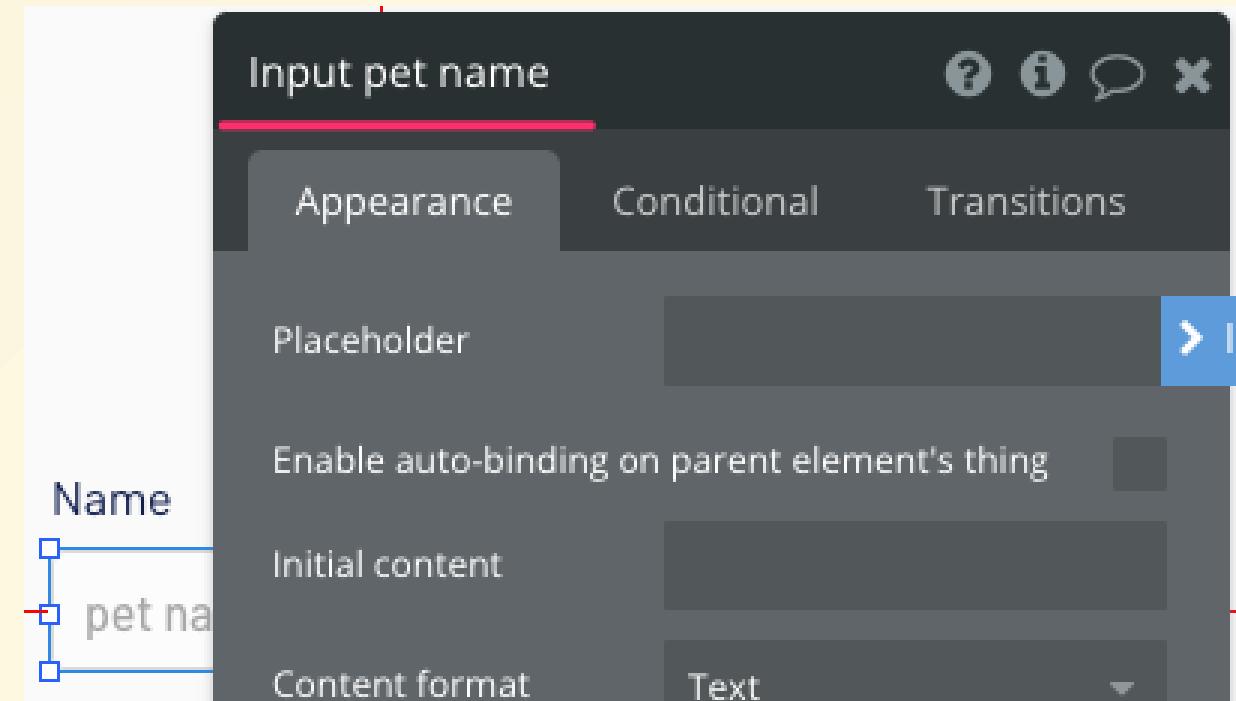


- The **This input should not be empty** option allows you to specify whether or not the input should be required.
  - In this case, we want to make it mandatory, so we'll check the box.



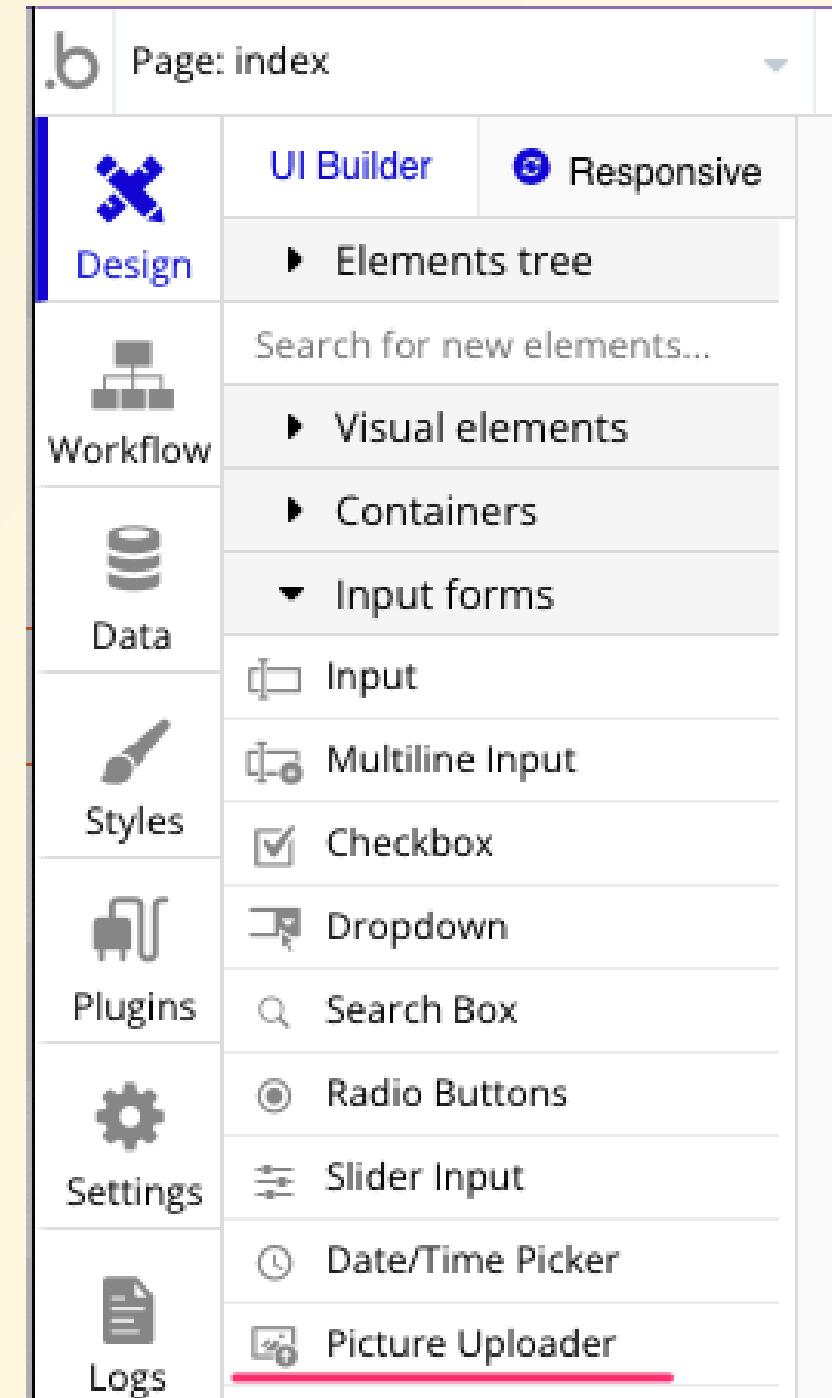
- Once you have set up your settings, you are good to go.
  - You can also specify other details, but I won't explain them here.
  - If you are interested, please take a look at 

- There is only one more item to set
- If you click on the top of the dialog, you can give a name to the element in this text box
- This will come in handy later when defining workflows, so let's use **Input pet name** here

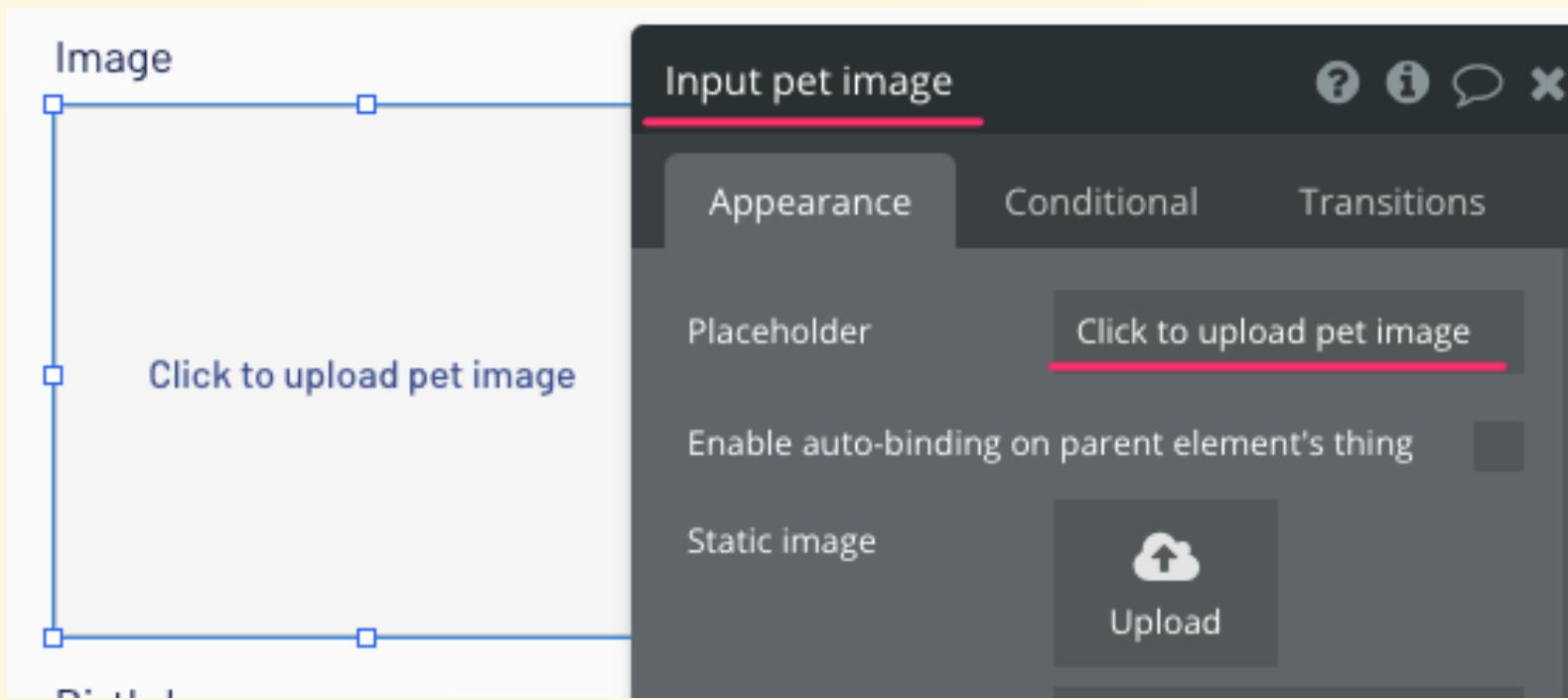


# Pet Image

- The next step is to place the elements for uploading the pet image
- Select **Picture Uploader** from the left panel and drag it on the right panel to place the element



- Double-click on this element as well to set the element's information
  - This time, enter "Click to upload pet image" in the Placeholder field.
  - Give the element a name of Input pet image .



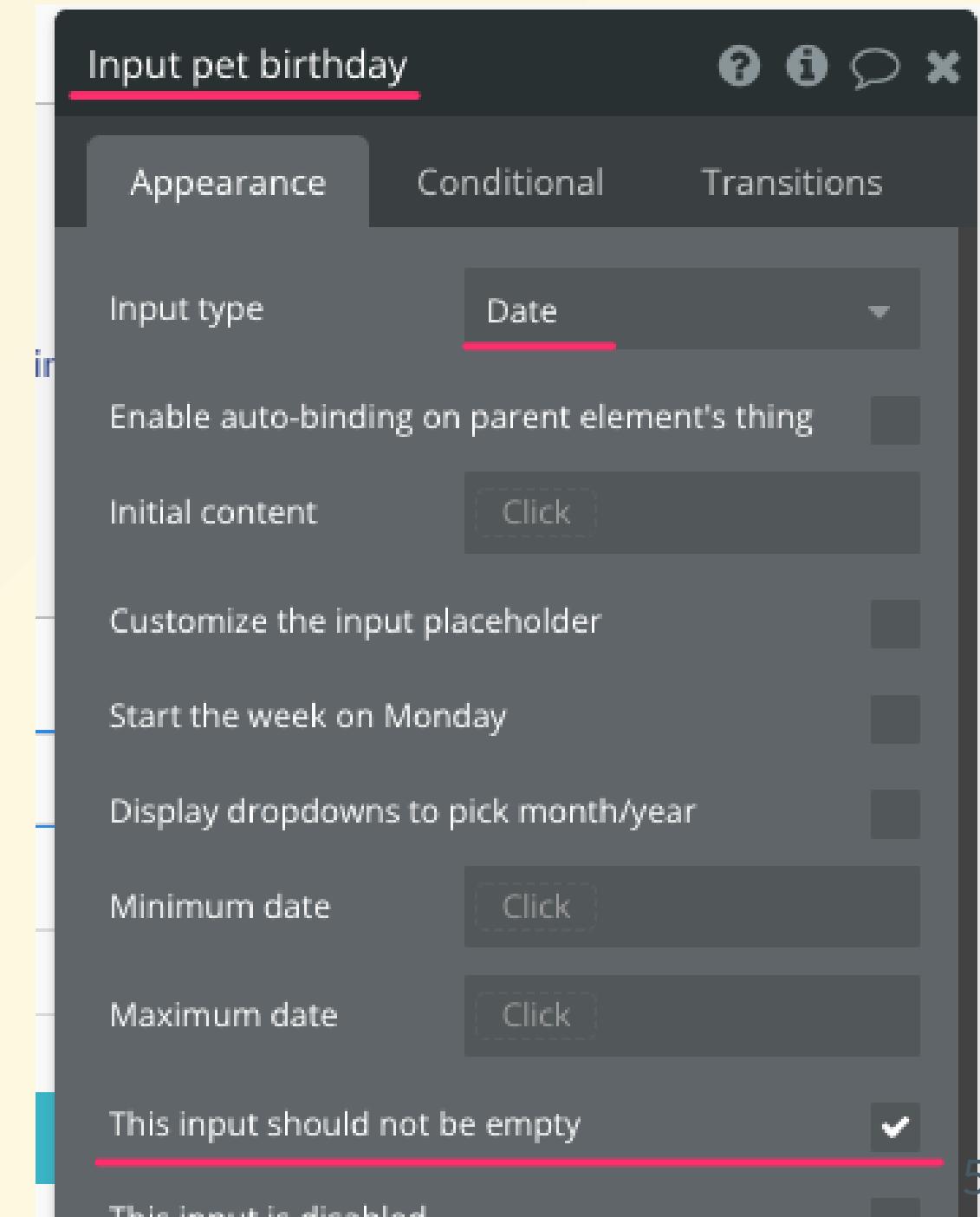
## Pet's birthday

- Next, we will place the elements for entering the pet's birthday.
- Select the **Date/Time Picker** from the left panel and drag it on the right panel to place the element

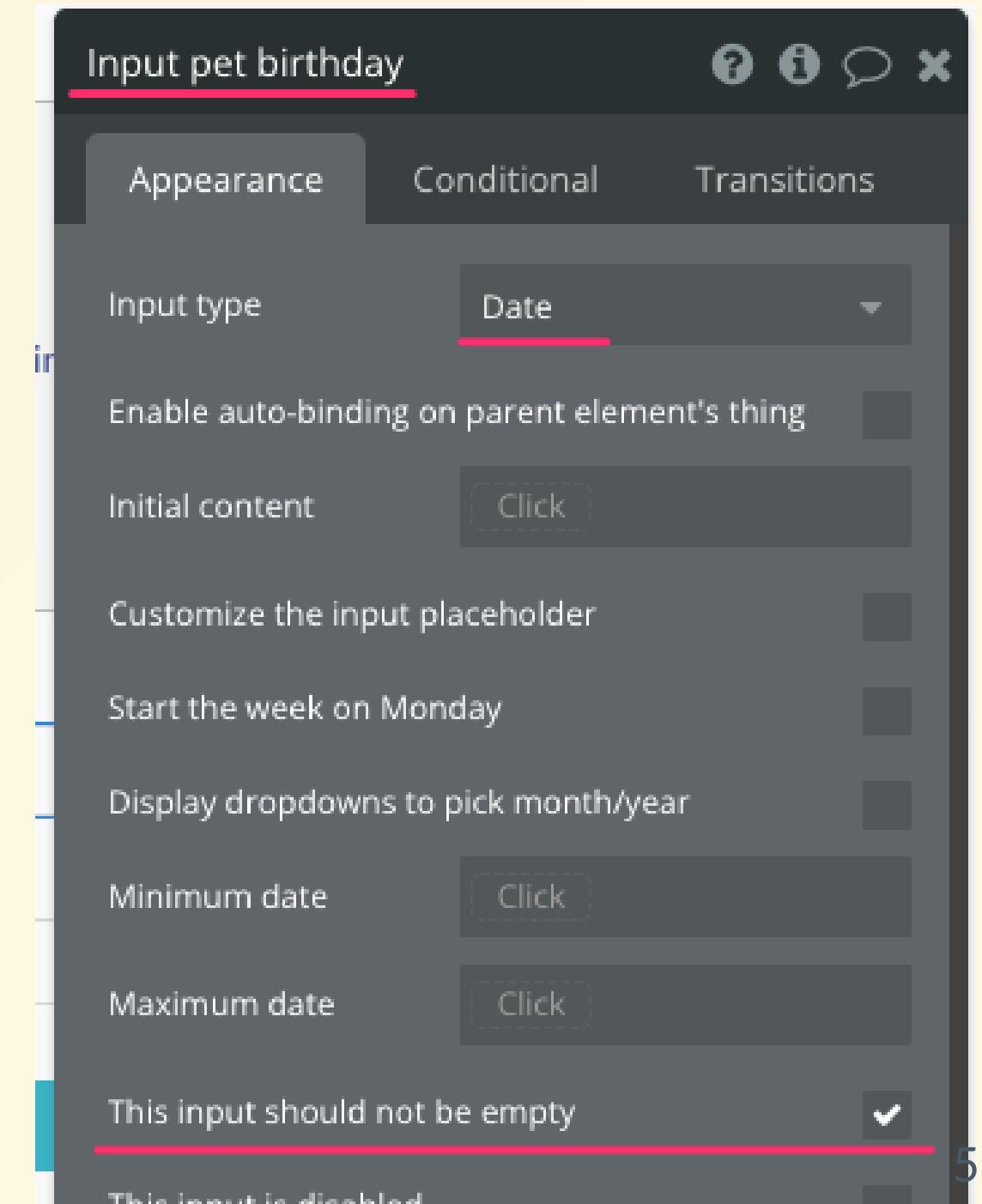
The screenshot shows the UI Builder interface with the following details:

- Page:** pet\_register
- UI Builder:** Responsive
- Design Panel:** Elements tree, Search for new elements...
- Workflow Panel:** Visual elements, Containers
- Data Panel:** Input forms
  - Input
  - Multiline Input
  - Checkbox
  - Dropdown
  - Search Box
  - Radio Buttons
  - Slider Input
  - Date/Time Picker** (highlighted with a red underline)
  - Picture Uploader
- Styles Panel:** Styles
- Plugins Panel:** Plugins
- Settings Panel:** Settings
- Logs Panel:** Logs

- Double click on the element to set the element details.
  - Select Date for the **Input type** to allow only date input.
    - Selecting **Date & Time** will allow you to enter both date and time.
    - In this case, we will use Date.



- Check the **This input should not be empty** box to make it mandatory.
- Specify **Input pet birthday** as the name of the element



## Pet's gender

- Finally, we will place an element to select the gender of the pet
- Select **Dropdown** from the left panel and drag it on the right panel to place the element

.b Page: pet\_register

UI Builder  Responsive

Design Elements tree

Workflow Search for new elements...

Data Visual elements

Containers

Input forms

Input

Multiline Input

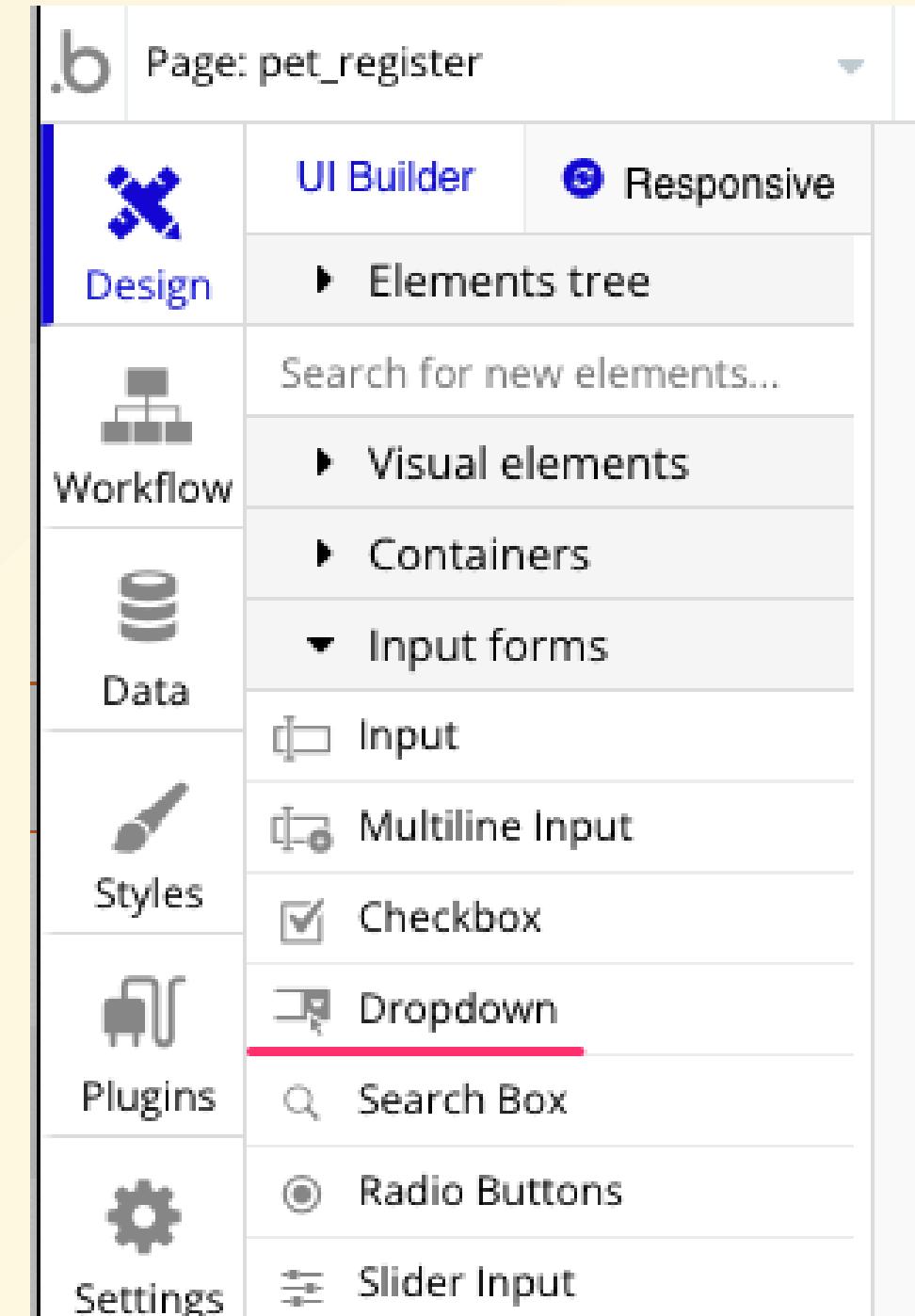
Checkbox

**Dropdown**

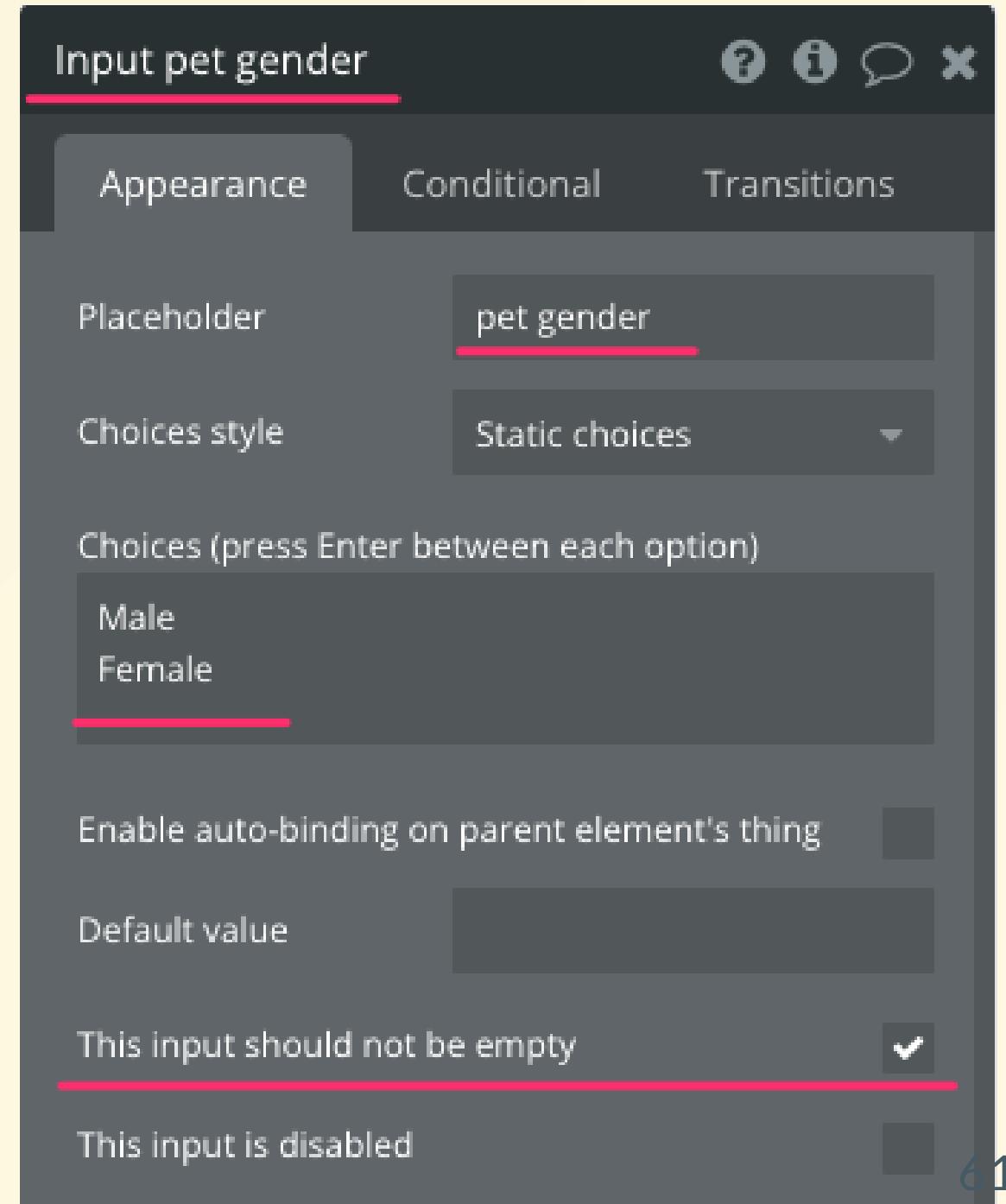
Search Box

Radio Buttons

Slider Input



- Double-click on the element to set the element details.
  - Type "pet gender" in the Placeholder field.
  - Type "Male" in the Choices field, hit Enter to start a new line, and type "Female" in the next line



- Check the **This input should not be empty** box to make it mandatory.
- Specify **Input pet gender** as the name of the element

Input pet gender

Appearance Conditional Transitions

Placeholder pet gender

Choices style Static choices ▾

Choices (press Enter between each option)

Male

Female

Enable auto-binding on parent element's thing

Default value

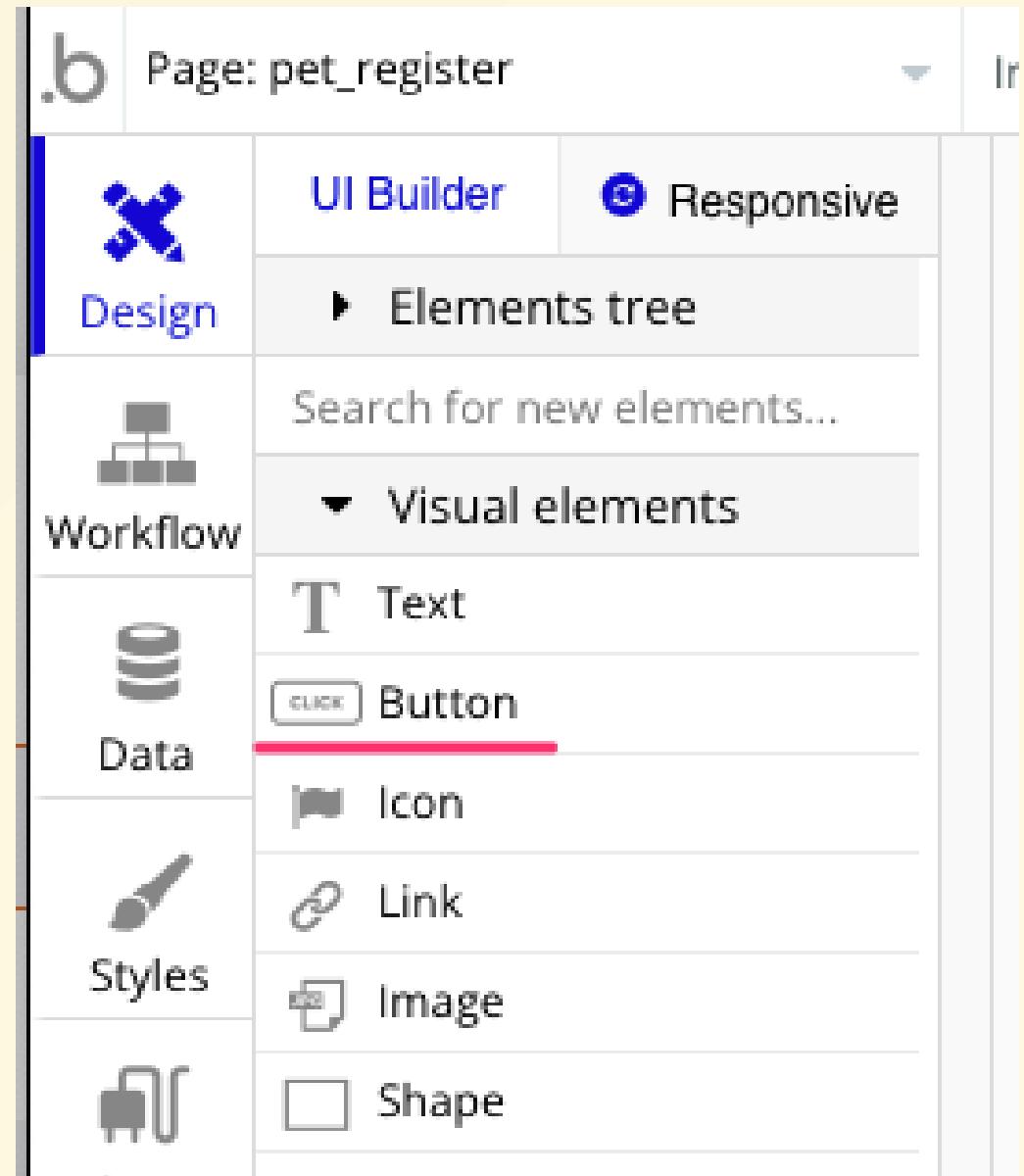
This input should not be empty

This input is disabled

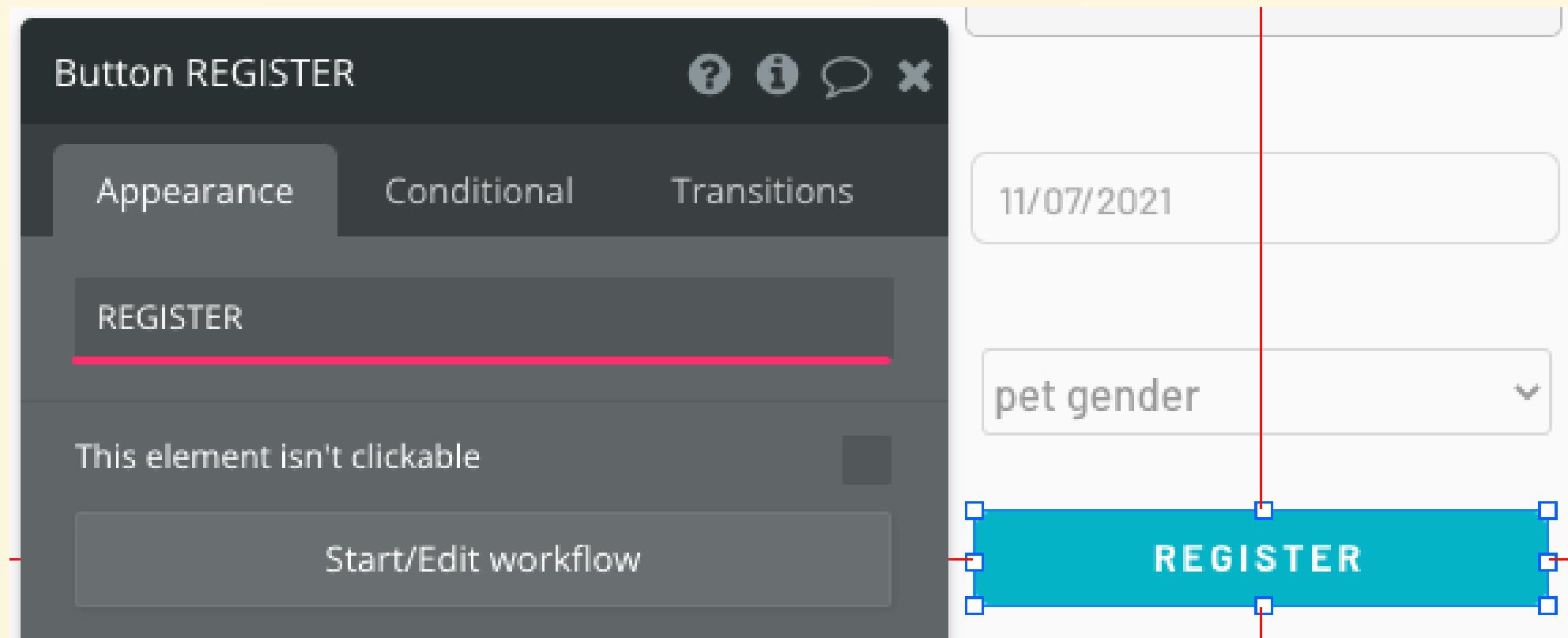
The screenshot shows a UI configuration interface for an input field. The title bar says 'Input pet gender'. There are tabs for Appearance, Conditional, and Transitions, with Appearance selected. Under Appearance, there's a Placeholder field containing 'pet gender' with a red underline. A Choices style dropdown is set to 'Static choices'. Below it is a Choices input field with 'Male' and 'Female' separated by a red underline. A checkbox labeled 'Enable auto-binding on parent element's thing' is checked. The 'Default value' field is empty. At the bottom, a red underline highlights the error message 'This input should not be empty', which is checked. Another message 'This input is disabled' is at the bottom right.

# Registration Button

- Now that you have entered your pet's information, you can place the registration button.
- Select **Button** in the left panel under **Visual elements** and drag it in the right panel to place the element.

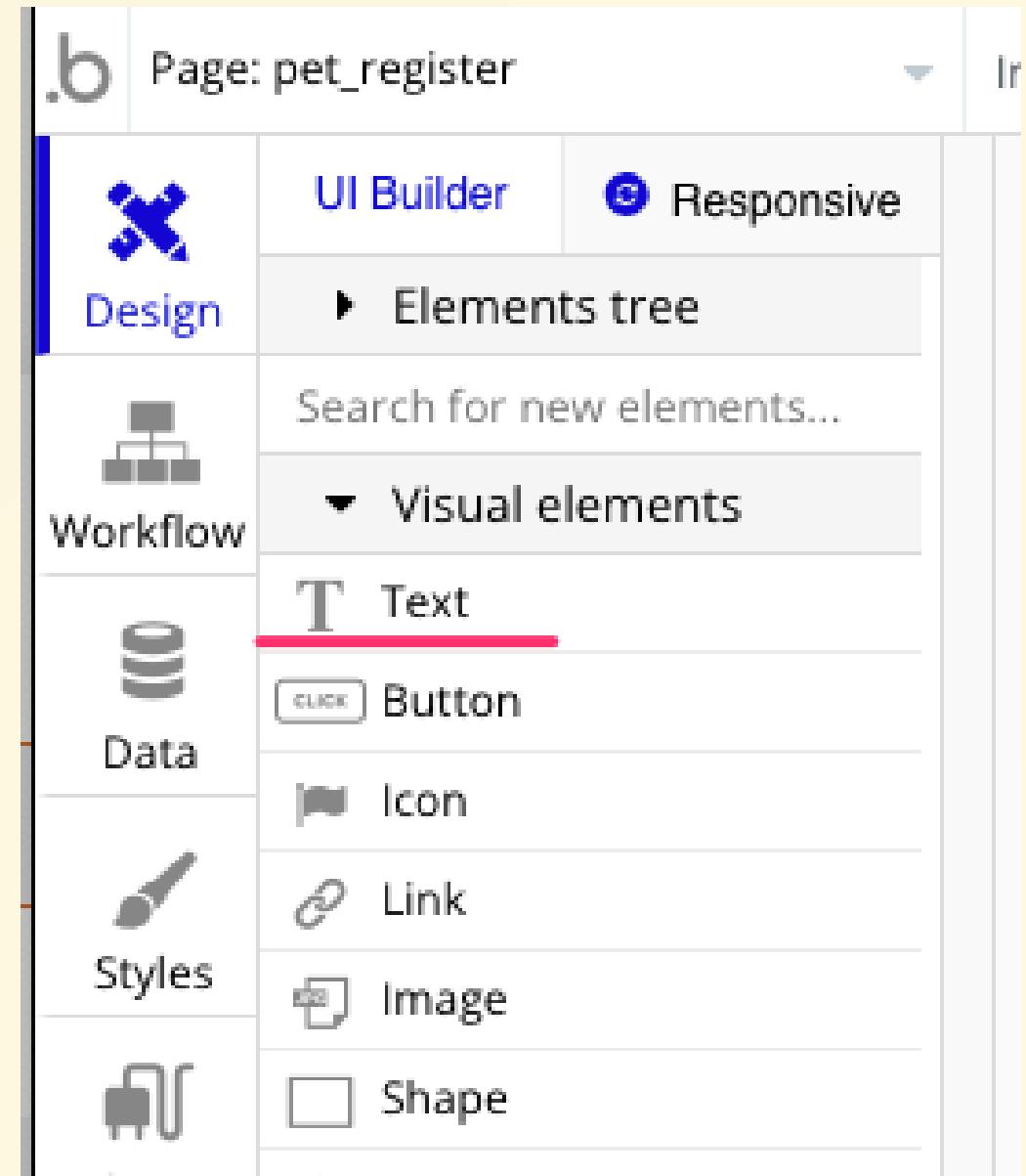


- Double-click on the element to set the details of the element.
  - At the top of the Appearance tab, click the ". .edit me..." at the top of the Appearance tab. and type in "REGISTER"

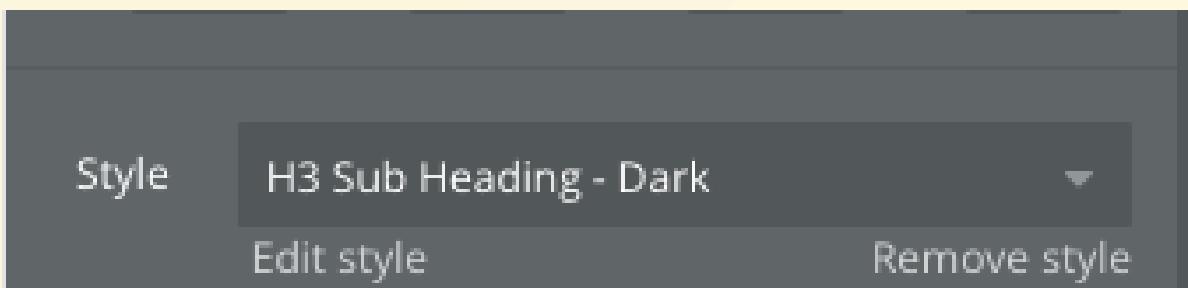


# Let's label each input element!

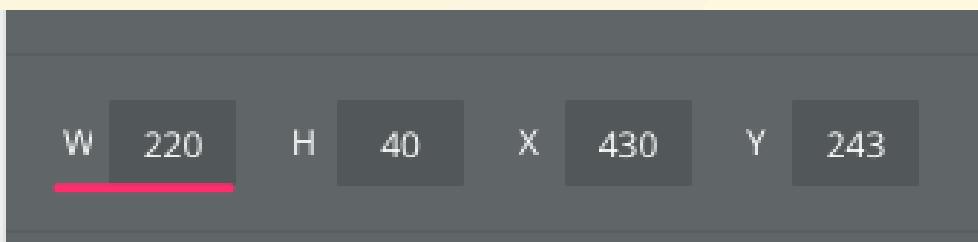
- Let's prepare a label in the upper left corner of each element, as shown in the Adalo screen
- Select **Text** in the left panel under **Visual elements** and drag it in the right panel to place the element.



- When placing the element, make sure that the height is at least 30px.
- Otherwise, the content of the element will not be displayed correctly!
- Double-click on the element to set the details of the element.
  - Double-click on the element to set its details. ".edit me..." at the top of the Appearance tab. and type "Name".
  - Select **H3 Sub Heading - Dark** for **Style** in the dialog interruption

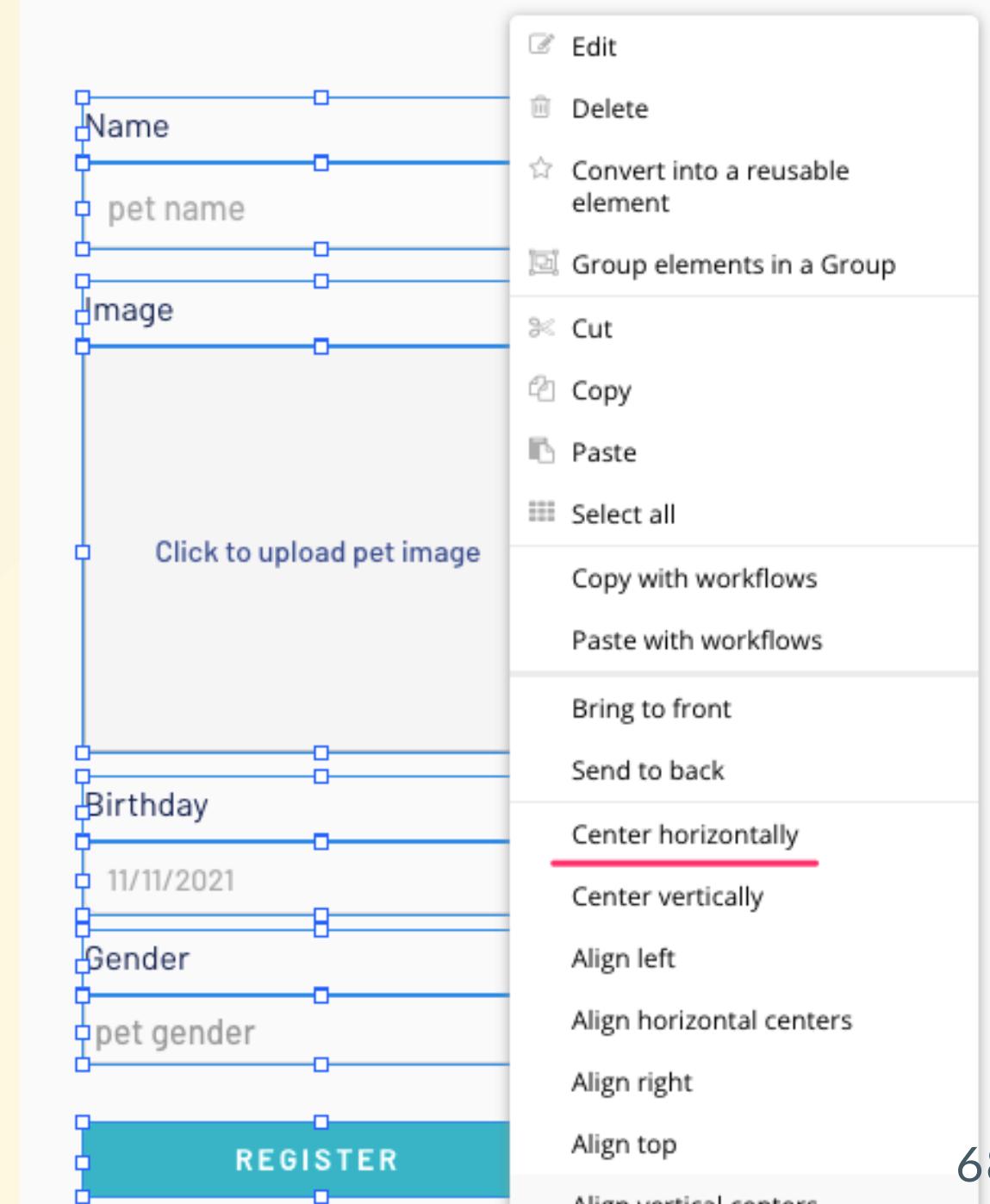


- Let's add the other labels in the same way
- You can follow the same procedure, but you can save time by copying and using the Name label that you just created.
  - Copy (Ctrl + C), paste (Ctrl + P), and rewrite the label content only.
- Finally, set the width of all the elements to 220px.
  - From the dialog that appears when you double-click on each element, change the **W** (Width) value to 220.
  - This is to make it look better when displayed on the screen.



# Finally, organize the elements and preview them

- Select all the elements placed in the right panel (except Header) by dragging them, right-click and select **Center horizontally** to center the input form horizontally.



- Once you've done all this, click on **Preview** in the upper right corner of the screen to show a preview and see it in action!
- At this point, make sure you are logged in!
  - Make sure you are logged in at this time! This is because this information is required for the list view and linkage later on!



LOG OUT

- The button in the upper right corner of the header is
- Are all the elements visible?
- Can you enter values in all the input forms?
- Does it display when an image is specified?

The image displays two side-by-side screenshots of a mobile application interface for registering a pet. Both screenshots show a white form on a black background.

**Left Screenshot (Initial State):**

- Name: pet name
- Image: Click to upload pet image
- Birthday: 11/12/2021
- Gender: pet gender
- REGISTER button (blue)

**Right Screenshot (Filled State):**

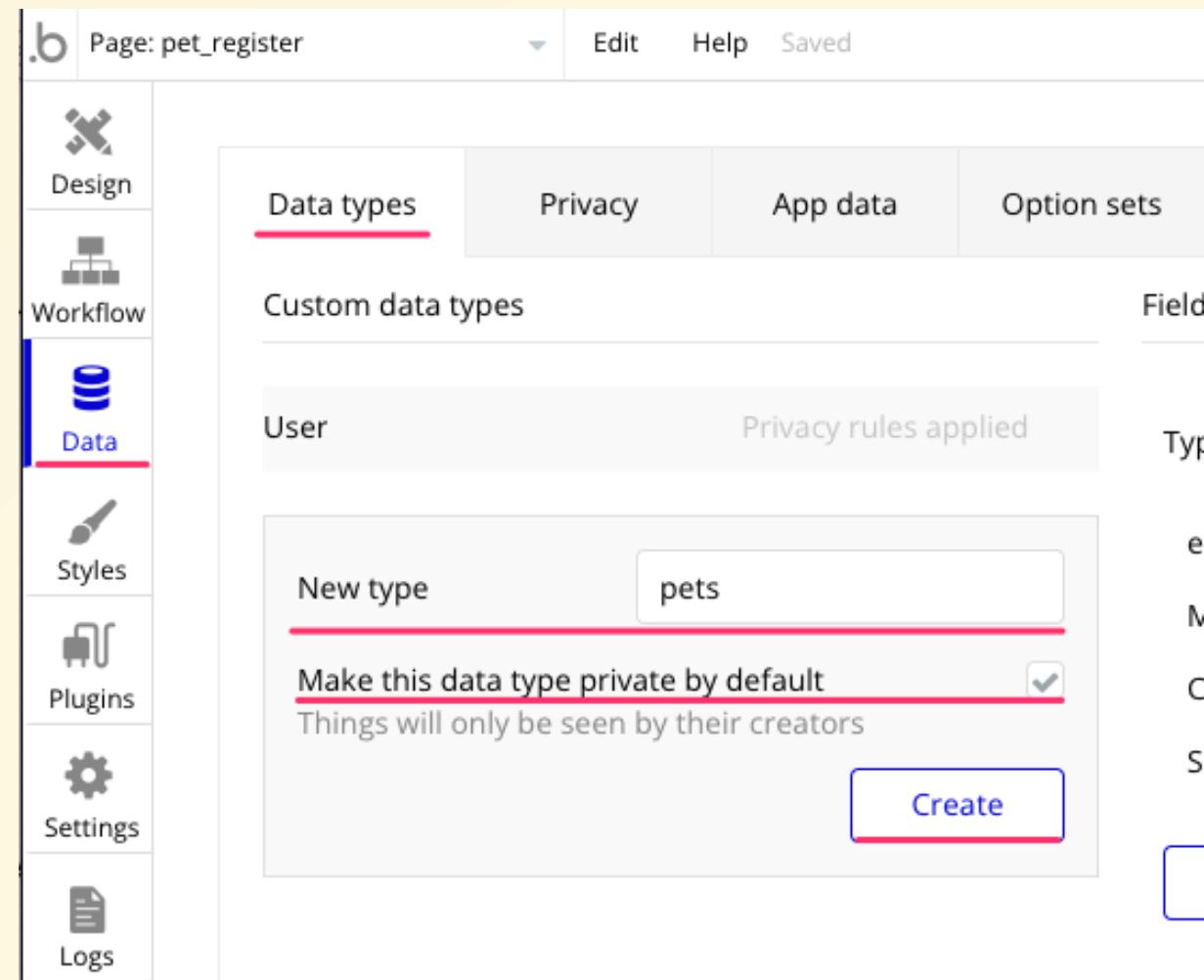
- Name: ペットの名前 (Pet's Name)
- Image: A small thumbnail image of a Shiba Inu dog standing on a bed of fallen leaves.
- Birthday: 10/20/2020
- Gender: Male
- REGISTER button (blue)

# Let's connect it to the database.

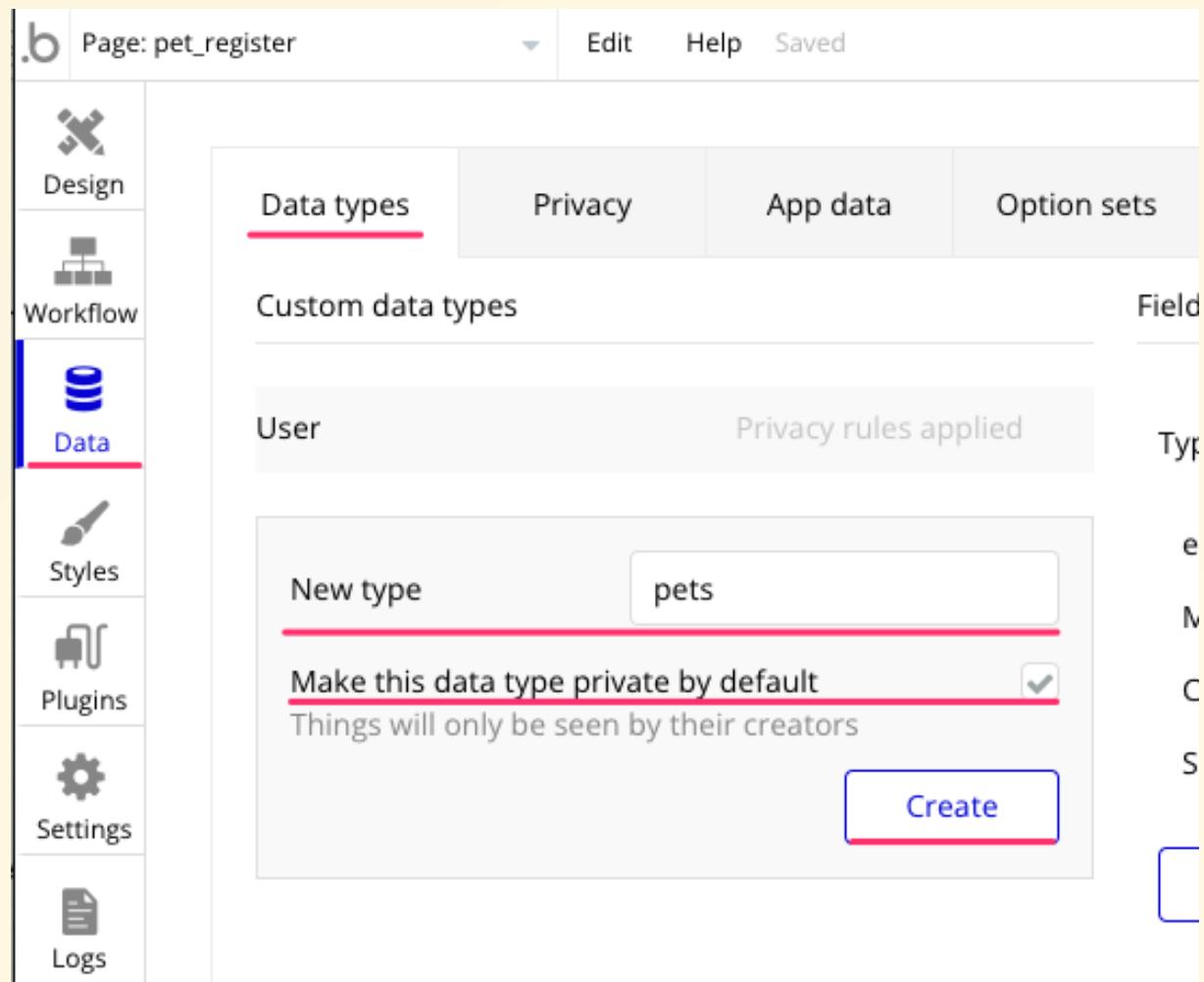
- Now that the screen layout is ready, let's connect it to the database.

# First of all, we need to prepare a box to store the pet information.

- Select Data from the tabs on the left panel.
- Under the Data types tab, type "Pets" in the text box labeled **New type** in the Custom data types section.



- Then, check the Make this data type private by default checkbox and click the Create button !



## By the way...

- Adalo and Bubble have different names for some of the elements of the database
- Type: Defines the type (box) of data.
  - In Adalo, it is called Collection
- Field: An element to represent the type of data
  - Property in Adalo.

- Let's add an element (field) to the type of Pets we created.
- Select **Pets** from **Custome data types**.
- Click on **Create a new field** under **Fields for type pets** on the right

Page: pet\_register

Edit Help Saved

Design

Workflow

Data

Styles

Plugins

Settings

Logs

Custom data types (show deleted types)

Pets

PetWeightLogs

User

New type

Make this data type private by default

Things will only be seen by their creators

Create

Type name: Pets

Birthday: date

Gender: text

Image: image

Name: text

Creator: User

Modified Date: date

Created Date: date

Slug: text

default

Upload

Create a new field

- Enter the name of the element in the **Field name** field.

### Create a new field

Field name

Field type

This field is a list (multiple entries)

**CREATE** Cancel

- In the **Field type** field select the type of element (text, number, date, etc.)

Create a new field

Field name

Field type

This field is a list (multiple entries)

**CREATE** Cancel

- Adalo allows you to specify the relationship between types via Relationship
- In Bubble, the **Field type** option also shows the registered types, so you can select the type you want to associate with it, just like in Adalo

### Create a new field

Field name

Field type

This field is a list (multiple entries)

**CREATE** Cancel

# Now let's add the necessary fields to Pets type

- Name: text
- Image: image
- Birthday: date
- Gender: text

Fields for type Pets

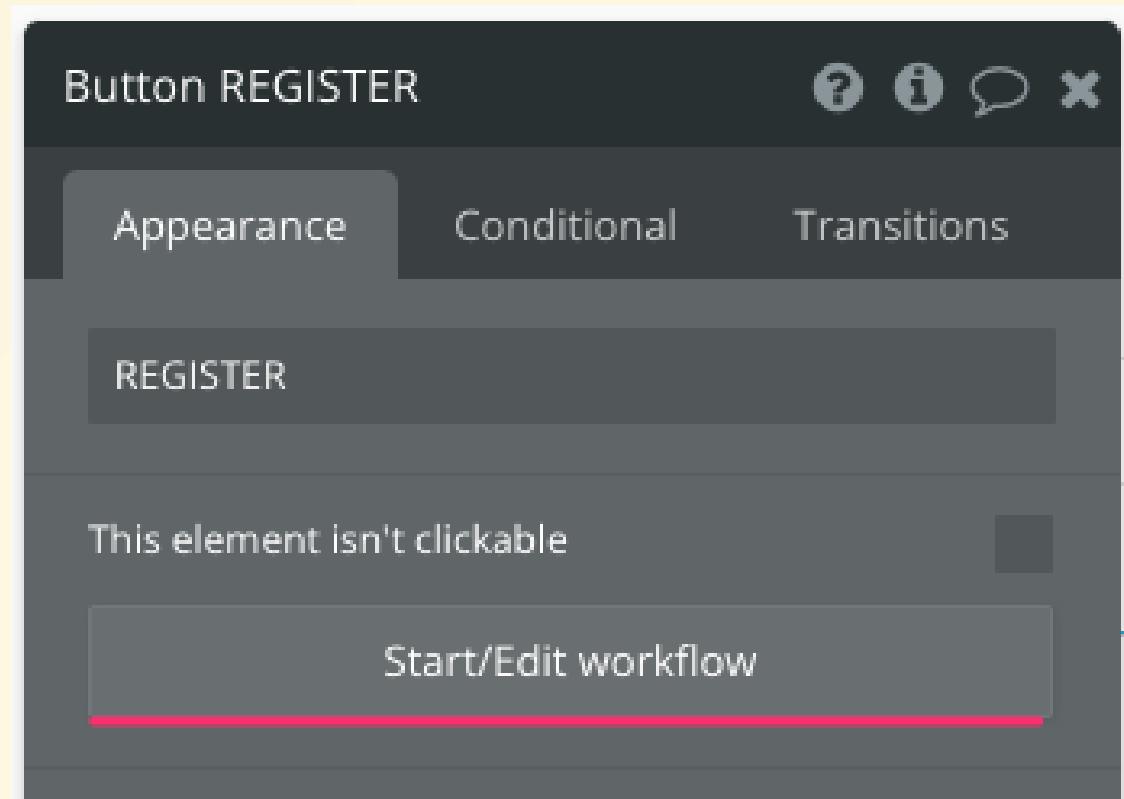
Type name	Pets	
Birthday	date	default
Gender	text	default
Image	image	default Upload
Name	text	default
Creator	User	Built-in field
Modified Date	date	Built-in field
Created Date	date	Built-in field
Slug	text	Built-in field

[Create a new field](#)

# Make it possible to register a pet after clicking the Register button.

- Now that the box for storing the pet information is ready, it's time to add the movement to save the information to the database when the registration button is clicked.
- In Bubble, all the movement settings are done in the Workflow tab.
- We can start with the Workflow tab, but for this example, we'll start with the button that will be the source of the movement.

- Select the Design tab from the left panel
- Double-click the "REGISTER" button on the Pette Registration screen in the right panel
- Click the Start/Edit workflow button in the "Appearance" tab



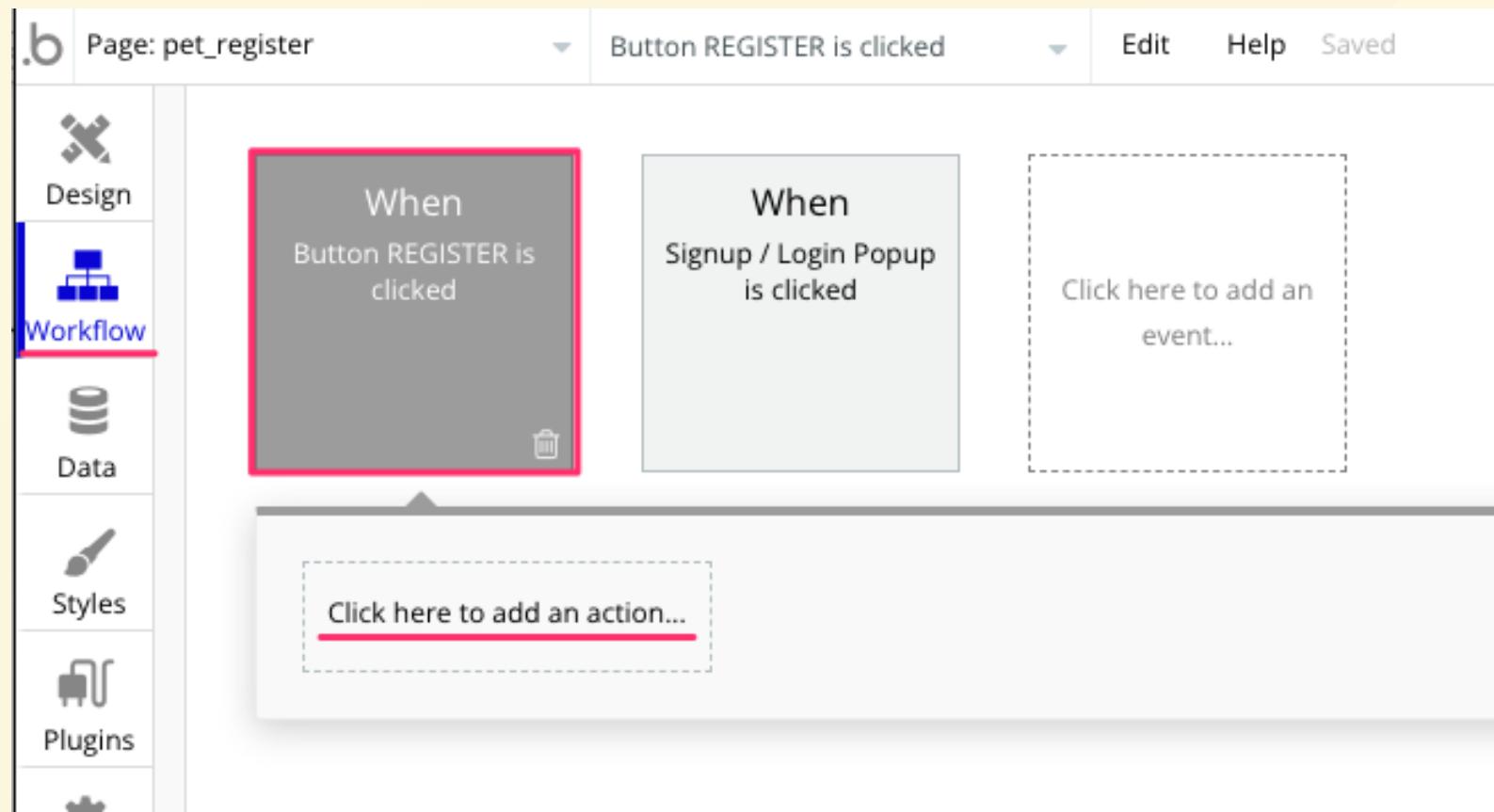
- This will switch to the Workflow tab, and in the When section, you should see `Button REGISTER is clicked
  - This is literally the workflow definition for when the "REGISTER" button is clicked

The screenshot shows a user interface for managing workflows. On the left, a vertical sidebar lists five tabs: Design, Workflow (which is currently selected and highlighted in blue), Data, Styles, and Plugins. The main content area has a header bar with the page title 'pet\_register', a dropdown menu, and buttons for Edit, Help, and Saved. The main workspace contains three main components:

- A large gray box with a red border containing the text 'When Button REGISTER is clicked'. A small trash can icon is located at the bottom right of this box.
- A smaller white box containing the text 'When Signup / Login Popup is clicked'.
- A dashed-line box with the placeholder text 'Click here to add an event...'.

At the bottom of the workspace, there is another dashed-line box with the placeholder text 'Click here to add an action...'.

- So, select **Click here to add an action...** below it. and configure the behavior when the button is pressed.



- Click here to add an action... Select 'Click here to add an action...' to display a popup window where you can specify various actions.
- Select **Data(Things)** for database operations (actions) like this one.

The screenshot shows the Bubble app's workflow editor for a page named 'pet\_register'. The 'Workflow' tab is active. The timeline contains two events: 'When Button REGISTER is clicked' and 'When Signup / Login Popup is clicked'. A modal window is open under the second event, titled 'Click here to add an action...'. The modal includes a search bar and a list of actions. The 'Data (Things)' action is selected, highlighted with a red border. Other actions listed include Account, Navigation, Data (Things), Email, Payment, Analytics, Element Actions, Plugins, and Custom Events.

Action Type	Action Sub-type	Description
Data (Things)	Create a new thing...	Create a new thing...
	Make changes to thing...	Make changes to thing...
	Make changes to a list of things...	Make changes to a list of things...
	Delete thing...	Delete thing...
	Delete a list of things...	Delete a list of things...
	Copy a list of things...	Copy a list of things...
	Set a thing's slug...	Set a thing's slug...
	Download data as CSV	Download data as CSV
	Upload data as CSV	Upload data as CSV
Plugins	Delete an uploaded file	
Custom Events	Install more data (things) actions...	

- Then you will see more sub-elements, select

Create a new thing...  
Select

Create a new thing...

The screenshot shows a workflow editor interface. On the left, a sidebar lists categories: Design, Workflow (selected), Data, Styles, Plugins, Settings, and Logs. The main area displays two events:

- When Button REGISTER is clicked**
- When Signup / Login Popup is clicked**

Below the events, a modal window is open, titled "Click here to add an action...". It contains a search bar and a list of actions categorized by type:

Action Type	Action Description
Account	Create a new thing... (highlighted)
Navigation	Make changes to thing...
Data (Things)	Make changes to a list of things...
Email	Delete thing...
Payment	Delete a list of things...
Analytics	Copy a list of things...
Element Actions	Set a thing's slug...
Plugins	Download data as CSV
Custom Events	Upload data as CSV
	Delete an uploaded file
	Install more data (things) actions...

- Select

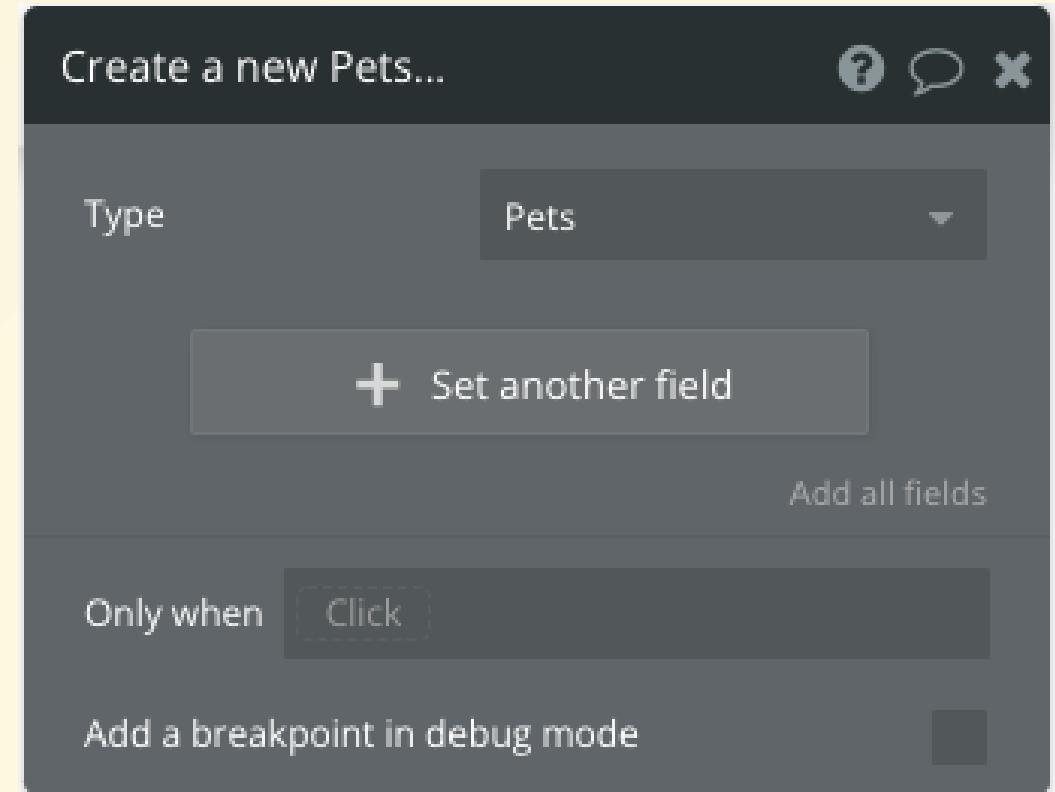
Create a new Pets... When the dialog box

Create a new Pets...' appears, specify the type of pets you want to register.

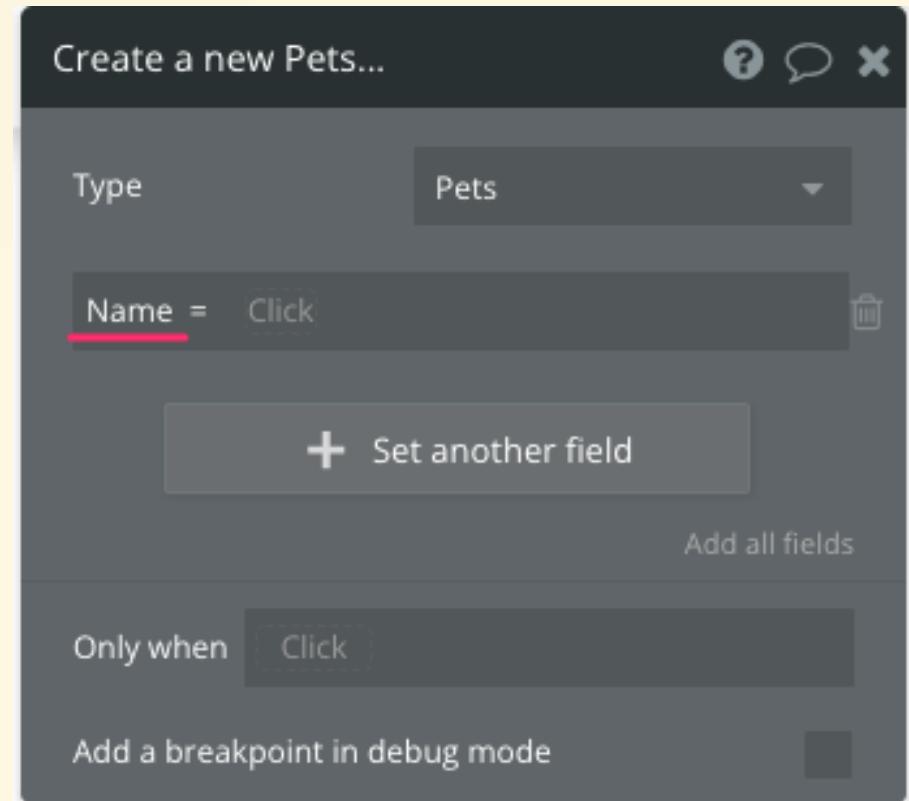
- This time it will be Pets.

- Then, a button

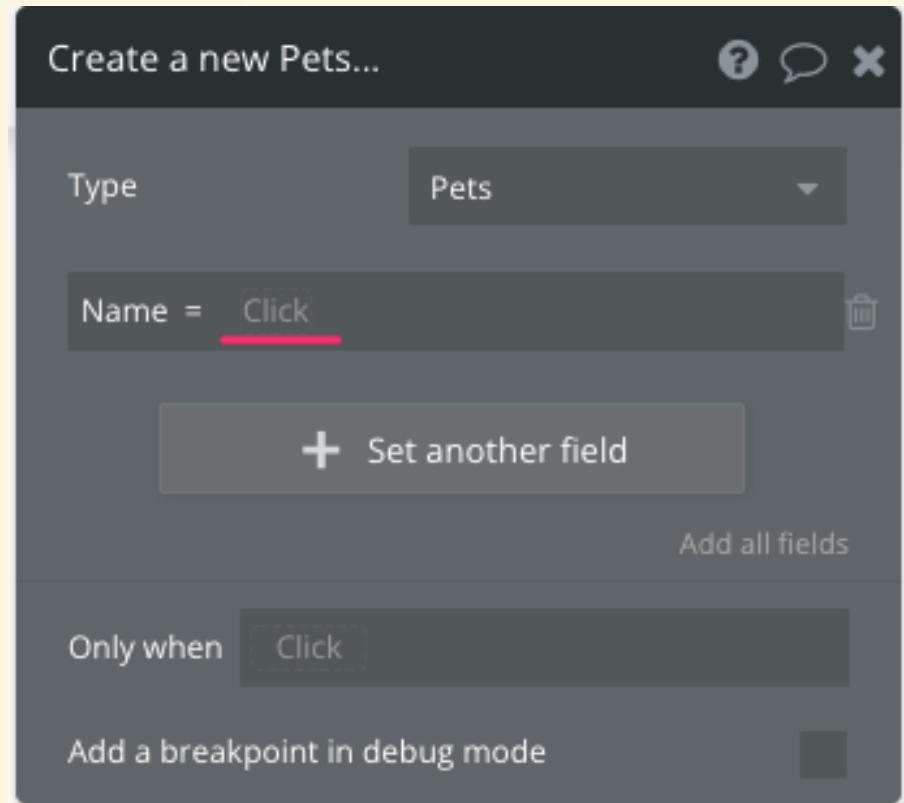
Set another field will appear, and as the title says, set the fields entered on the screen to the "field" of Pets



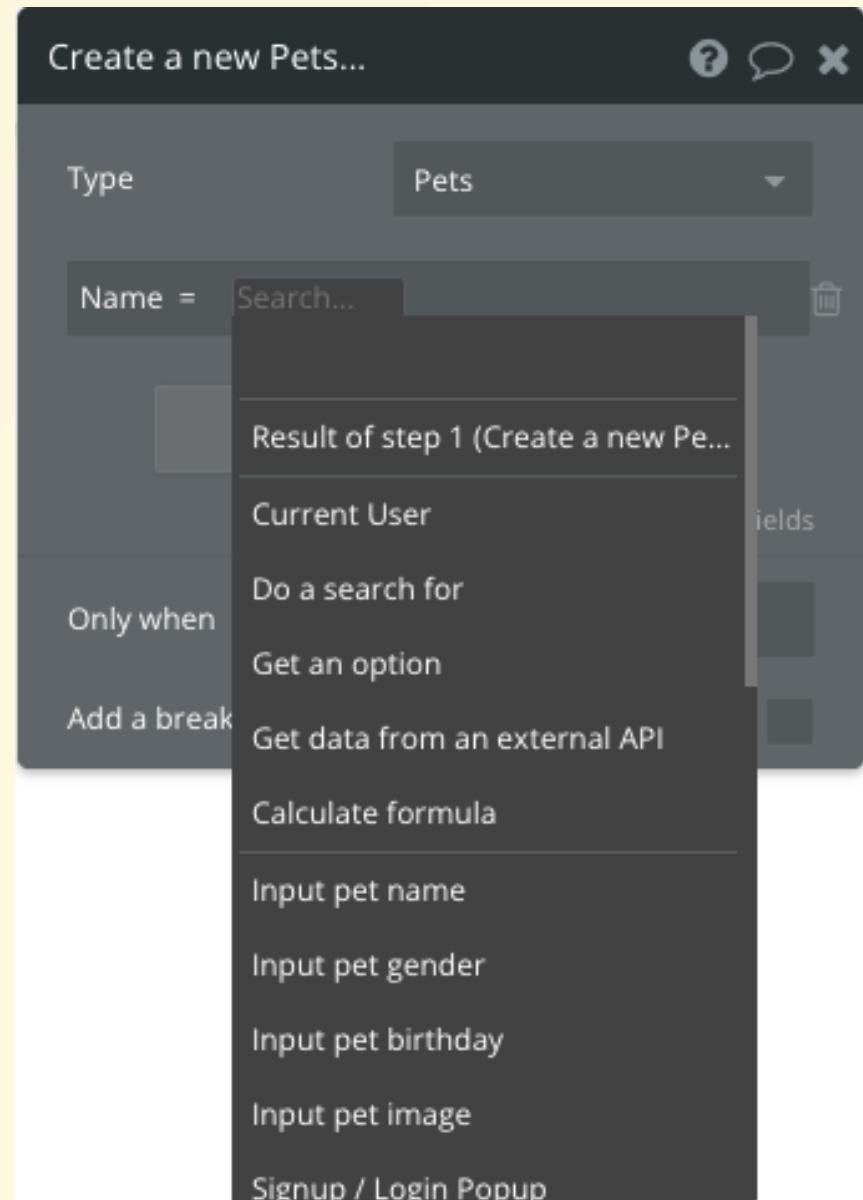
- Here, we specify the name of the "field" of the item we want to store in Pets on the left side, and the actual value to be stored in the "field" on the right side.
  - First, select **Name** for the left side



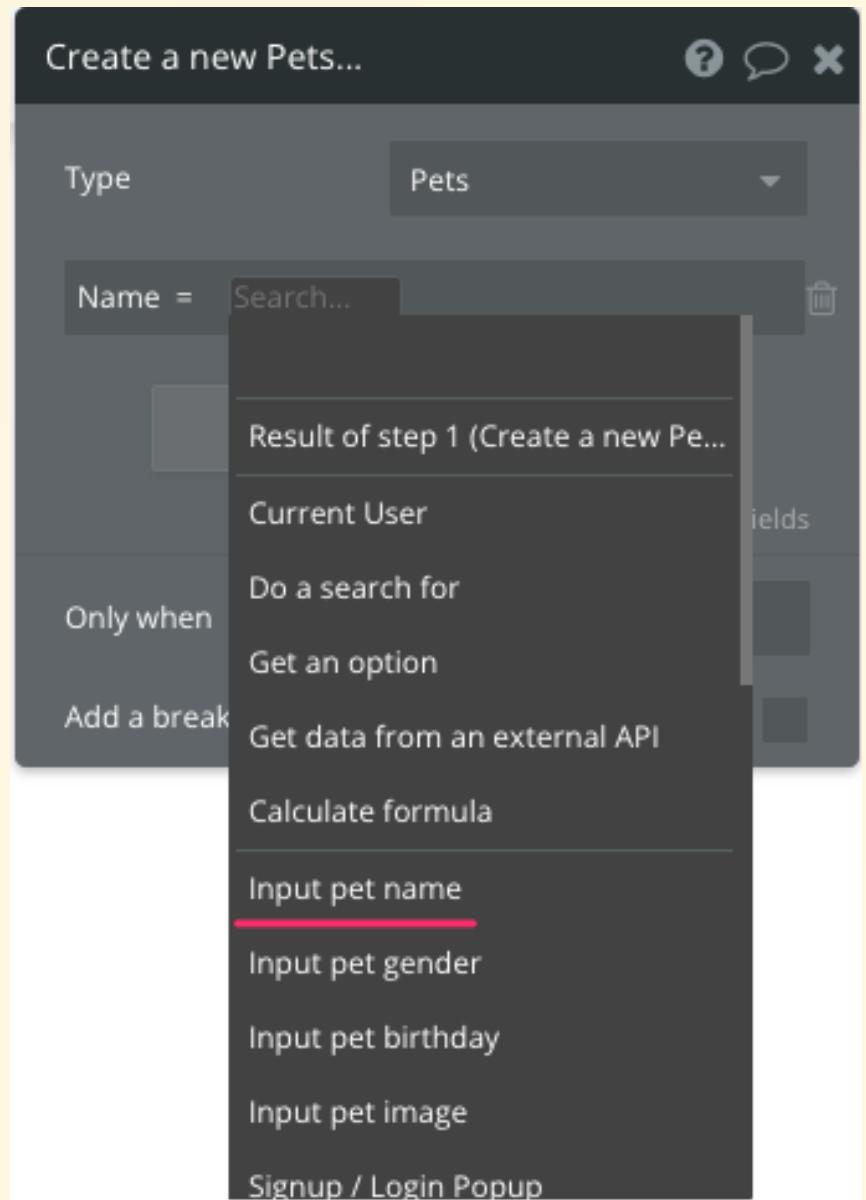
- If you specify the left side, you will see `Click` on the right side, sandwiched between `=`.



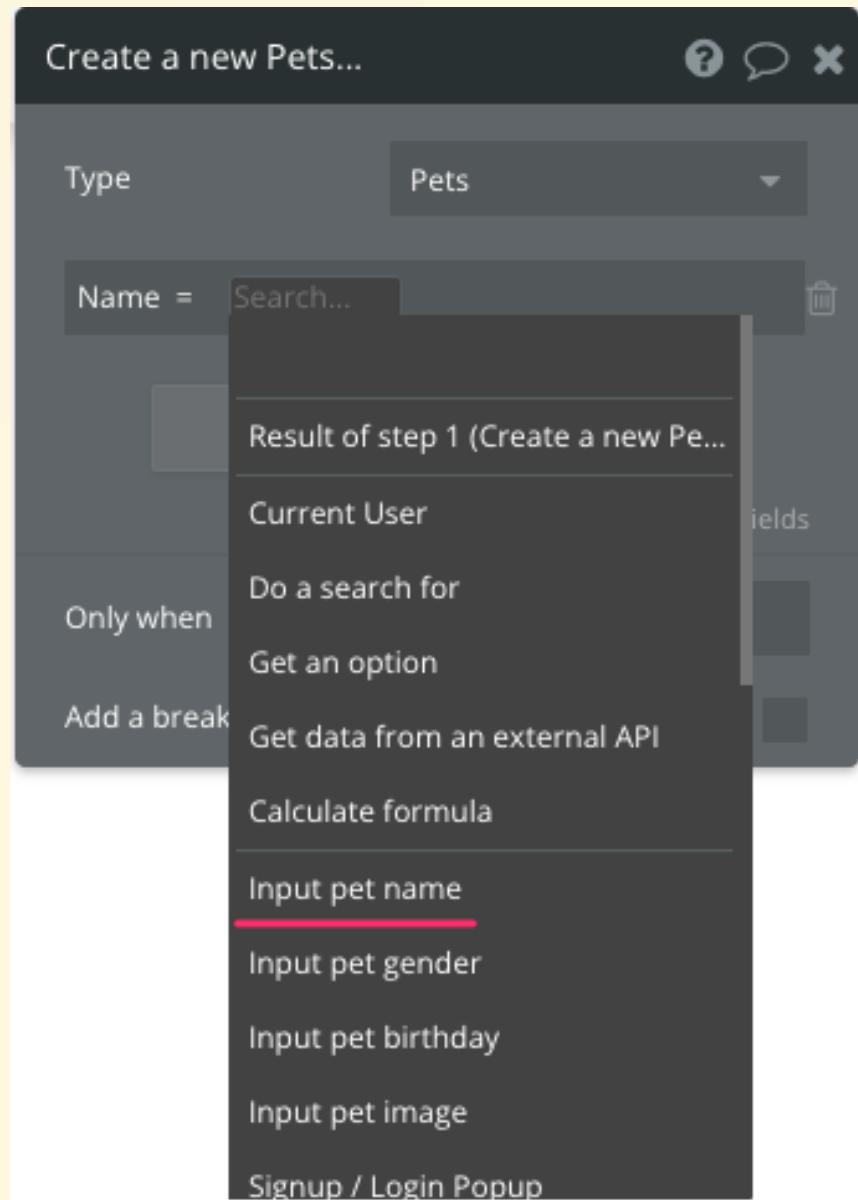
- If you click here, you'll see a variety of options.
- I'm not going to go into details here, but you can specify what you want to set for the field selected on the left side



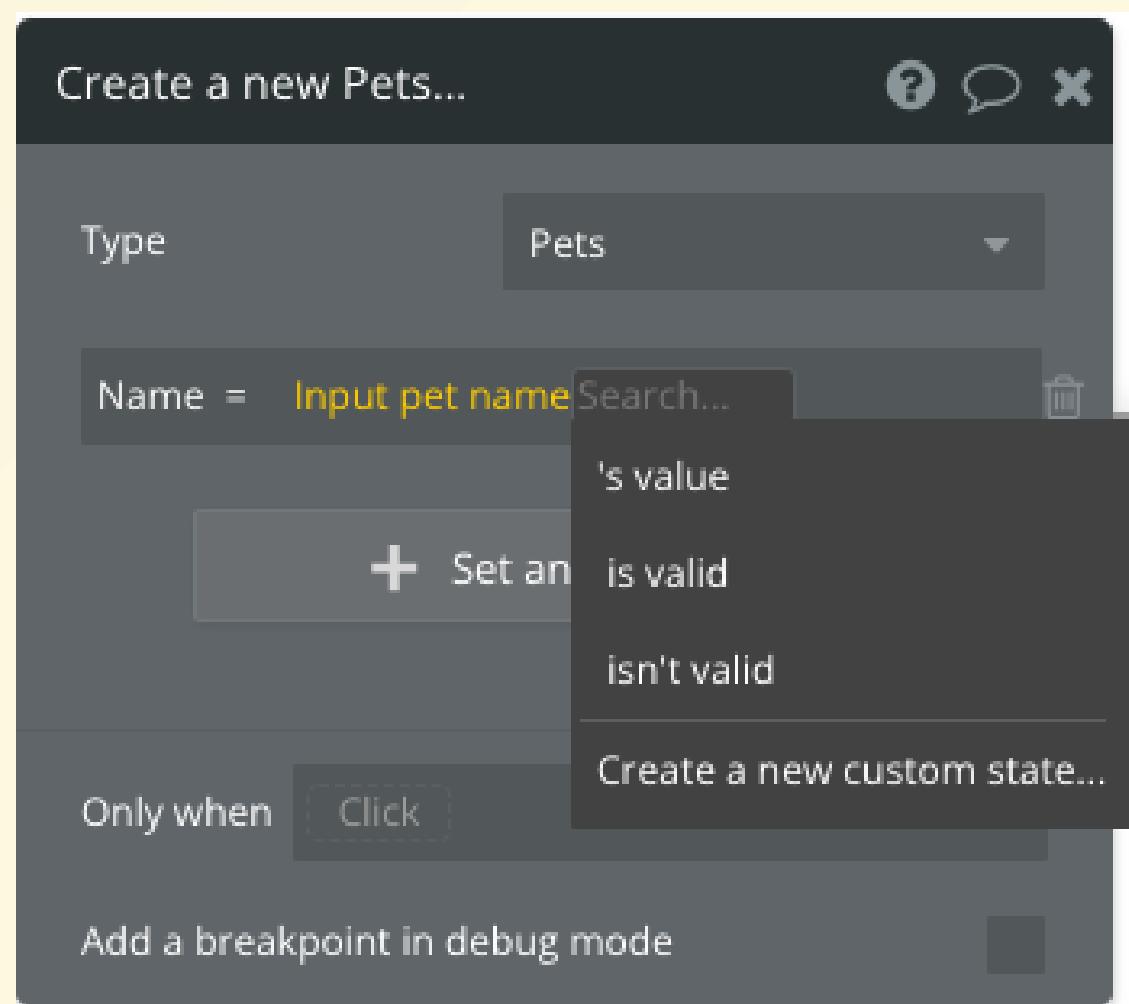
- Select the **Input pet name** field since this will be the name of the pet entered on the screen.



- This `Input pet name` is the name you gave to the input element when you created the screen, and it is displayed here.
- This makes it easier to uniquely identify the input elements when there are multiple input elements on a screen, for example



- If you select **Input pet name**, more options will be displayed
- We want to use the "value" of the pet name entered on the screen, so we will select **'s value**.
  - The other options **is valid** and **isn't valid** can be used to check if the input value is valid or not



- If you specify all of them, it looks like this

Create a new Pets...

Type Pets

Name = Input pet name's value 

Image = Input pet image's value 

Birthday = Input pet birthday's value 

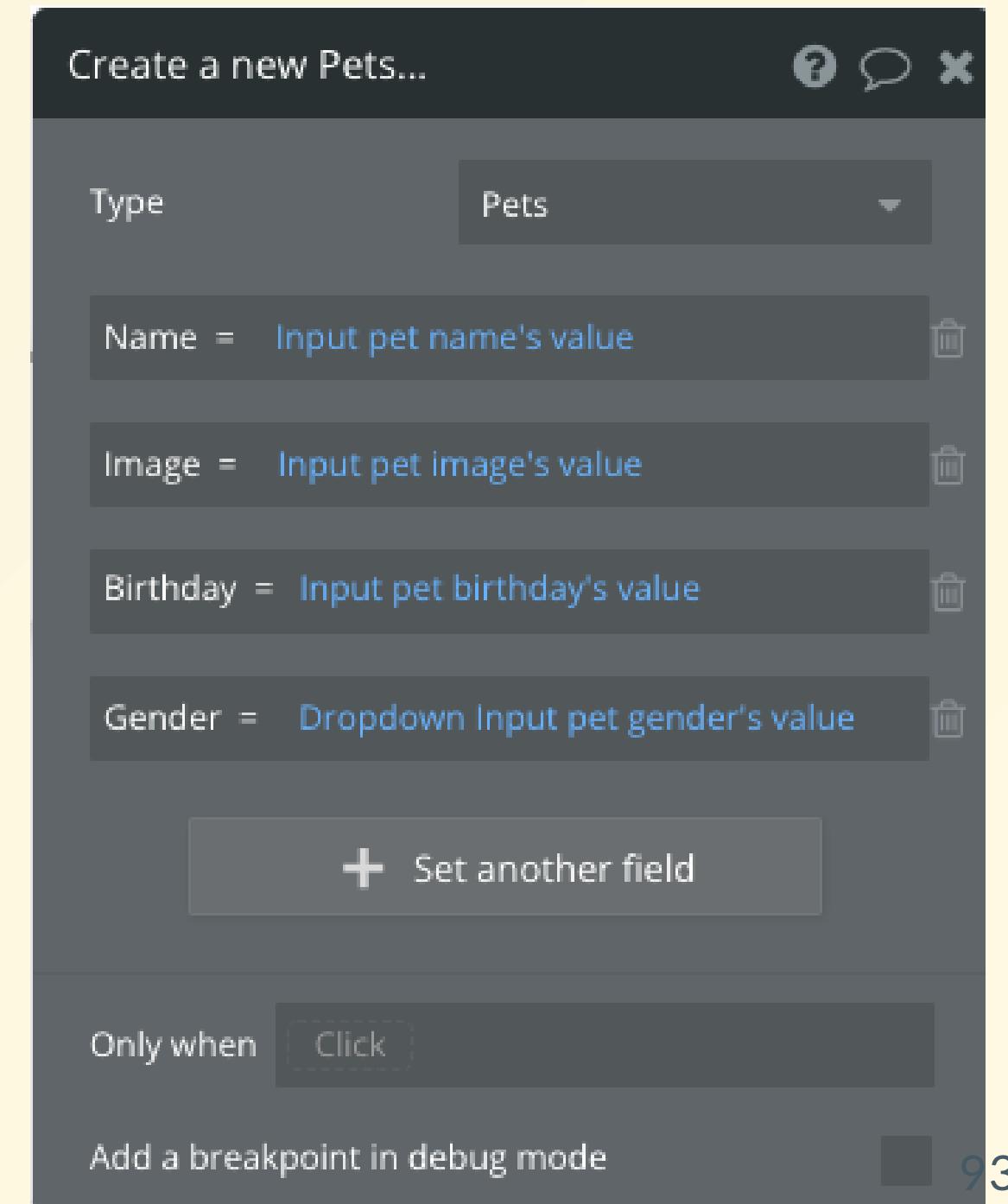
Gender = Dropdown Input pet gender's value 

+ Set another field

Only when  

Add a breakpoint in debug mode 

93



# Let's try to make it work!

- Once you've set up the settings so far, let's check if your pet is actually registered in the preview!
  - By the way, you can click "Ctrl + P" to see the preview (shortcut key).
- Fill in all the input elements and click the "REGISTER" button.
- Nothing will happen on the screen, but let's see if the data has actually been registered.

- Close the preview, open the Data tab, select App data, and when you select **All Pets** in the left panel, make sure that the information you just entered from the screen is saved

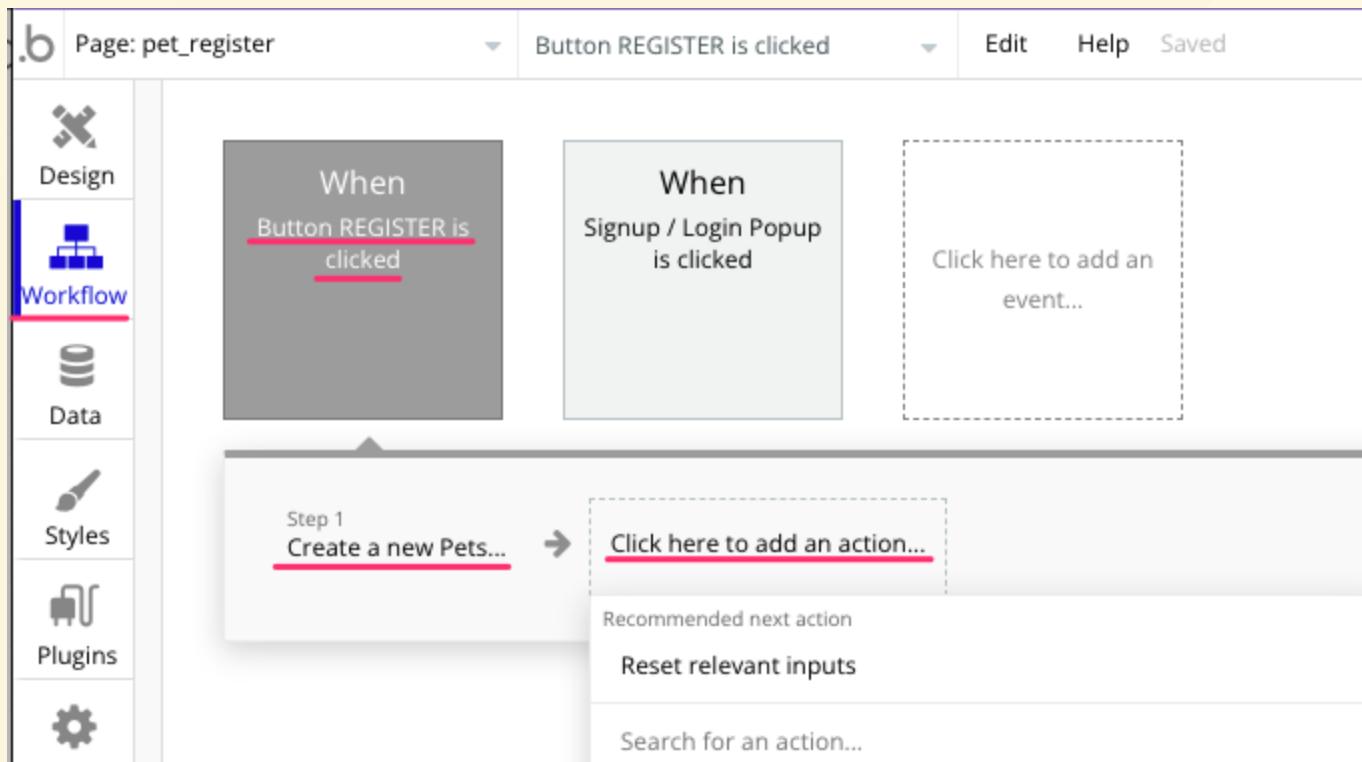
The screenshot shows the Airtable interface with the following details:

- Page:** pet\_register
- Tab:** Data (highlighted with a red border)
- Sub-tab:** App data (highlighted with a red border)
- View:** Application data - All Pets - Development version
- Buttons:** New view, Primary fields, Search, 1 entries (displaying 1), New entry, Delete (0), Upload
- Search bar:** Search view names or data types...
- Table Headers:** Birthday, Gender, Image, Name, Created Date
- Data Row:** (Nov 1, 2021 12:00 am) | Male |  | たま | Nov 8, 2021 10:41 pm
- Left Panel:** Shows filters for All Pets and All Users, each with edit icons.

- Now we're up to the point where we can save what we've entered on the screen to the database! 
- Next, let's create the screen transitions.

# Let's try screen transitions.

- Click on the Workflow tab in the left panel.
- You will see the behavior when you click the "REGISTER" button that you just set.



- Here, to the right of "Create a new Pets..." to the right of "Create a new Pets...", click `Click here to add an action... Click here to add an action...

The screenshot shows a workflow editor interface. On the left, there's a vertical toolbar with icons for Design, Workflow (which is selected), Data, Styles, Plugins, Settings, and Logs. The main area has a header bar with 'Page: pet\_register', 'Button REGISTER is clicked', and 'Edit Help Saved'. Below this, there are two main event boxes: 'When Button REGISTER is clicked' and 'When Signup / Login Popup is clicked'. A third box, 'Click here to add an event...', is shown with a dashed border. At the bottom, there's a step labeled 'Step 1 Create a new Pets...' followed by an arrow pointing to a dropdown menu with the text 'Click here to add an action...'. This dropdown also includes 'Recommended next action' and 'Reset relevant inputs'. A search bar says 'Search for an action...' and a table lists actions under 'Account' and 'Navigation': 'Go to page ...' (under Account), 'Refresh the page' (under Navigation), and 'Go to previous page' (under Navigation). The 'Click here to add an action...' text is highlighted with a red underline.

- Now select **Navigation** and click on **Go to page...** as a sub-element. Click `!

The screenshot shows a workflow editor interface. On the left, there's a sidebar with icons for Design, Workflow (which is selected), Data, Styles, Plugins, Settings, and Logs. The main area has a header: 'Page: pet\_register' and 'Button REGISTER is clicked'. Below the header, there are two event boxes: 'When Button REGISTER is clicked' and 'When Signup / Login Popup is clicked'. Underneath these is a step box labeled 'Step 1 Create a new Pets...'. A context menu is open over this step box, with the 'Action' tab selected. The menu includes a 'Recommended next action' section with 'Reset relevant inputs' and a 'Search for an action...' input field. Below this is a list of actions categorized by icon: Account (User icon), Navigation (Link icon), Data (Things) (Database icon), and Email (Email icon). The 'Navigation' category is currently selected, and the 'Go to page ...' option is highlighted with a red underline.

- When Button REGISTER is clicked
- When Signup / Login Popup is clicked
- Click here to add an event...

Step 1  
Create a new Pets...

Click here to add an action...

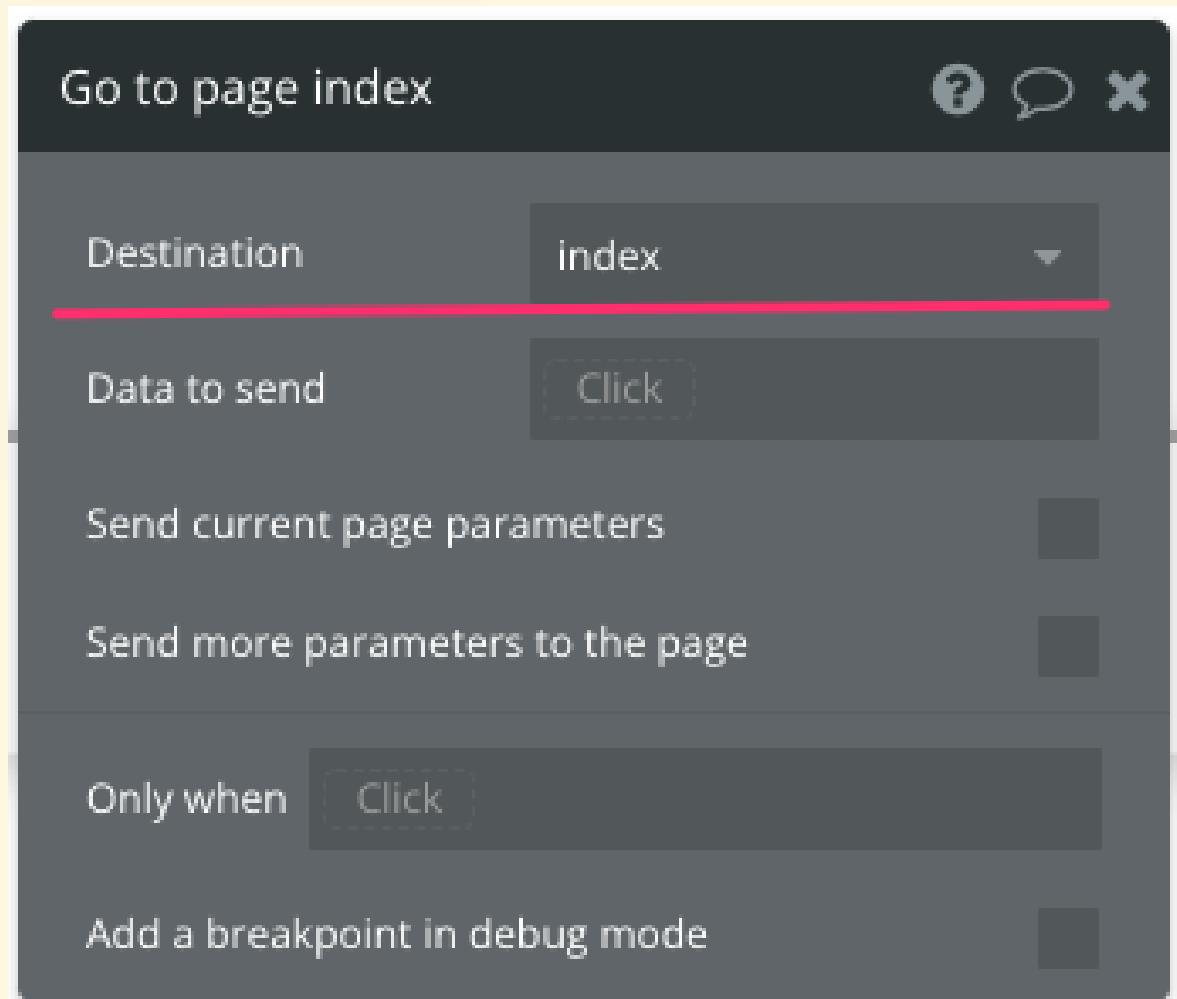
Recommended next action

Reset relevant inputs

Search for an action...

Account	Go to page ...
Navigation	Refresh the page
Data (Things)	Go to previous page
Email	Open an external website
	Add a pause before next action

- Then a dialog box will appear.
- In this case, we want to go to the Pet List page, so select `index`.



- Now let's preview it!
- Enter your pet's information as before, and click the "REGISTER" button.
- You should now be taken to a completely new page.
  - We haven't created the pet list page yet, so it's not a problem now.
- Now you've mastered the screen transition! 

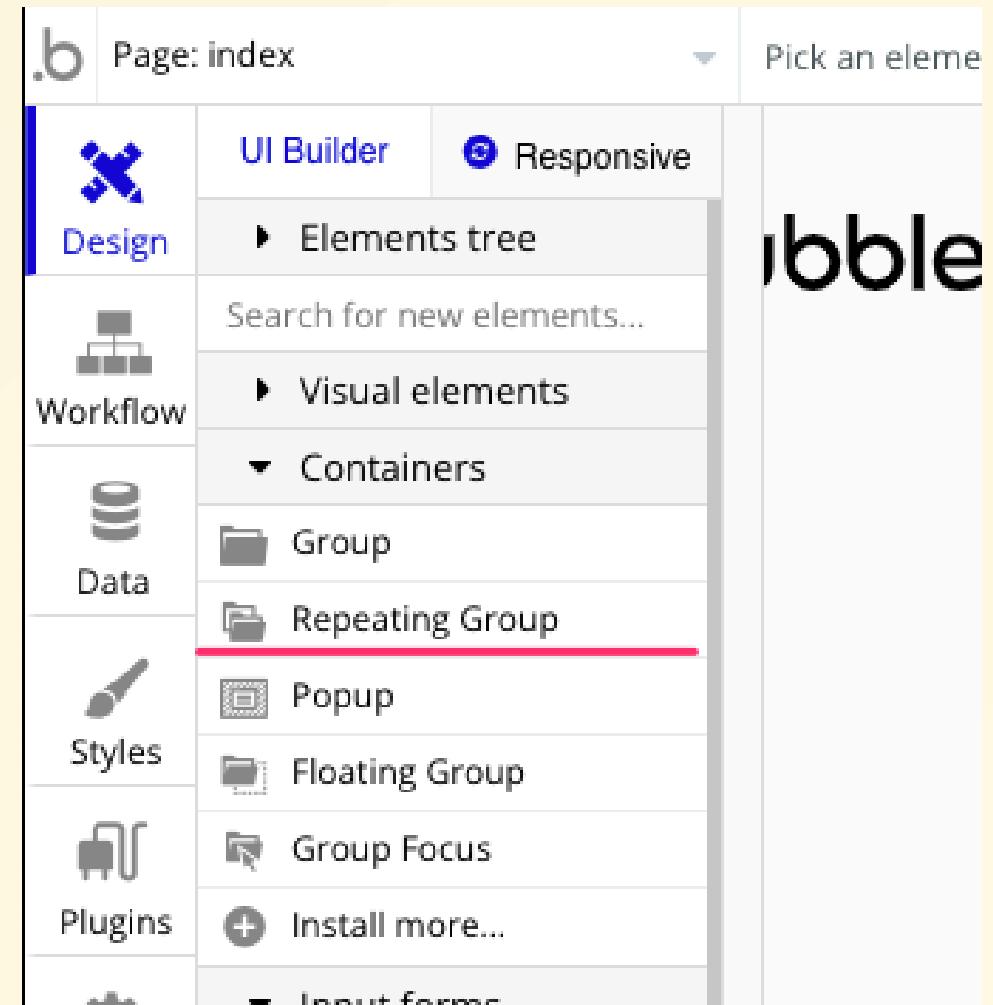
# Pet List View

- Next, let's create the pet list view.

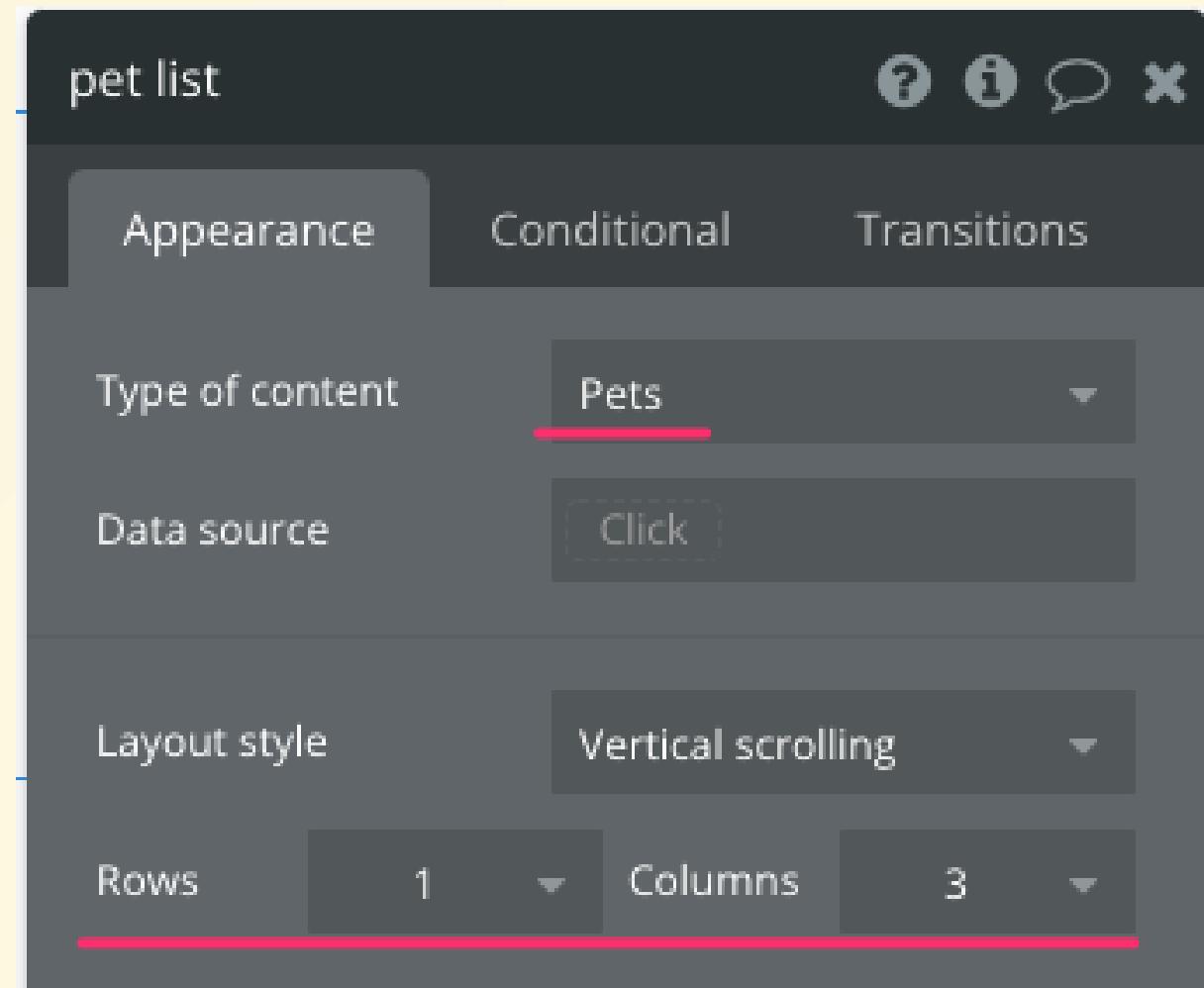
The screenshot shows a user interface for a pet management application. At the top left is the logo ".bubble". On the right side, there are three navigation links: "Bubble Forum", "FAQs", and a teal-colored "LOG OUT" button. Below the header, there is a grid of three images of dogs, each labeled with its name in Japanese:

- A black Shiba Inu standing outdoors, labeled "タマ" (Tama) below the image.
- A white Chihuahua lying down, labeled "ボチ" (Botchi) below the image.
- A tan Shiba Inu drinking from a water fountain, labeled "マロン" (Maron) below the image.

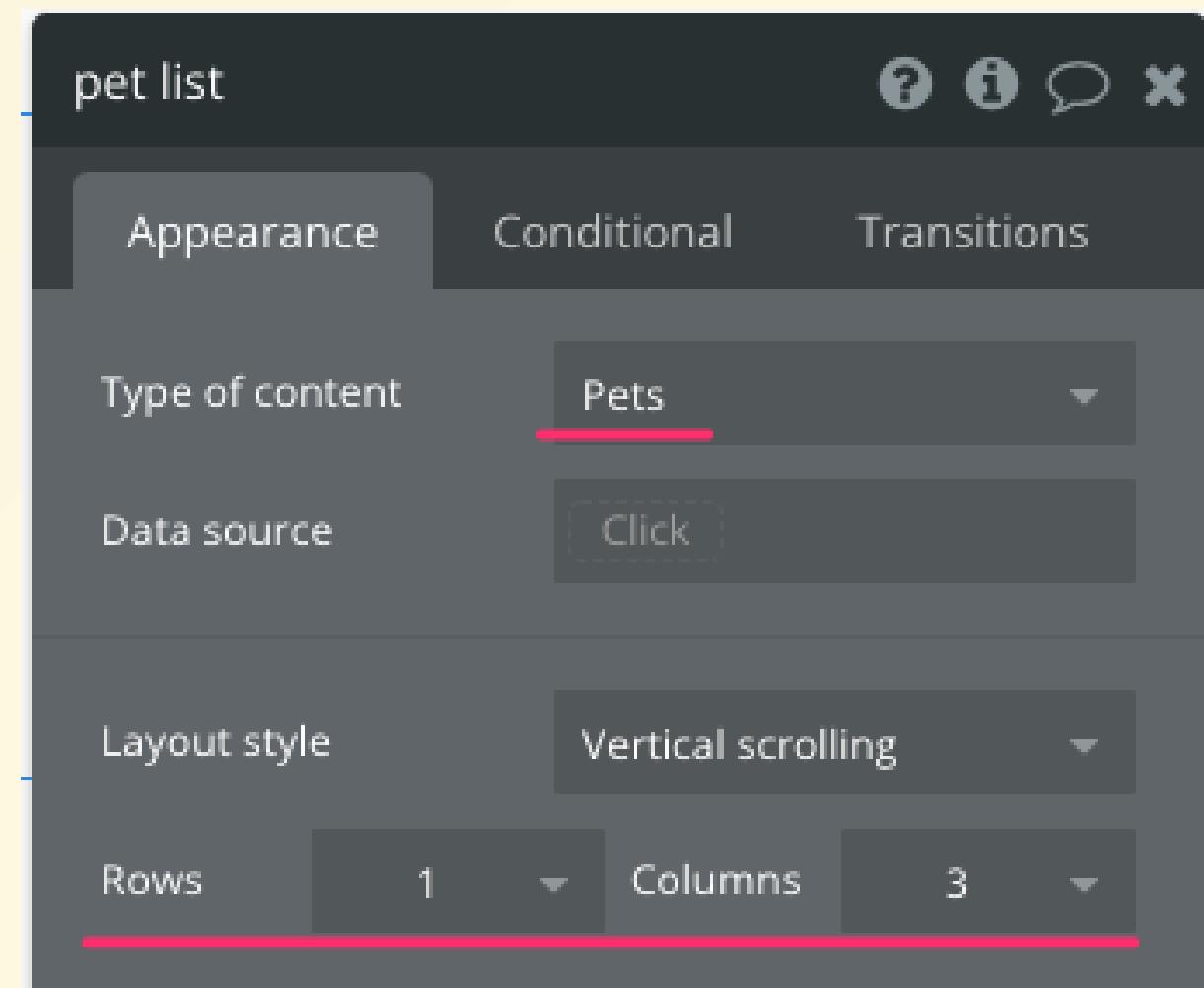
- Select **index** from the list of pages in the upper left corner.
- If you want to repeat the same element, select **Repeating Group** from the list of `Containers
- Drag it to the canvas in the right panel



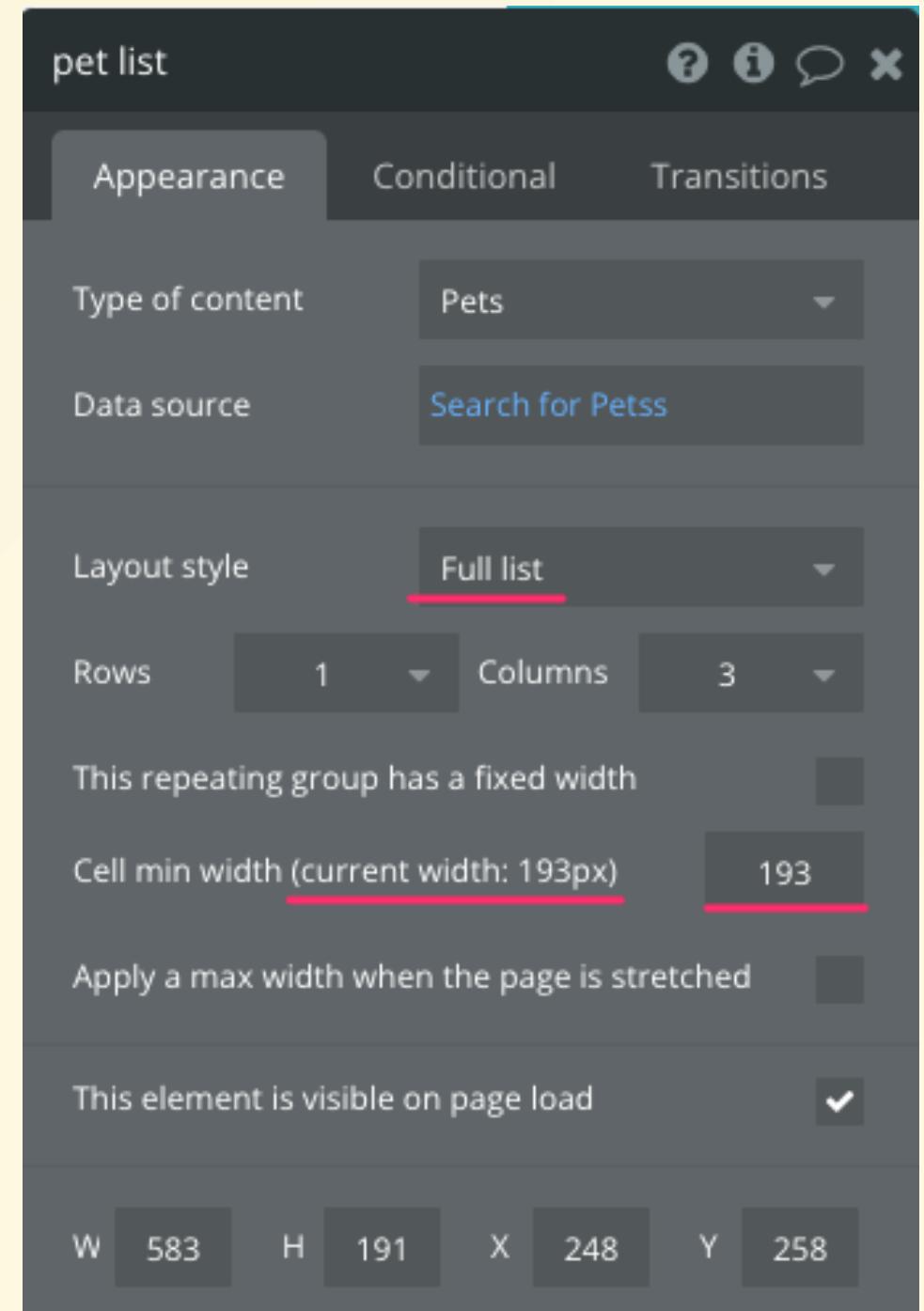
- The popup for the element's advanced settings will appear, so let's configure it
- Enter `pet list` for the element name at the top.
- For `Type of content`, specify the type of data to be repeated.
  - In this case, it's `Pets`.



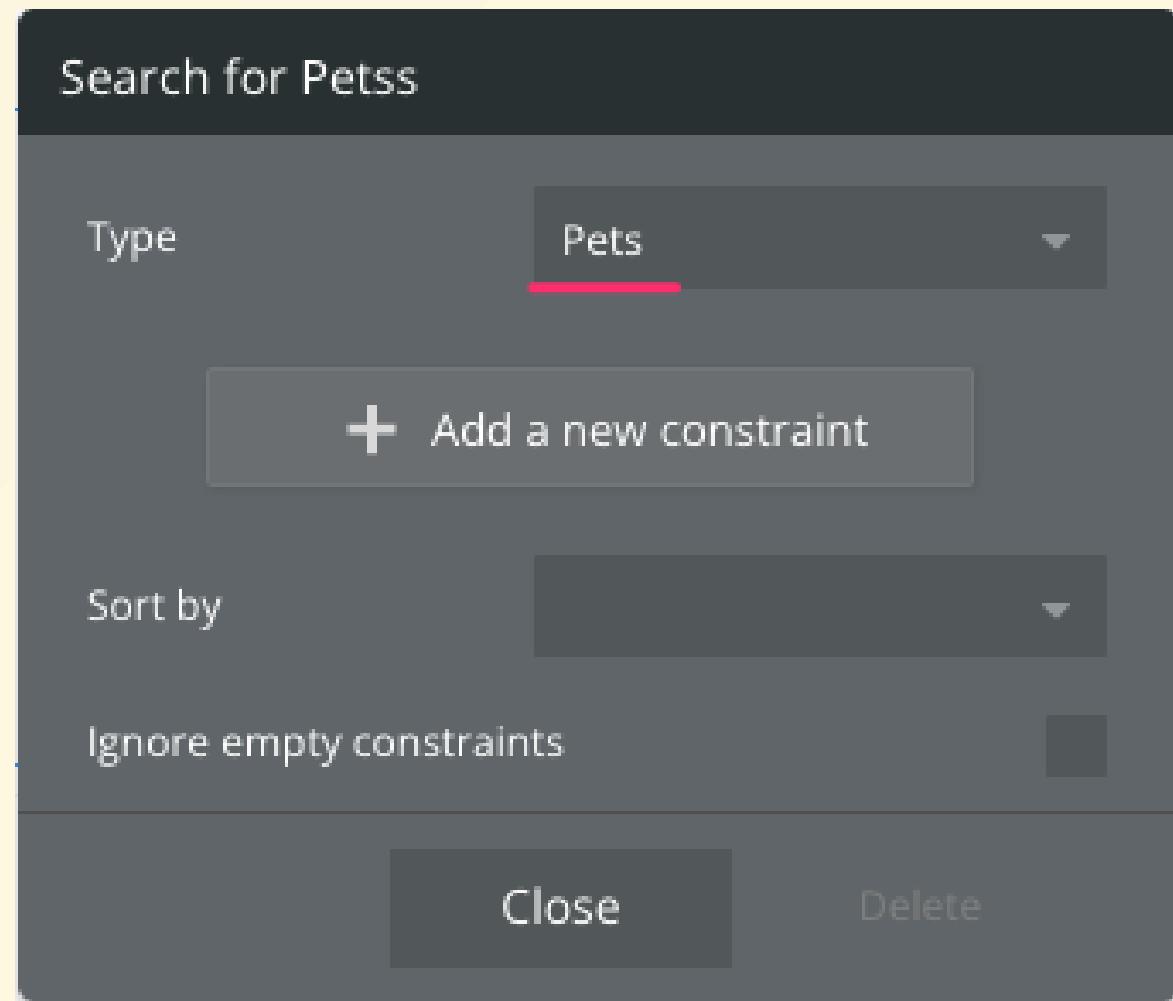
- **Rows** is the number of rows to display by default.
  - If there is more than one row of data, the number of rows is automatically set to **1**.
- **Columns** is the number of columns per row.
  - In this case, **3**.



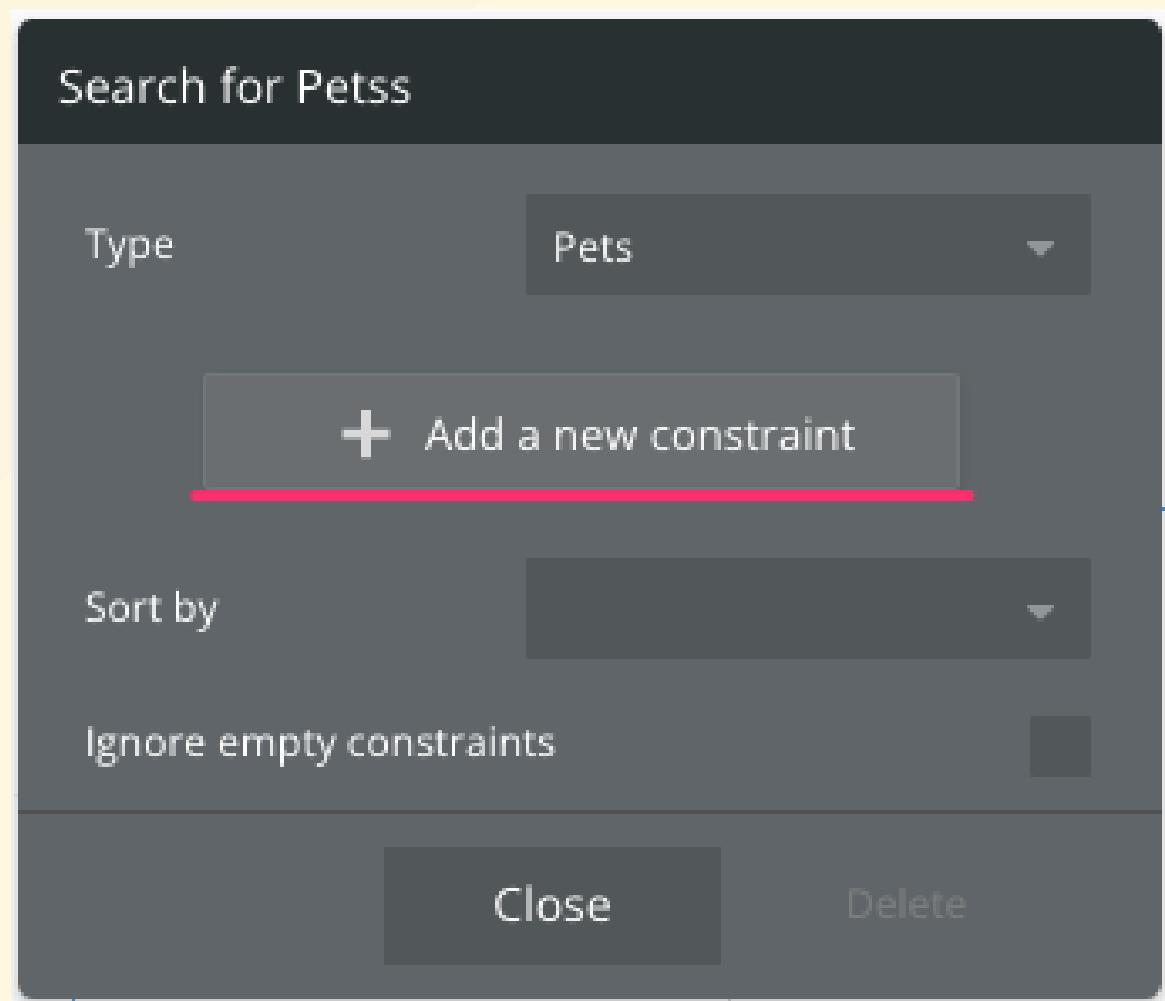
- Select **Full list** for "Layout style" to automatically expand the list if there are more than 4 Pette data.
- Also, set the Cell min width to the same value as the current width listed in parentheses (193 in the image example) to display the full width of the image in each cell.



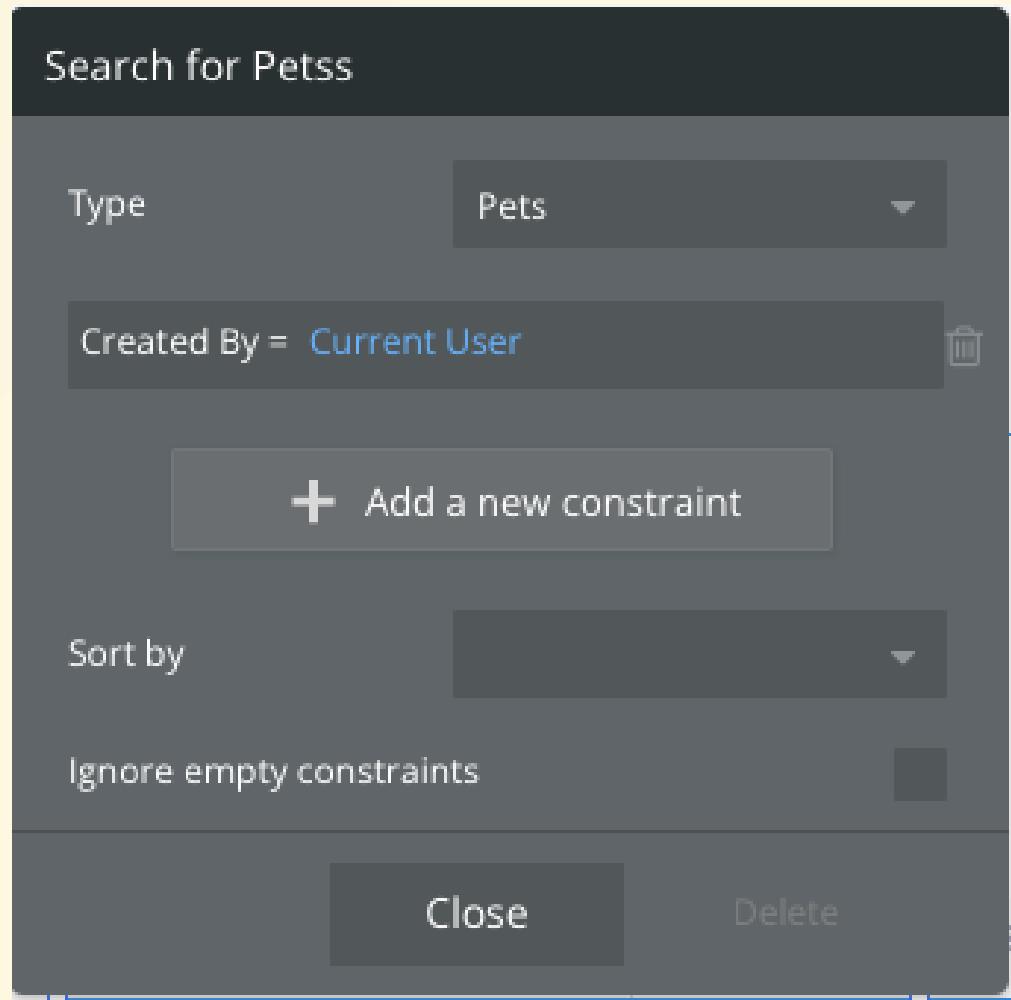
- Then click **Do a search for** from the **Data source** Click
  - This is to specify the data source to be listed
- Another popup will appear and you can specify **Pets** for Type



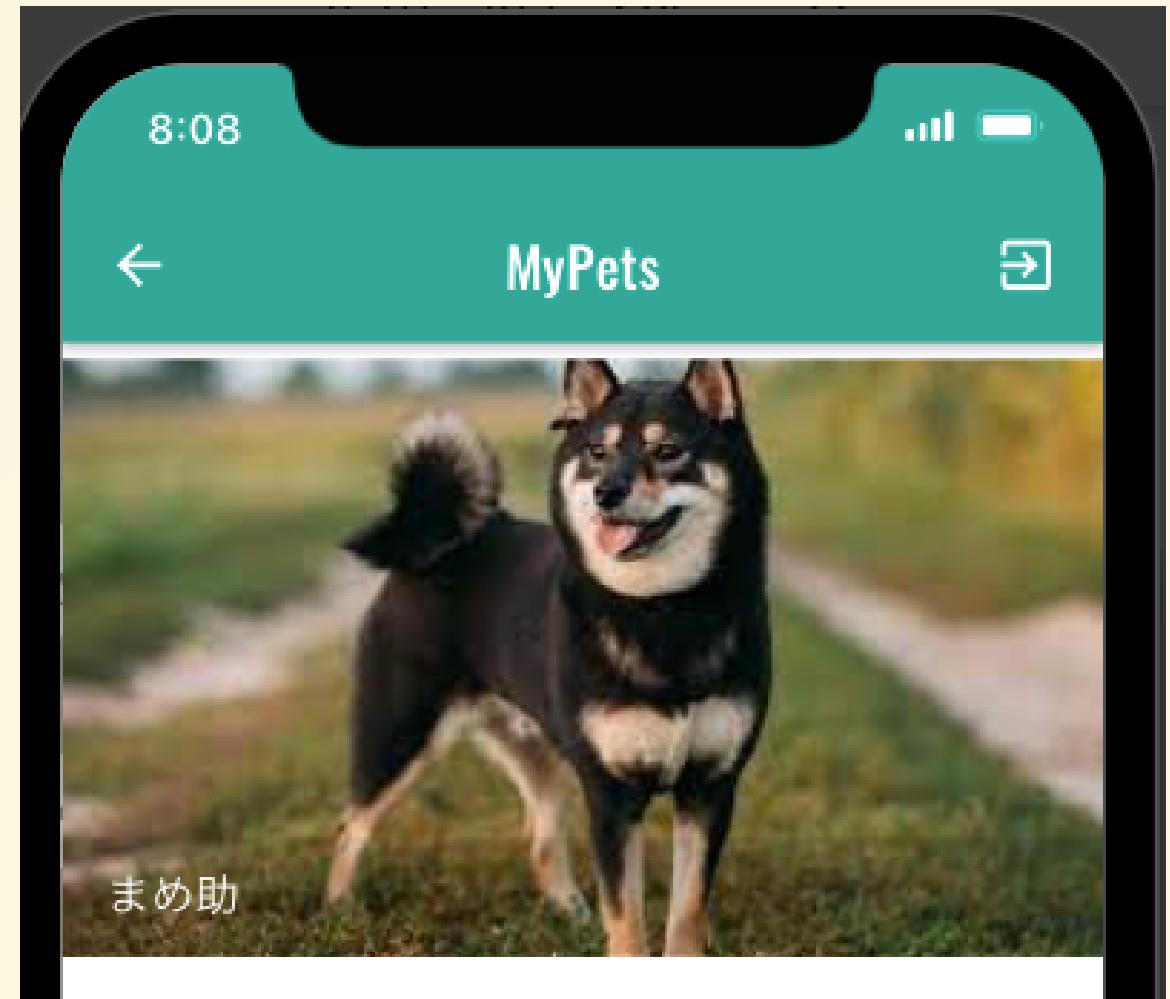
- Since this is the only way to display the list of pets even if the user is not logged in, we will add a condition to display only the pets registered by the logged-in user
- Click "Add a new constraint" to add more conditions.
  - What will be the contents of the configuration?



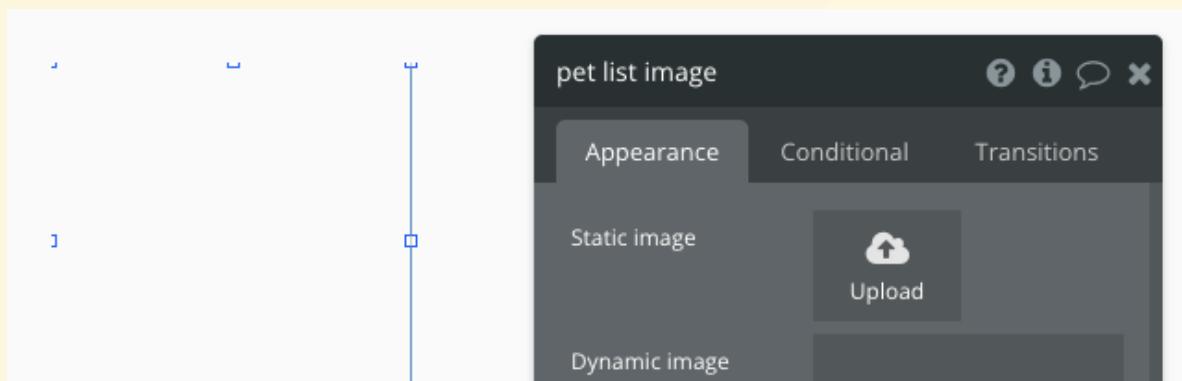
- The condition to be set is that the "Created By" of the pet must be the same as the "Current User".



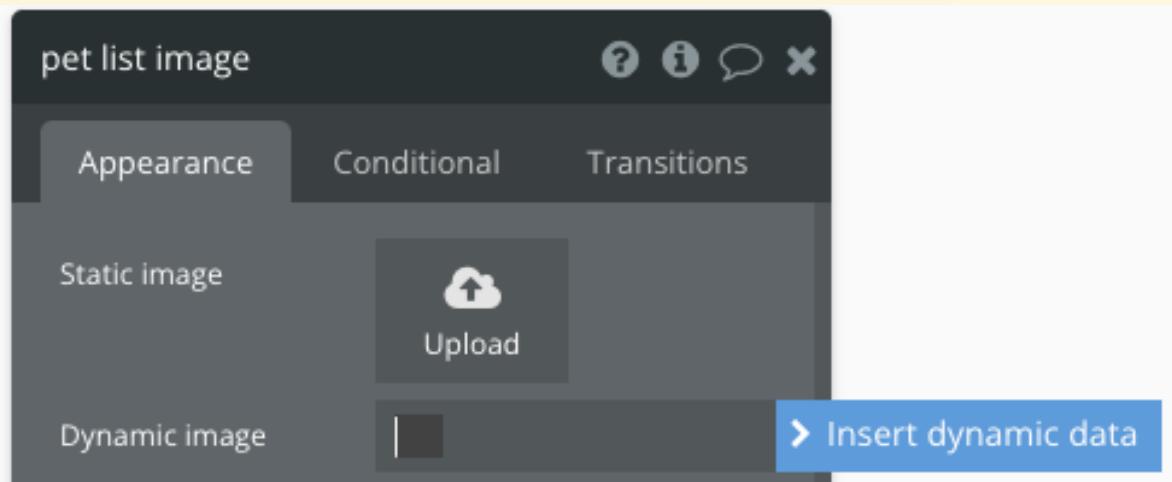
- The next step is to set the content to be repeated.
- Let's display the image and the name, just like we did in Adalo



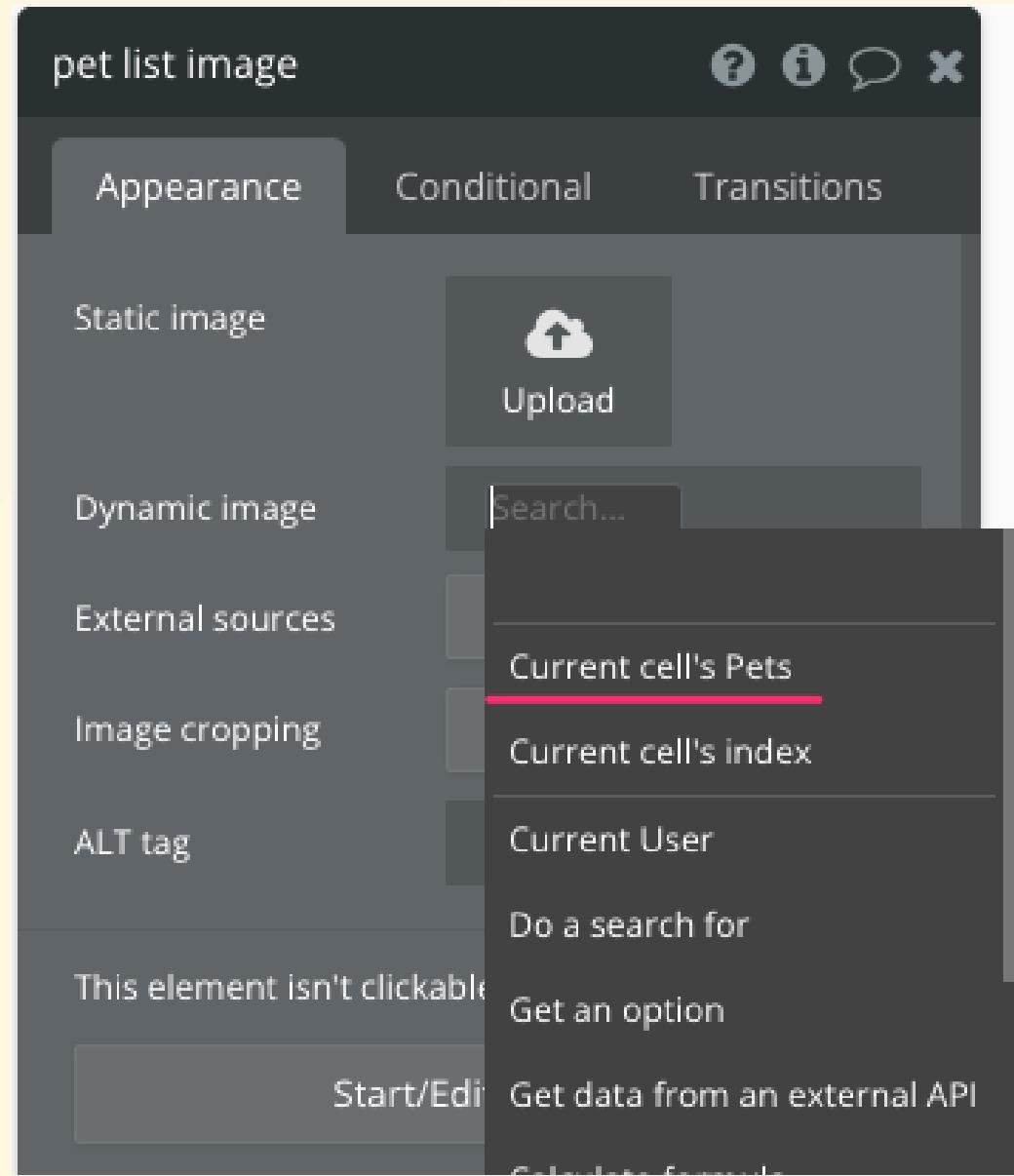
- First, select **Image** from **Visual elements** and drag it to the right panel.
  - If you drag the image to include it in the Repeating Group you just placed, you won't have to move it later.
  - You can also include it in the Repeating Group afterwards
  - After placing the element, drag it into the Repeating Group to expand it.
- Let's name the element **pet list image**.



- After placing the elements, select "Dynamic image" from the advanced settings popup that appears.
- You will see a button (or something like that) that says  
**Insert dynamic data .**
  - This function can be used when you want to display values and images dynamically, instead of displaying fixed values and images.



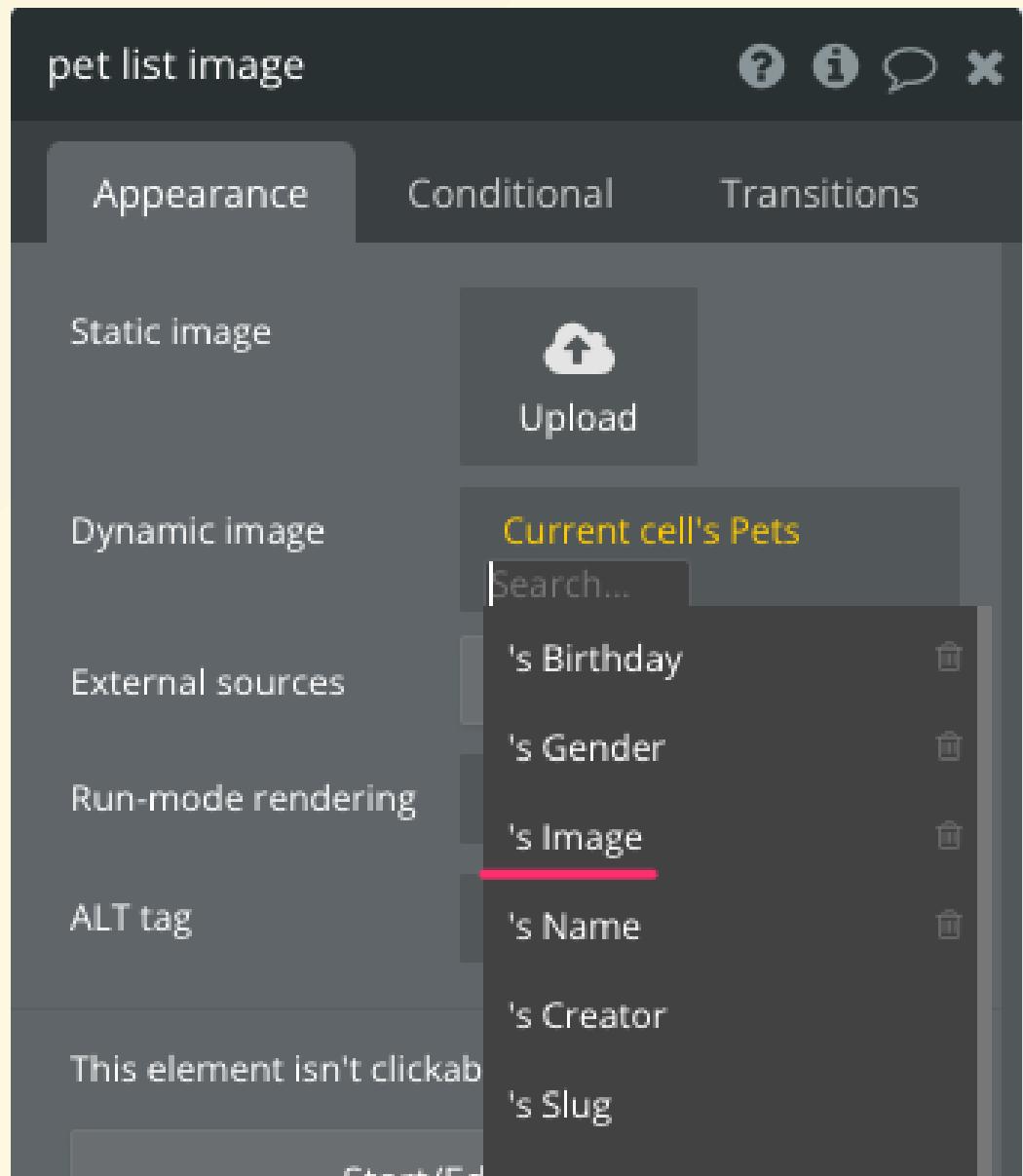
- Click on the pull-down menu, and then click on **Current cell's Pets**.
  - This can be used if you literally want to use the data of the current cell's pets



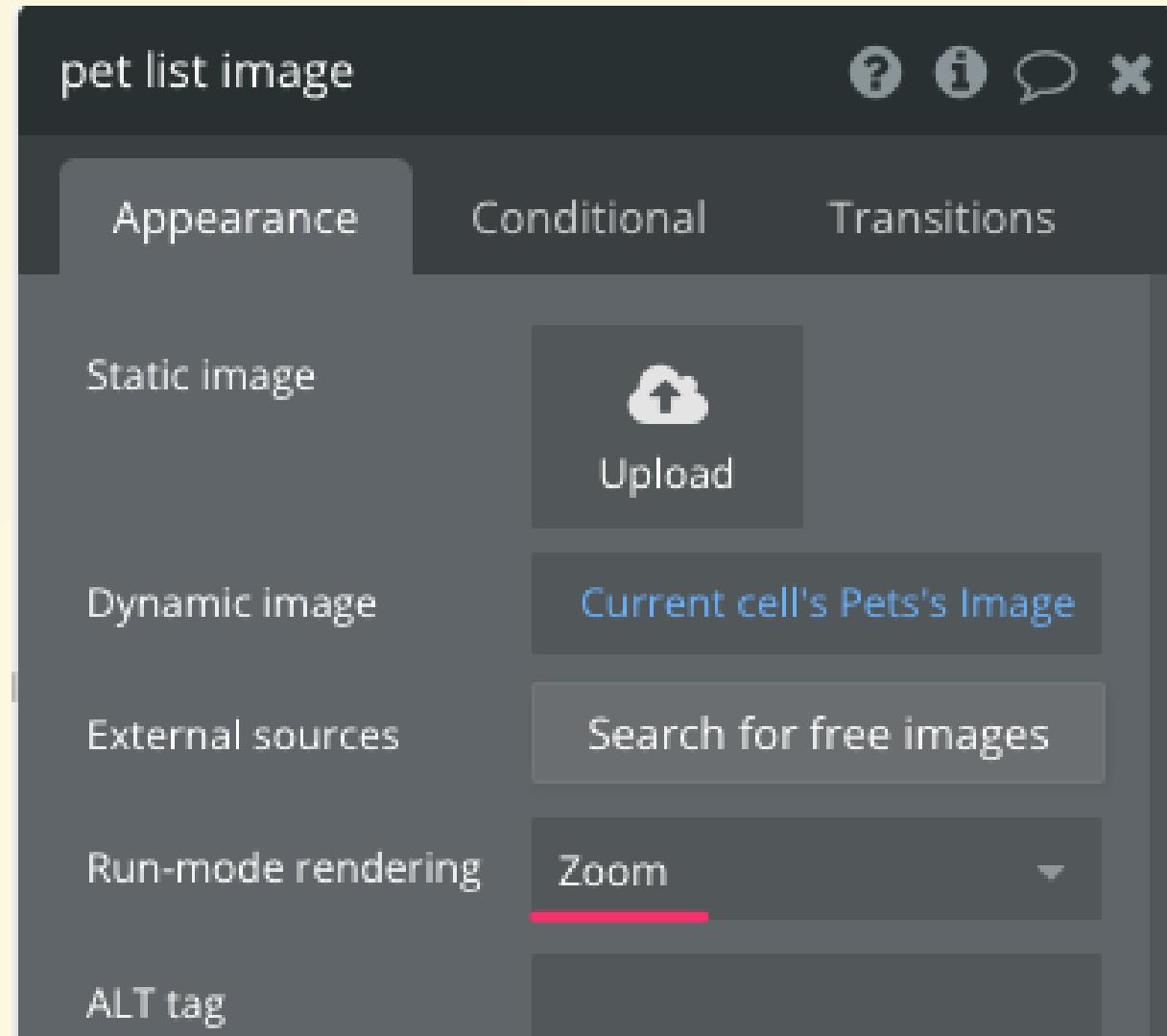
- If you select

Current cell's Pets , a pull-down menu will appear, showing the fields that the Pets type has.

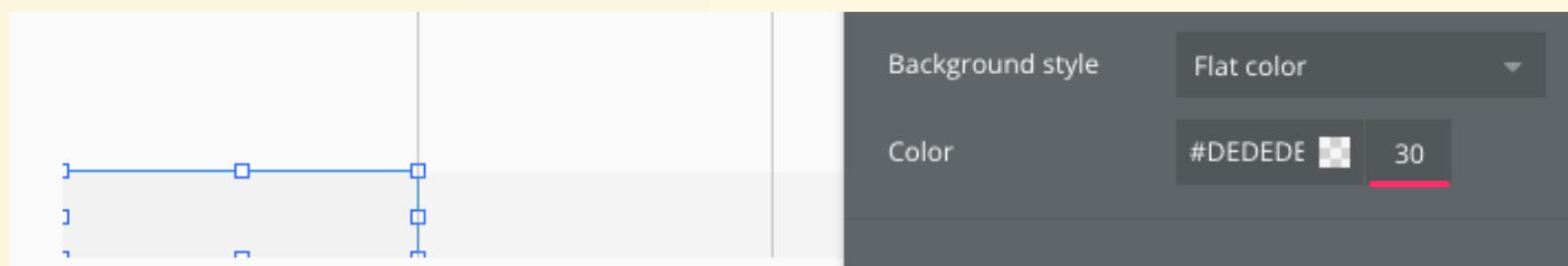
- The command will display the images of the pets in the current cell, so if there are multiple pet data, each cell will have its own image



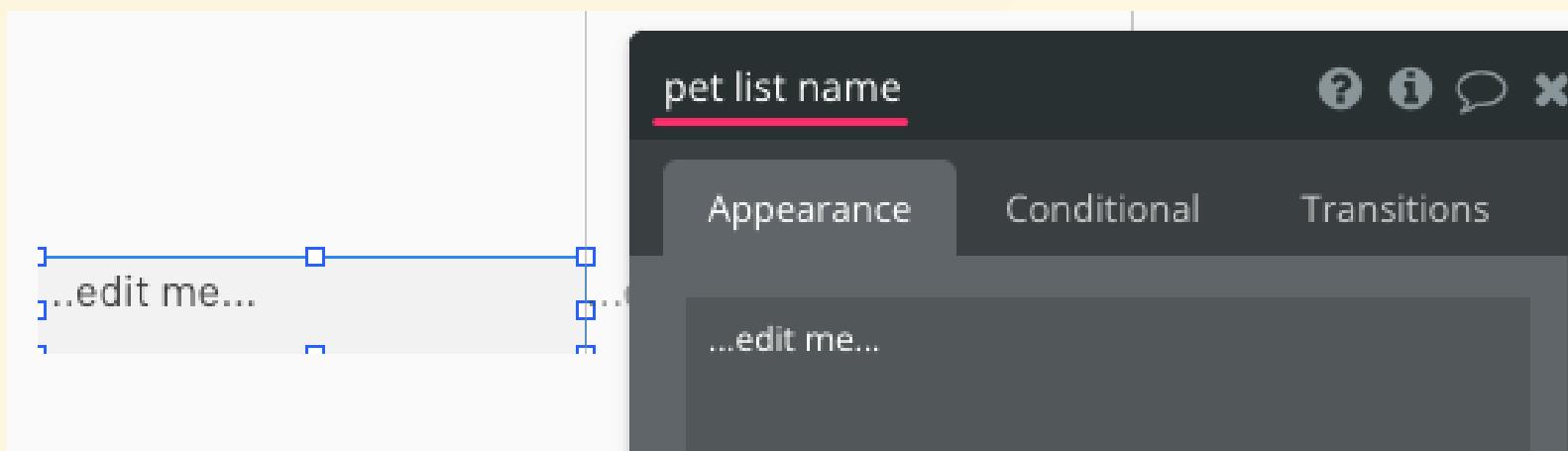
- Now that we've set the image to fill the width of the cell, we need to change the "Run-mode rendering" value of the image element from `Streth` to `Zoom`.
  - Change the value of the "Run-mode rendering" of the image element from `Streth` to `Zoom`, otherwise the image will automatically change its aspect ratio in the cell.



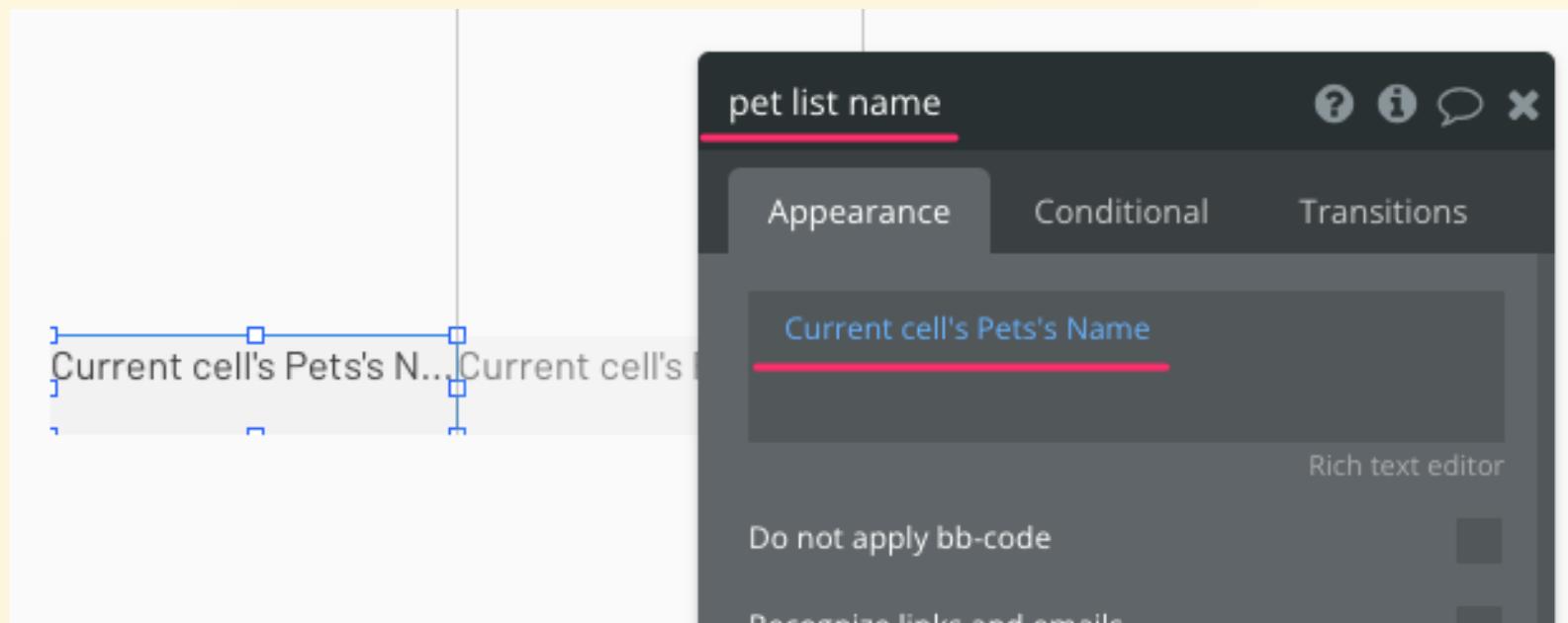
- Next, let's set the background for the name to be displayed below the image.
- Select **Shape** from **Visual elements** and drag it to the right panel.
  - Drag it to the right panel and include it in the Repeating Group.
  - After placing it, move it to the bottom of the Repeating Group and drag it to the full width to expand it.
- Set the value next to the Color of the placed shape to **30**.
  - This is the transparency of the image, with 100 being non-transparent and 0 being transparent



- Finally, we will display the pet's name
- Select **Text** from **Visual elements** and drag it onto the Shape you just placed.
  - Drag it to include it in the Repeating Group.
  - After placing it, drag it to the same size as the Shape to expand it
- Name the element **pet list name**.



- Like the pet image, the pet name also uses Dynamic data
- .... .edit me... Click on the ...edit me...' button, and as with the image, Insert dynamic data will be displayed.  
From there, select Current cell's Pets --> 's Name.



# Now that you have done this, let's preview it!

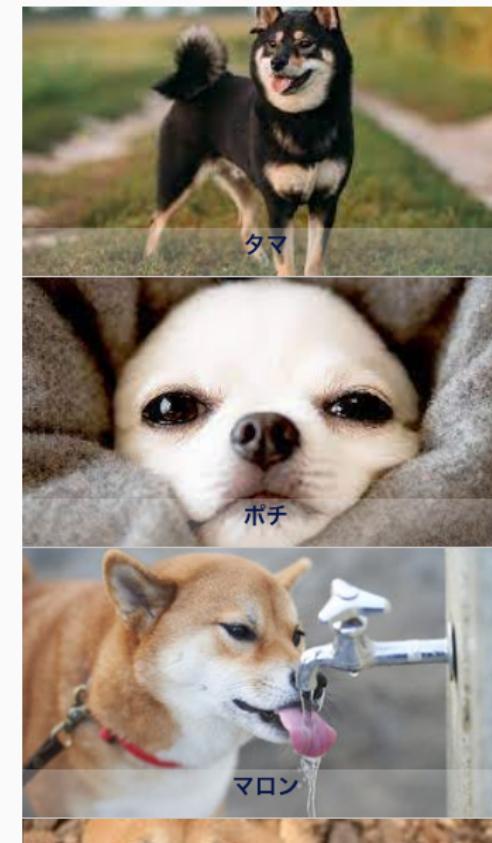
- Do you see the pet's image and the pet's name as the list of pets?

The screenshot shows a user interface for a pet-related application. At the top left is the logo ".bubble". At the top right are three navigation links: "Bubble Forum", "FAQs", and a teal "LOG OUT" button. Below the header is a horizontal scroll bar. The main content area displays three images of dogs with their names written in blue text below each image. From left to right: 1. A black and tan Shiba Inu standing in a grassy field, labeled "タマ". 2. A close-up of a white Chihuahua's face, labeled "ポチ". 3. A tan and white Shiba Inu drinking from a water fountain, labeled "マロン".

- By the way, Repeating Group is responsive, so if you change the width of your screen, it'll automatically resize the image for you!

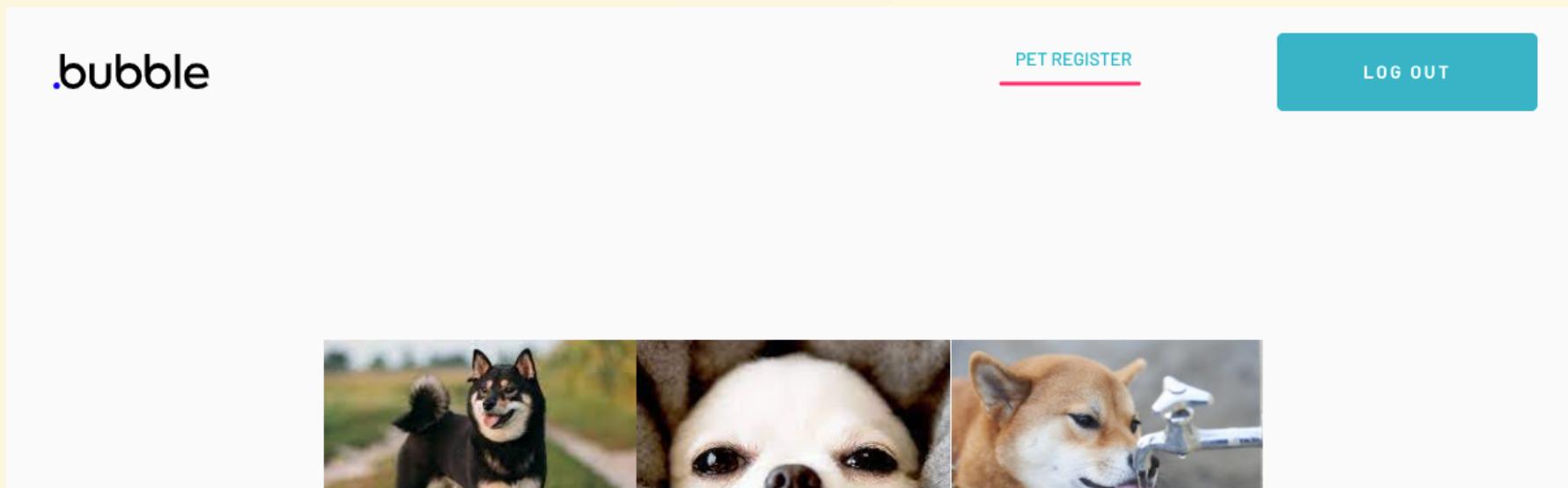
.bubble

LOG OUT

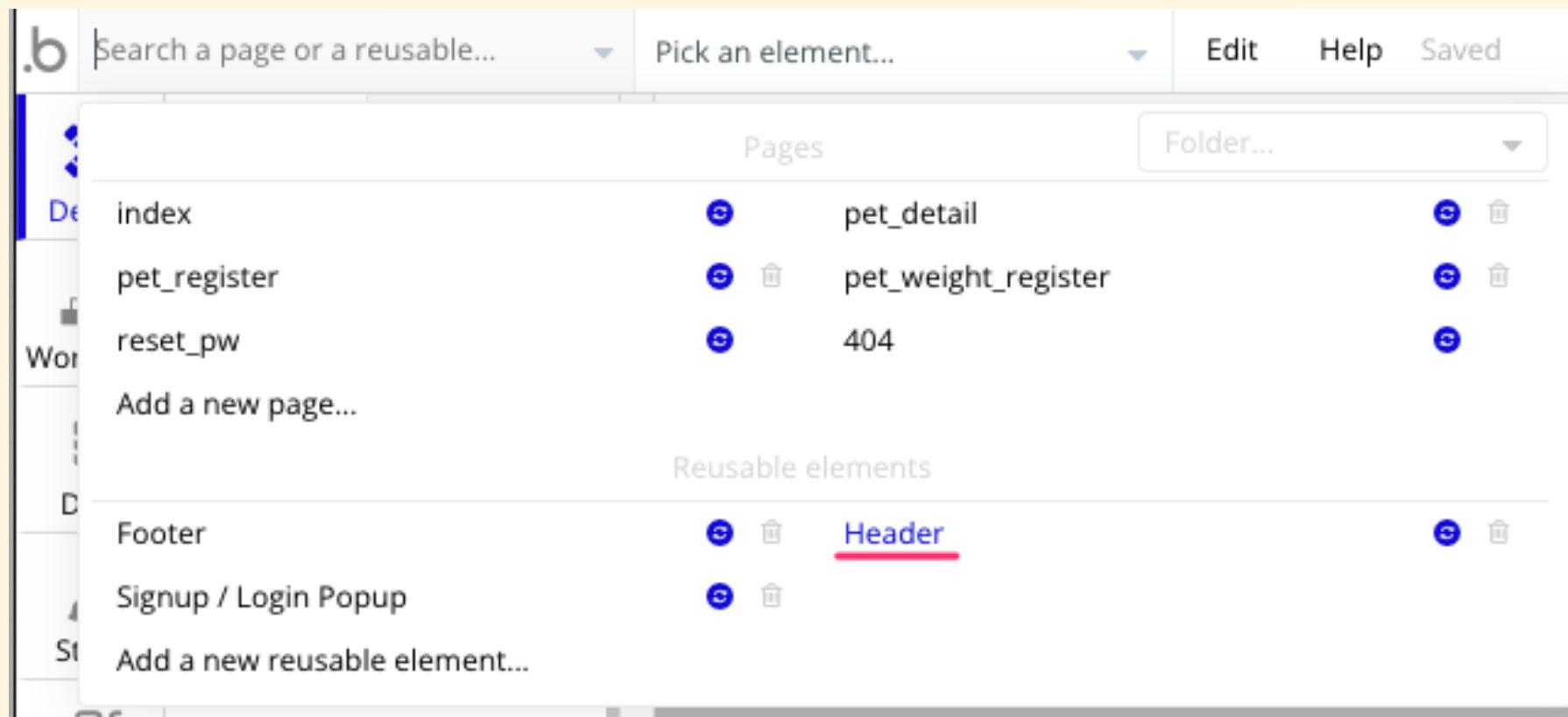


# Let's prepare a lead line from the pet list to the registration screen!

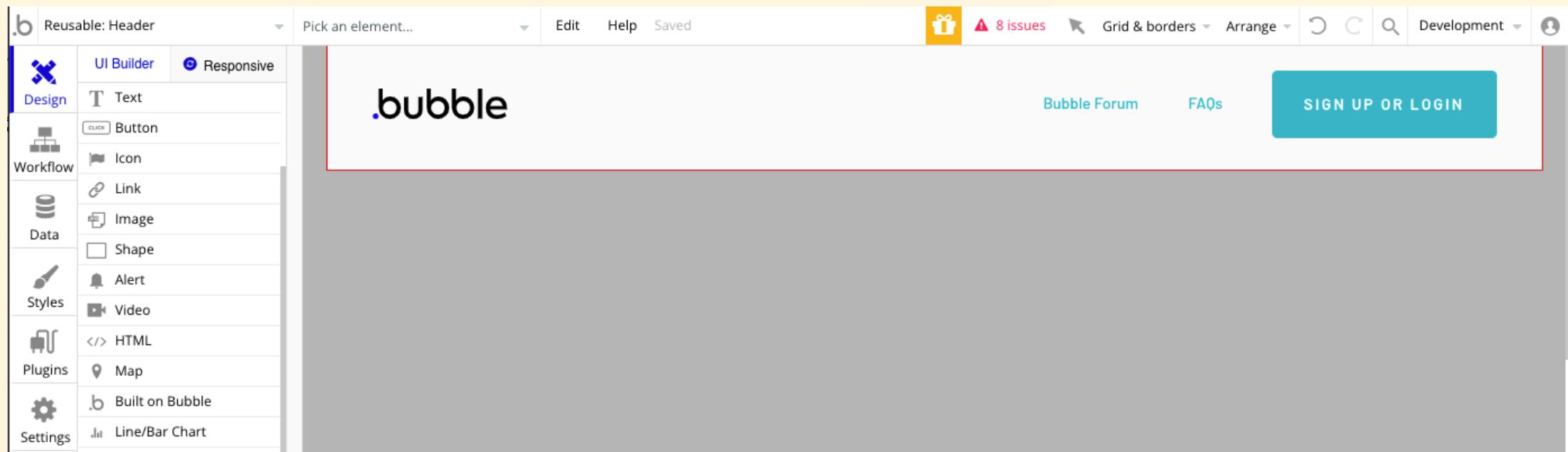
- Now that we've created the pet list screen, let's prepare a line from the list screen to the registration screen.
- Let's prepare this line so that we can move to the registration page from the Header page at any time.



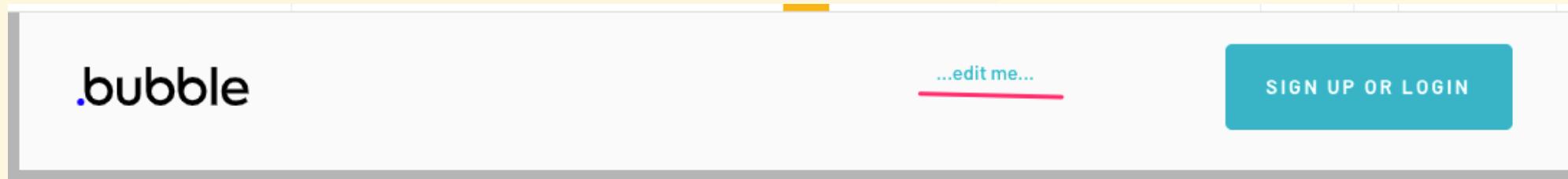
- Click on the menu in the upper left corner and select Header from it.



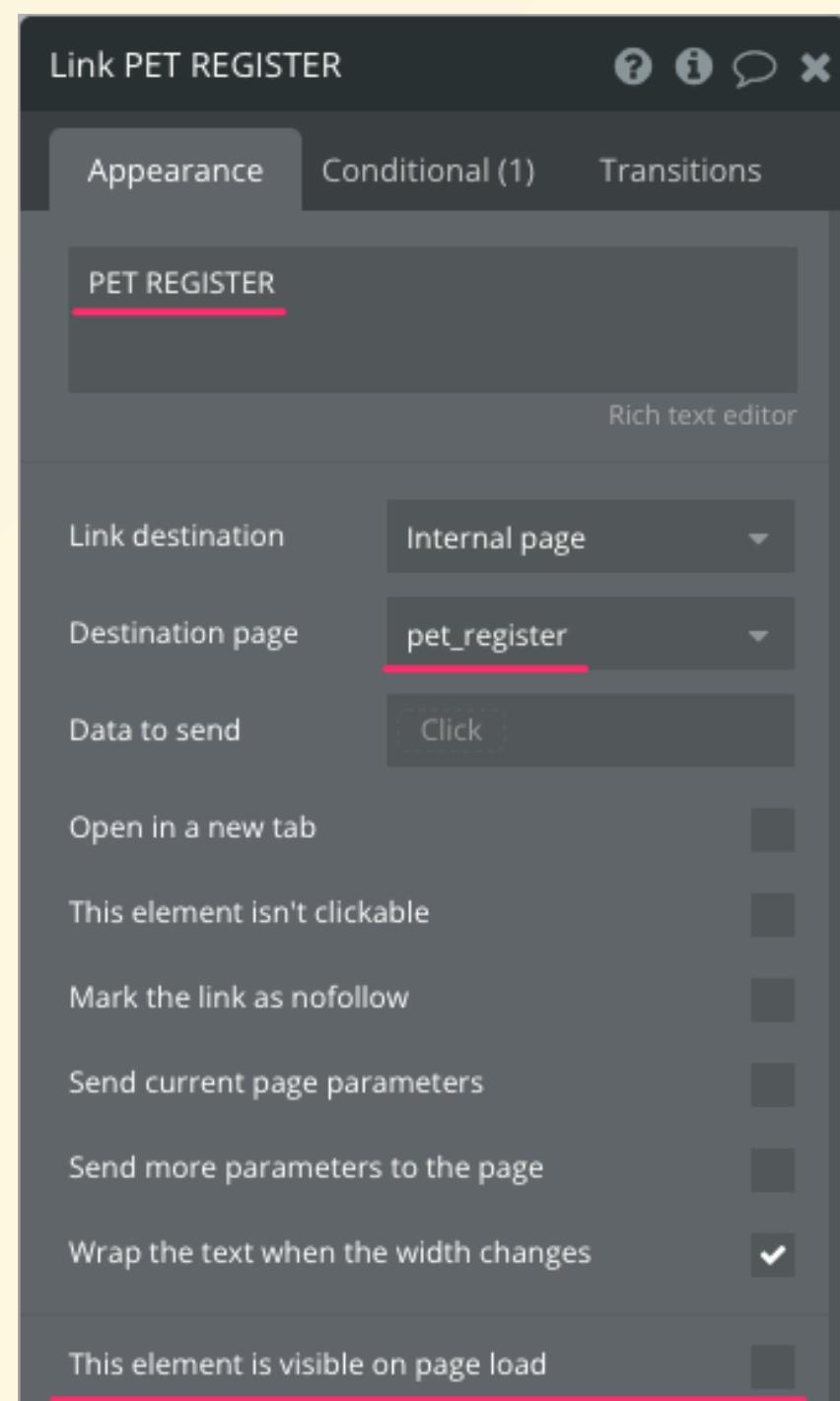
- Then the header elements will be displayed in the right panel, and you can edit them here



- First of all, remove the original **Bubble Forum** and **FAQs** links as they are no longer needed.
- Then select **Link** from the **Visual elements** and drag it to the header area.

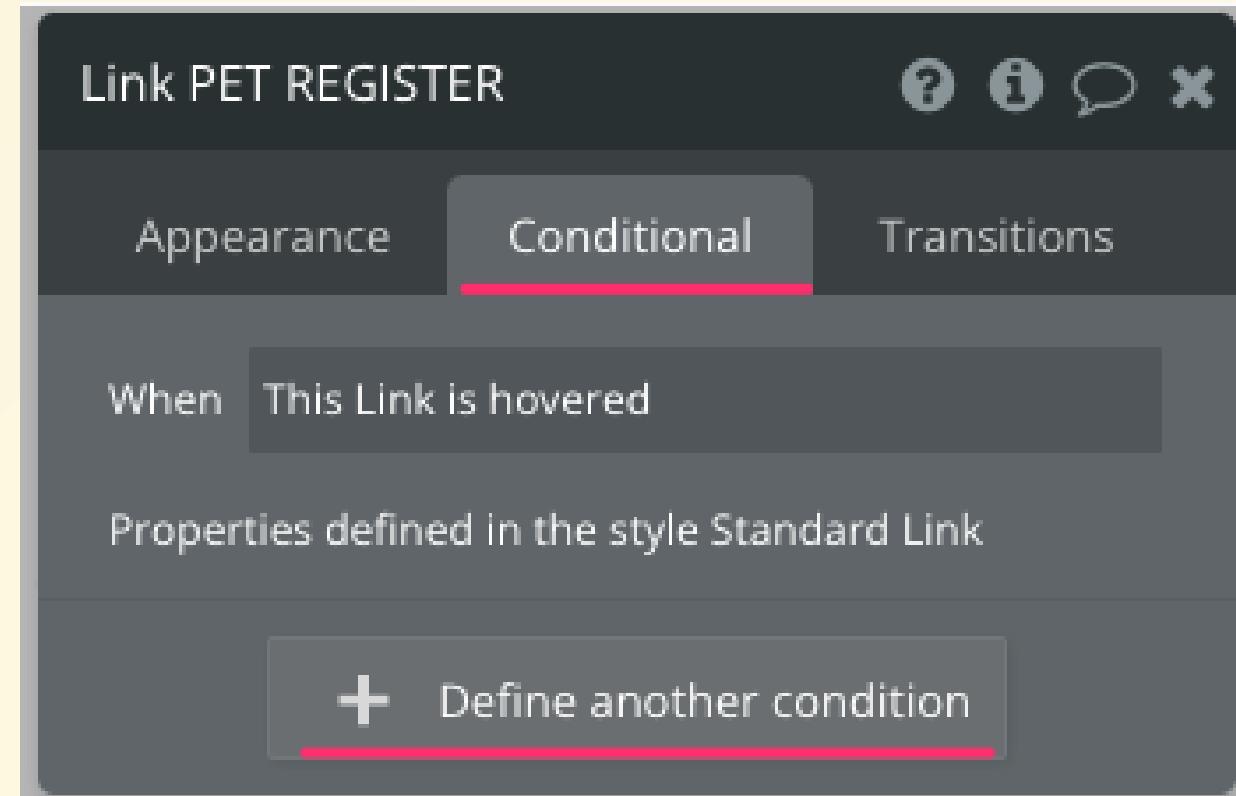


- Name the link "PET REGISTER".
- Select `pet_register` as the "Destination page".
- Uncheck "This element is visible on page load"
  - Uncheck this, and the element will always be hidden.
  - I'll explain why later.



- We want this link to register pets to be displayed only when the user is logged in, so we'll set up the control for that
- From the same dialog, select the **Conditional** tab and click

Define another condition.



- Here, we can define a simple behavior for this link itself.
- In this case, we only want the link to be displayed when the user is logged in, so in the "When" section, we first set the conditions under which we want the link to be displayed

Link PET REGISTER

Appearance Conditional (1) Transitions

When This Link is hovered

Properties defined in the style Standard Link

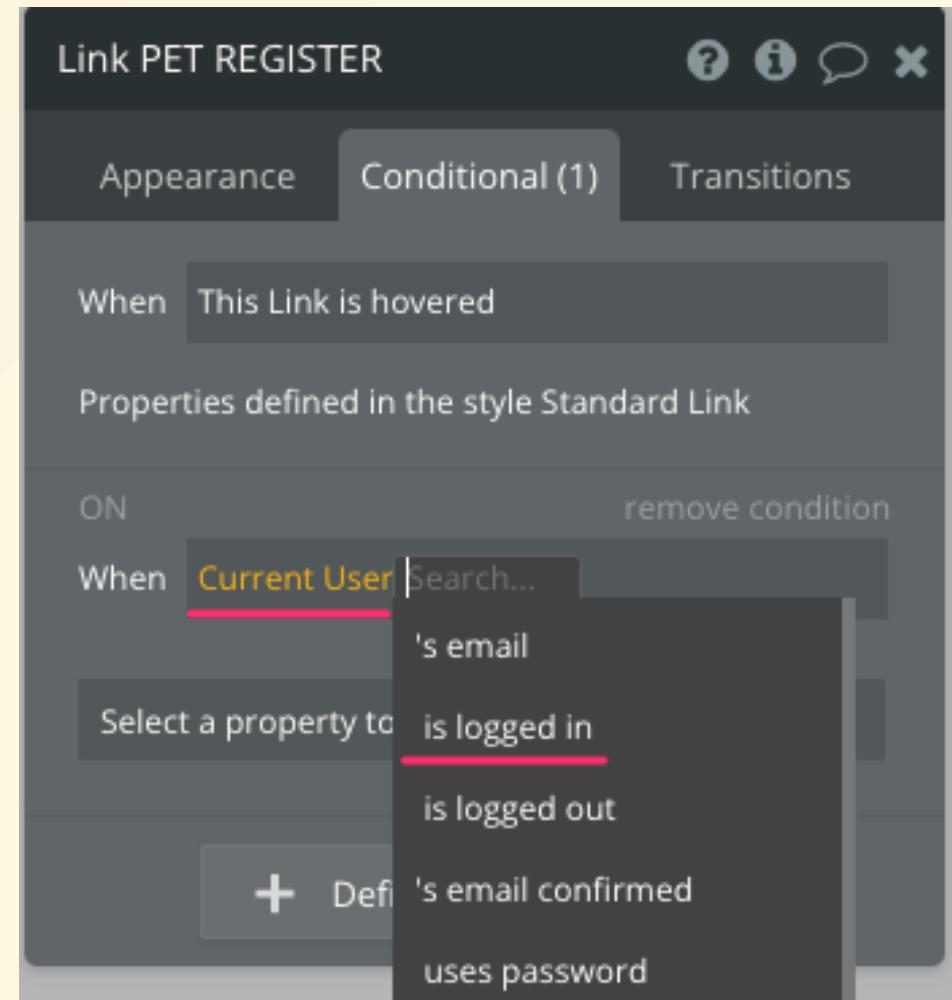
ON remove condition

When Click

Select a property to change when true

+ Define another condition

- Click to select **Current User** and then **is logged in** to specify the condition "if the current user is logged in" !

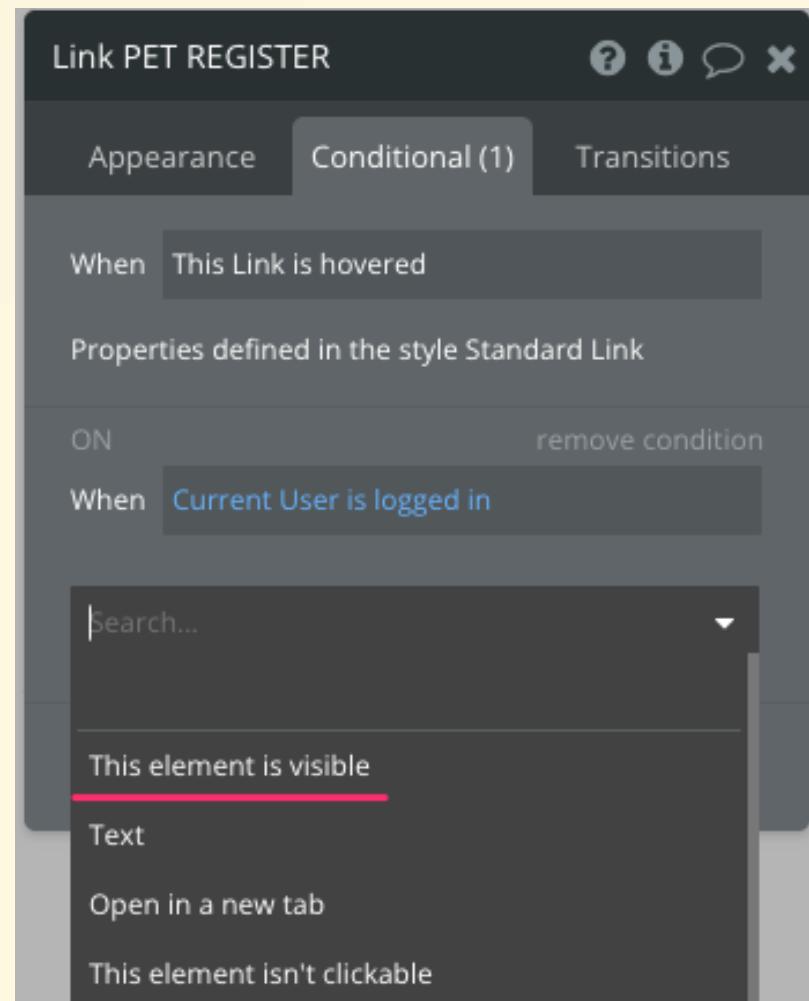


- Next, let's set the behavior when When is satisfied.

- Select

This element is visible  
from the pull-down menu of  
Select a property to  
change when true

- - This will literally make this element visible.
- Once selected, check the behavior



## Now let's preview it!

- Once back to index, run preview
- You should be logged in now, so you should see the link "PET REGISTER".
- From there, click LOG OUT in the upper right corner, and you should see the "PET REGISTER" link disappear.

# Let's create the Pet Detail screen.

- Next, let's create the pet detail screen.
- The key point here is to pass the information of the selected pet in the list screen.

Name

タマ

Image



Weight Logs

Birthday

2021年11月1日

Gender

男の子

[← Back to list](#)

REGISTER

# First, prepare a new detail screen.

- Select "Add a new page..." from the top left. from the top left corner.
- "Page name" is `pet_detail`.
- For "Clone from", choose `pet_register` since the screen structure is similar.

- If we clone the page, it will be the same as the registration page, so we will review it.

削除前

Name

Image

Click to upload pet image

Birthday

Gender

REGISTER

- First of all, we will delete all the input elements that we prepared as input items

削除後

Name

Image

Birthday

Gender

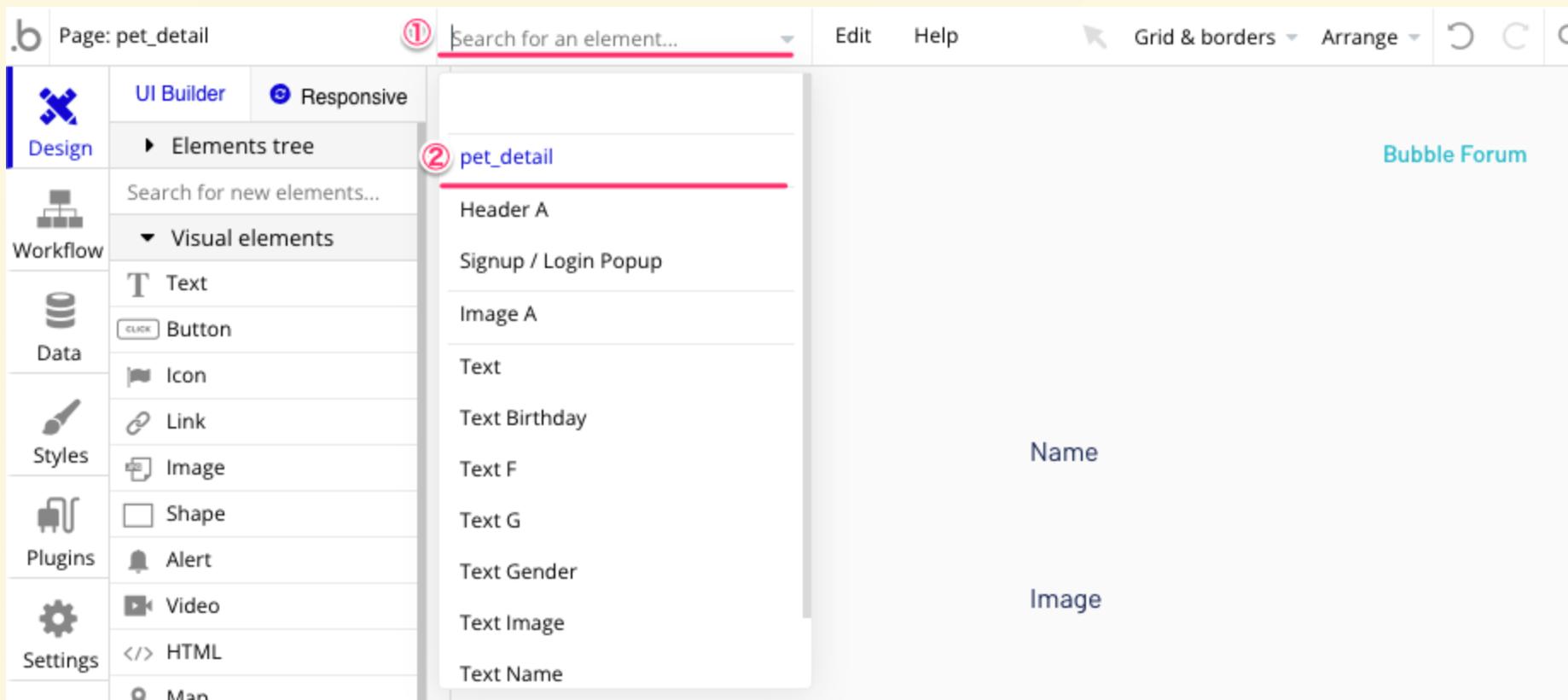
REGISTER

- Next, let's place the elements for display.
- Let's drag each element from "Visual elements".
  - Name: Text
  - Image: Image
  - Birthday: Text
  - Gender: Text
- The width (w) of the elements should all be 220px

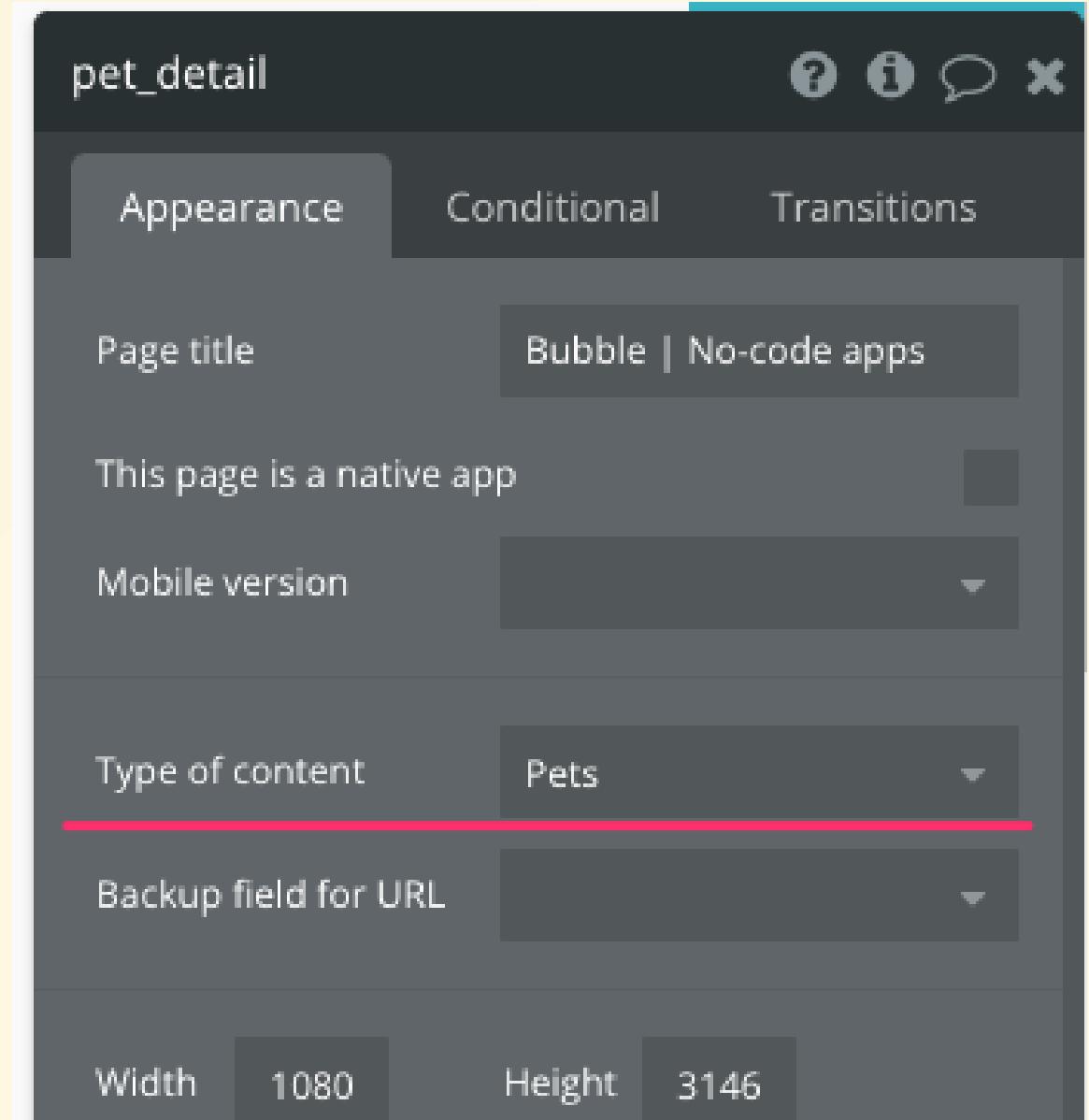
The diagram shows a wireframe layout of a registration form. It consists of a large rectangular container divided into four horizontal sections by red vertical lines. The top section contains a text input field labeled 'Name' with placeholder text '...edit me...'. The second section contains an image input field labeled 'Image' with a blue border and four corner handles. The third section contains a text input field labeled 'Birthday' with placeholder text '...edit me...'. The bottom section contains a text input field labeled 'Gender' with placeholder text '...edit me...'. At the bottom of the form is a teal-colored button with the word 'REGISTER' in white capital letters.

- Once the elements are ready, we will actually define the values to be referenced from the database.
- First of all, let's display the advanced dialog for this screen itself.

- Select `pet_detail` from the upper left part of the right panel to display the dialog.
  - You can also select an element from here, which is useful if you have several overlapping elements!



- From there, you will find a field called **Type of content**, where you can specify **Pets**.
- This way, you can specify what type of database you want to display this screen from, and for each field, you only need to specify which field of that type you want to use



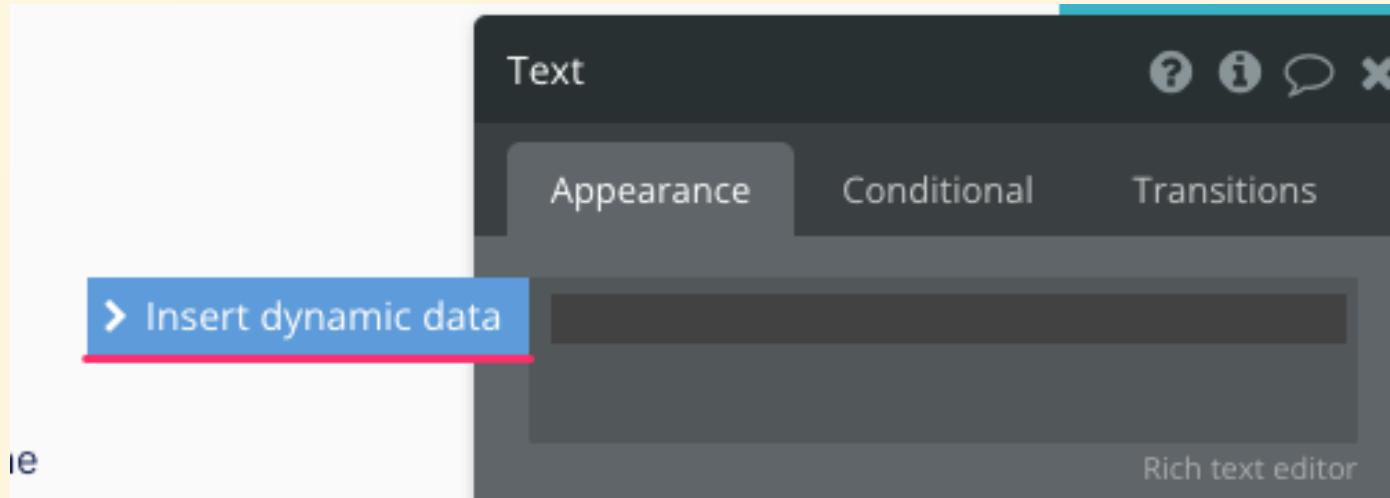
- By the way, you can now specify the type, but you will have to specify the specific pet information (whether it's a pooch or a ball) when you move from the list screen.

- Now, let's connect the display element to the Pets type field.
- Let's start with Name.
- If you want to dynamically display a value retrieved from the database, as in this case, do you remember how to do it?



- That's right! We'll use "Dynamic data!"
- Double-click on the Name element and type  
.... .edit me... Double click on the Name element and click on the

Insert dynamic data` button.

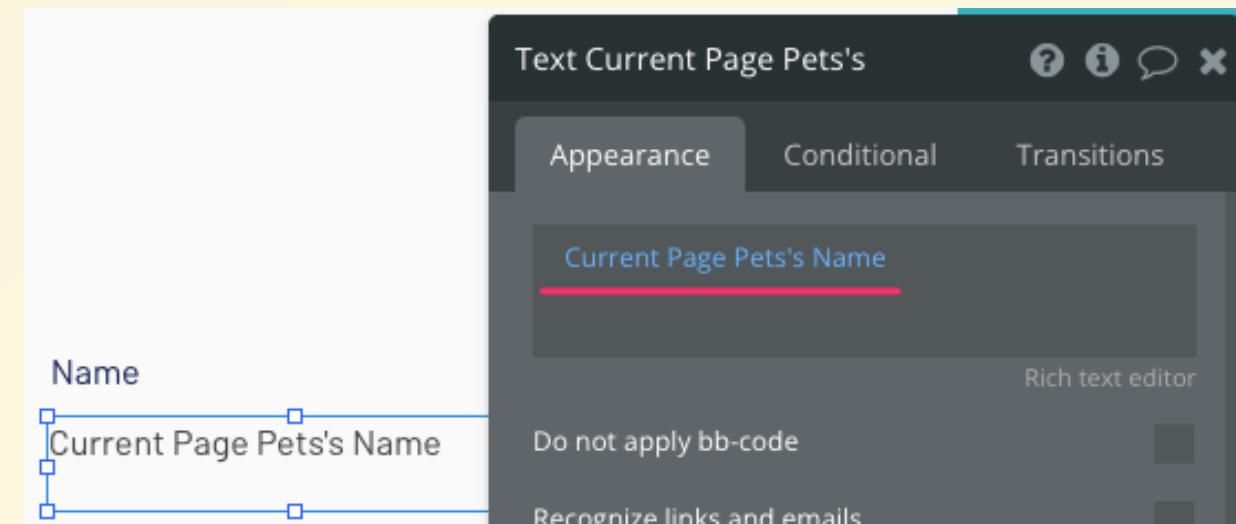


- Then, a pull-down will appear, and click

Current Page Pets's from the pull-down.

- This literally means the information of the pets assigned to the current page

- Then the fields that the Pets type has will be displayed, select 's Name .



- In the same way, let's set the Dynamic data for Image / Birthday / Gender.
- After the settings are done, select all the elements related to the pet details and center them
  - Right-click on the selected element --> `Center horizontally

Name

Current Page Pets's Name

Image

Current Page Pets's Image

Birthday

Current Page Pets's Birthday

Gender

Current Page Pets's Gender

REGISTER

- When you're done, preview the settings.
- The values are not displayed correctly, right?
- This is because you haven't specified which pet's information is displayed in the list screen yet. This is because you haven't specified which pet's information you want to see in the list screen yet.
  - The **Lorem ipsum** is a

Name  
Lorem ipsum...

Image

Birthday  
Nov 10, 2021 10:15 pm

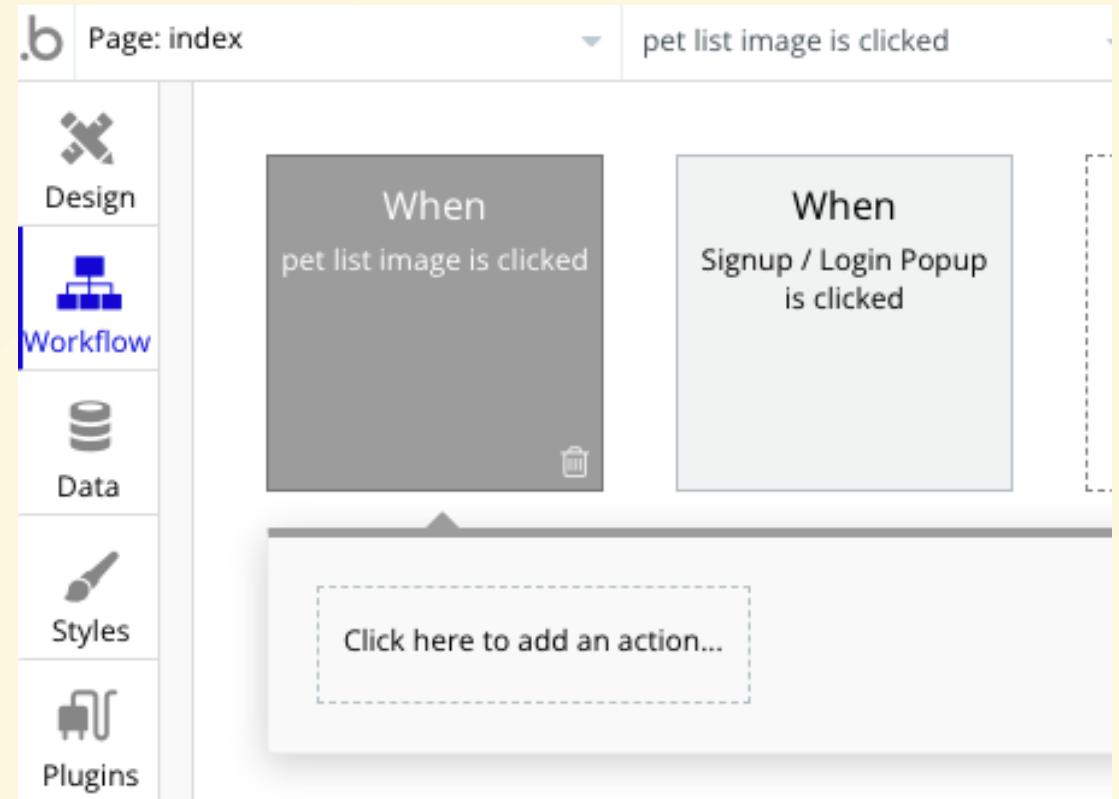
Gender  
Lorem ipsum...

REGISTER

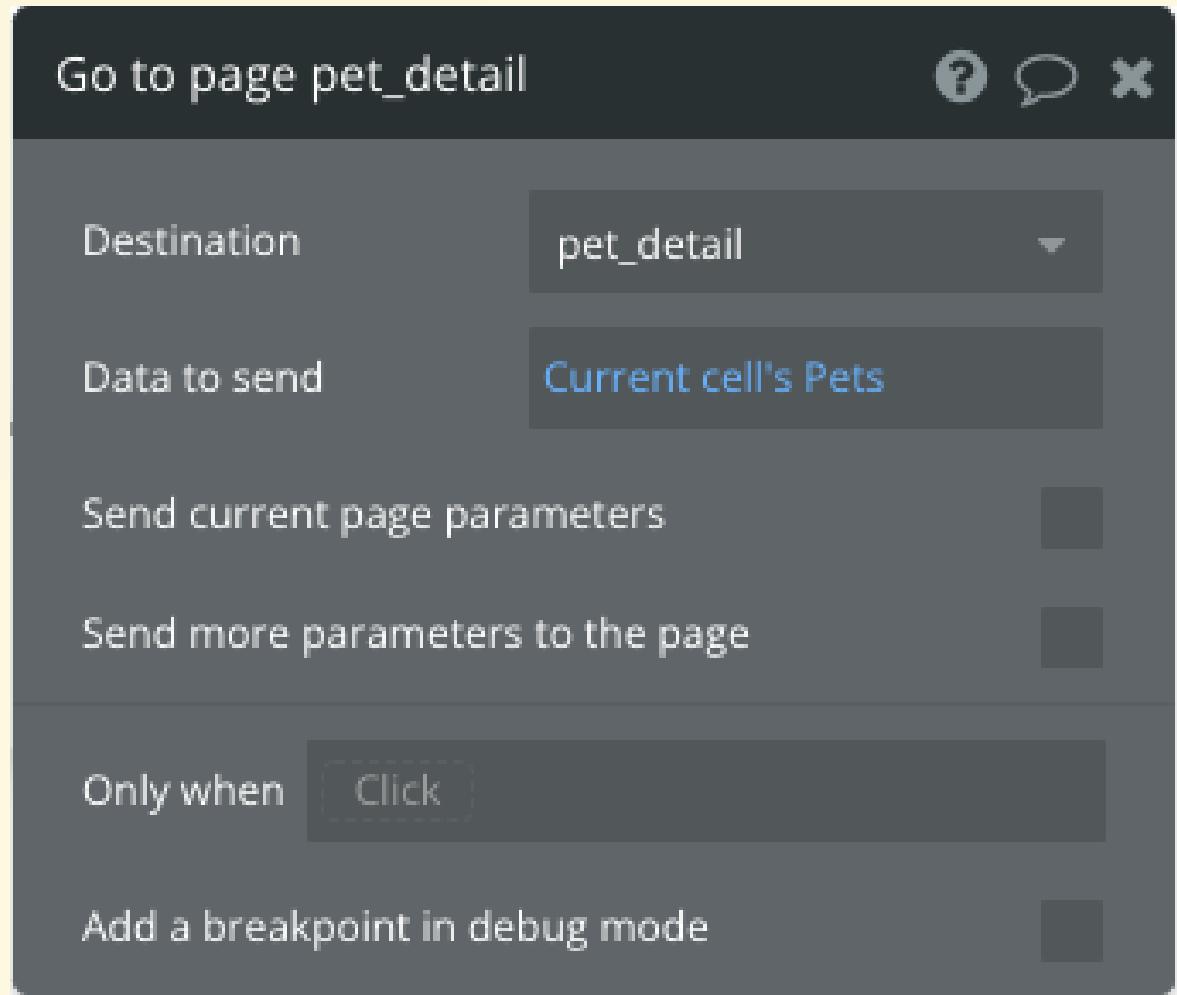
# Now let's connect the list page to the detail page.

- Switch to the index page from the menu on the top left.
- When you click on the image of a pet in the "Pet List" screen, you will be taken to the "Details" screen of the pet.
- So, let's set up a workflow for the pet image in the list screen

- Double-click the pet image and click **Start/Edit workflow**.
- You should now be in the state before setting the behavior for the "When pet list image is clicked" box



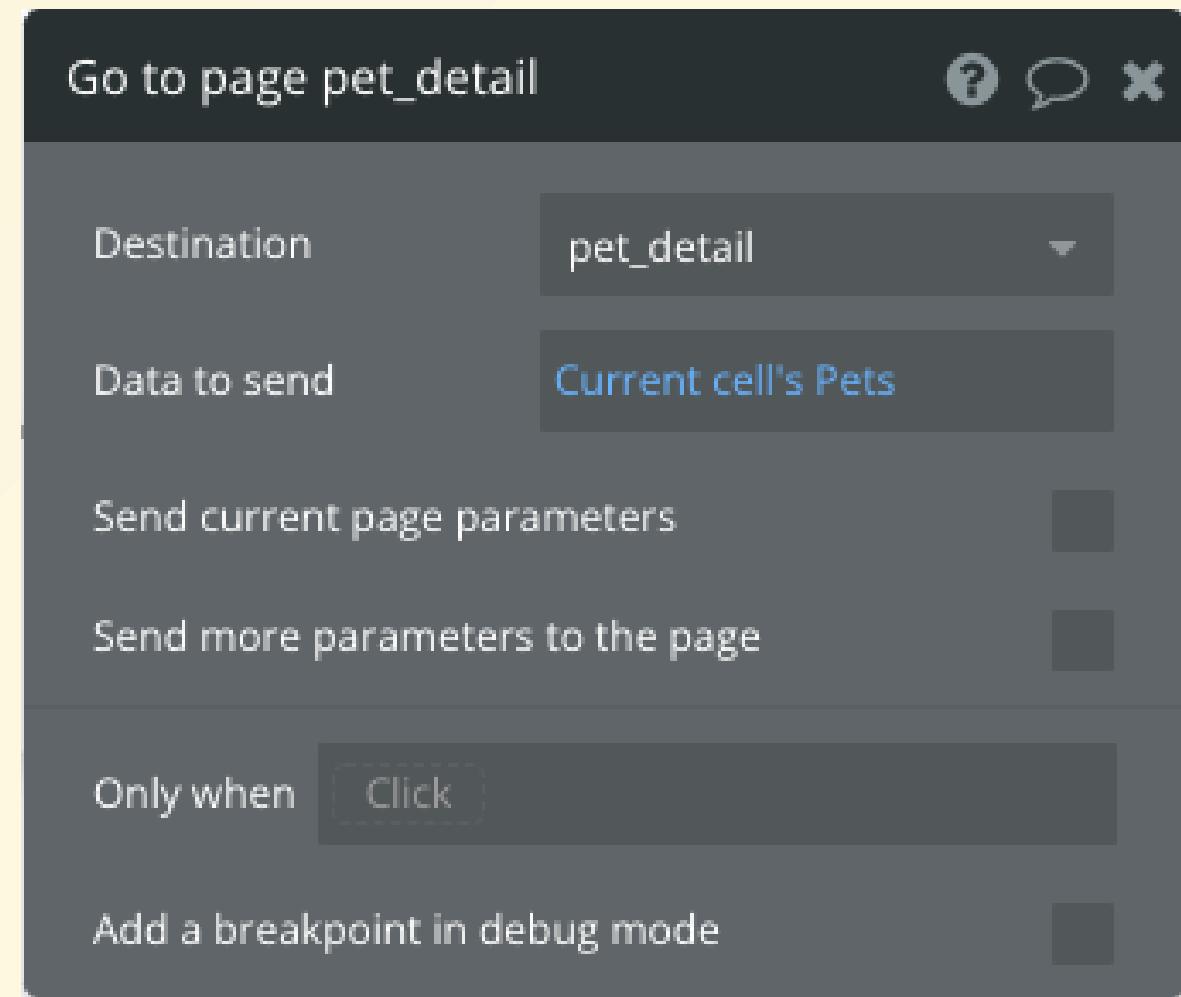
- So from "Click here to add an action", select Navigation -> Go to page
- A dialog will appear, and **Destination** should be `pet_detail`.



- And specify

Current cell's Pets for  
Data to send.

- Now you can specify that the pet information of the current cell will be sent to the destination when moving to the pet detail page.



- Now let's preview it!
- Did you see that when you select a pet image in the list screen, the detail screen of that pet will be displayed?

# Appendix

- Display advice
- The display of Birthday and Gender is a bit bland, so let's change the display format

Name

まめ助

Image



Birthday

Nov 11, 2021 12:00 am

Gender

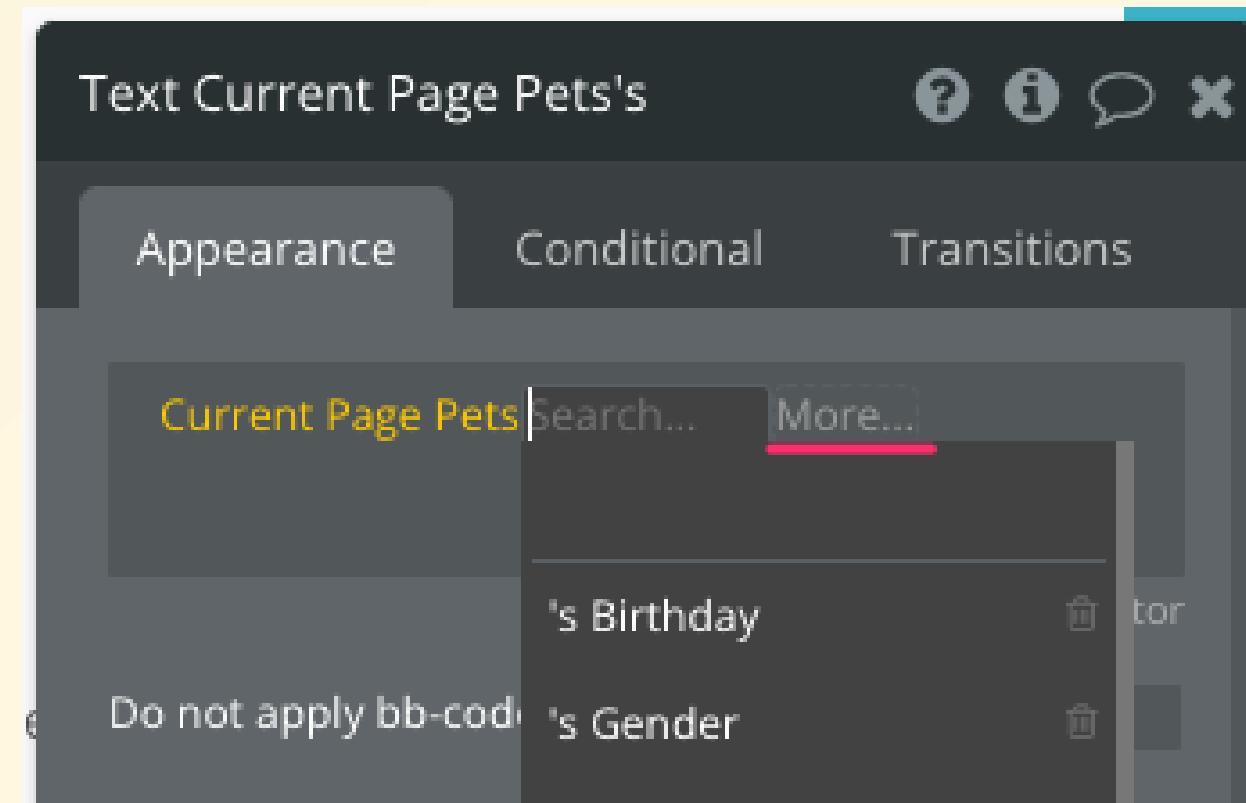
Male

REGISTER

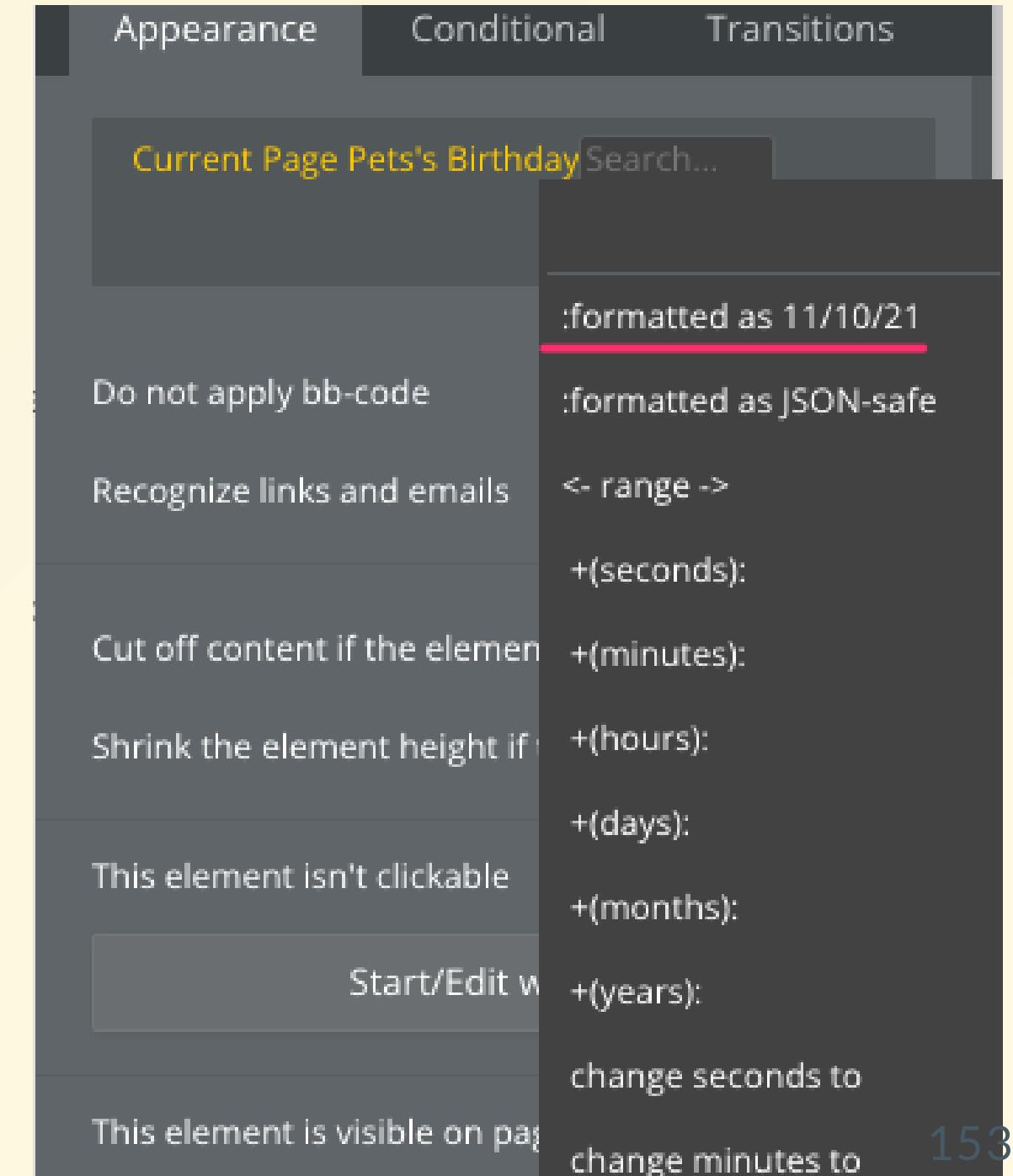
# Appendix1: Change the birthday format to Japanese style.

- Open the pet\_detail screen in the Design tab.
- Double-click on the Birthday display element to open the advanced settings dialog.

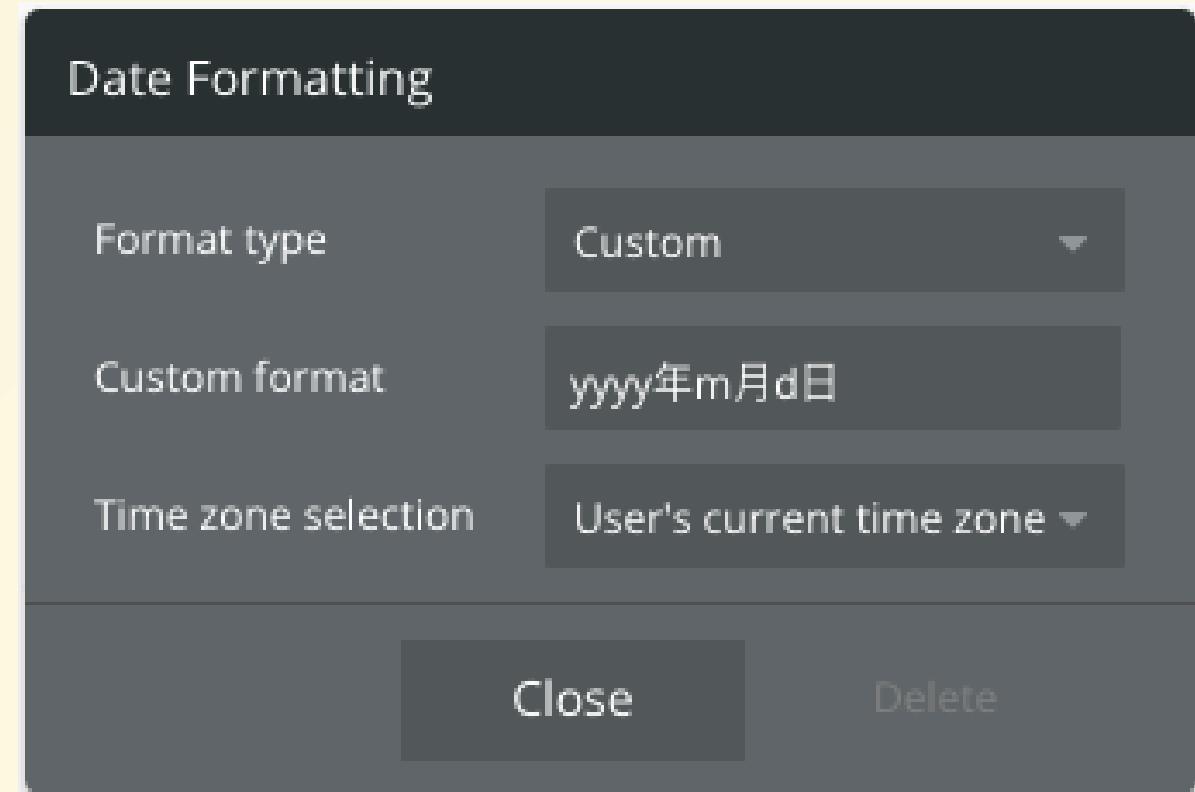
- Then, click the **Current Page Pets's Birthday** section, and you will see a **More** section behind it.



- Then you can further specify the display format of the Birthday value
- This time, click on :formatted as DD/MM/YY at the top
  - DD/MM/YY will contain today's date



- Then you will see the Date Formatting dialog next to it
- Here you can specify the format in detail.
- This time, select Custom for Format type and yyyy year m month d day for Custom format.



- Now let's preview it.
  - Open the index page once and run the preview
- If you open the pet details from the pet list, you will see that the Birthday date is in the format you specified

Name

まめ助

Image



Birthday

2021年11月11日

Gender

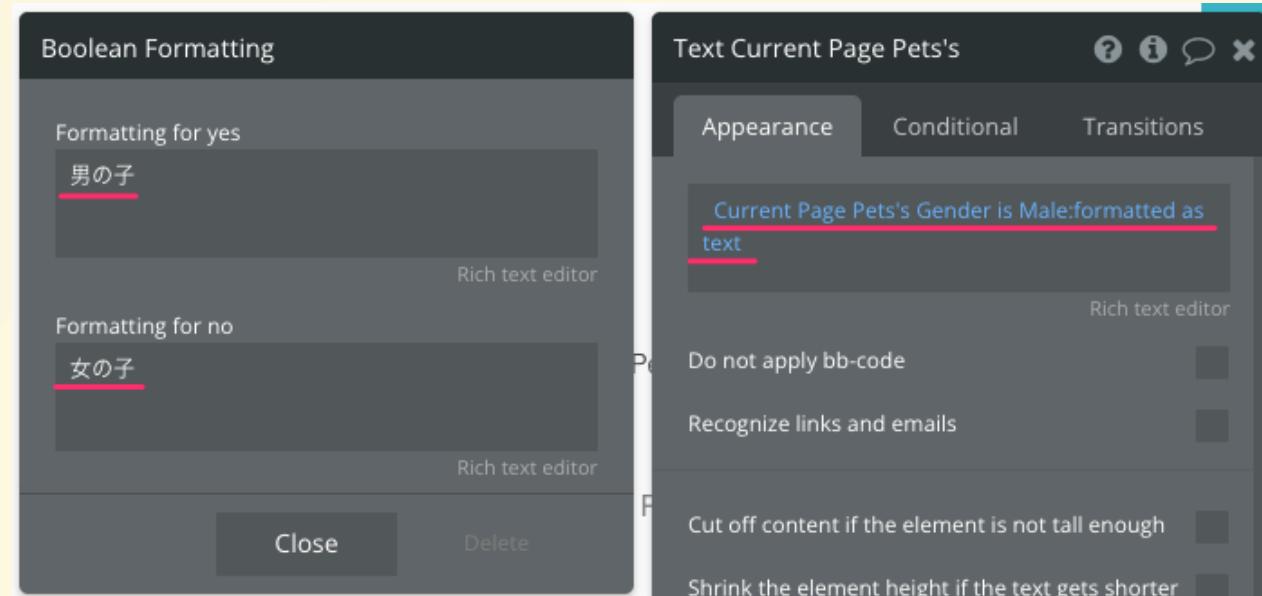
Male

REGISTER

## Appendix2: Let's change the gender label.

- Open the pet\_detail screen in the Design tab.
- Double-click on the Gender display element to open the Advanced Settings dialog box.
- In the same way as before, click the Gender part of **Current Page Pets's Gender**, click **More**, and then enter the following

- Select `is`, type `Male` manually and click enter, then select `:formatted as text`.
- This will bring up a dialog called `Boolean Formatting` where you can enter the following
  - Formatting for yes: Boy
  - Formatting for no: Girl



- This would write the following process

If the current pet's Gender is Male, display "Boy", otherwise display "Girl".

Now there's a little programming involved!

- Now let's run the preview from the index page
- If you open the pet details from the pet list, you will see that the gender values are "boy" and "girl".

Name

まめ助

Image



Birthday

2021年11月11日

Gender

男の子

REGISTER

## **Appendix3: Let's set up a line from the detail screen to the list screen.**

- Let's prepare a line to return to the list from the detail screen.
- I'm sure you've got an idea of what I'm talking about, so I'll just give you a hint.

# Hints

- The element to use is **Link** of **Visual elements**.
- In the case of the Link element, the screen transition is not Workflow, but the element itself can be set to "Destination page".
- Finished image

Name

まめ助

Image



Birthday

2021年11月11日

Gender

男の子

[← Back to list](#)

REGISTER

# Keep track of your pet's weight

- Next, let's take a look at managing your pet's weight!



Weight

pet weight

← Back to detail

ADD

- First, let's prepare the Type.
- From the Data tab, enter `PetWeightLogs` as the New type and click Create.
- Next, we need to set up the field.

- The weight of the pet entered from the screen
  - WeightKg: number
- Connect the weight of the pet to the field
  - pet: Pets

The screenshot shows a software interface for managing data types. On the left, there's a vertical sidebar with icons for Design, Workflow, Data (which is selected), Styles, Plugins, and Settings. The main area has tabs for Data types, Privacy, App data, Option sets, and File manager. Under 'Data types', there's a section for 'Custom data types'. A table lists existing data types: 'Pets' (Publicly visible), 'PetWeightLogs' (Publicly visible, currently selected and highlighted in red), and 'User' (Privacy rules applied). To the right, under 'Fields for type PetWeightLogs', there are several fields: 'Type name' (PetWeightLogs), 'pet' (with a dropdown menu showing 'Pets'), 'WeightKg' (with a dropdown menu showing 'number'), 'Creator' (User), 'Modified Date' (date), and a 'default' field containing a placeholder '1'. There are also status indicators at the top right, including a gift icon, a warning icon with '16 issues', and other navigation links like Edit, Help, Saved, and Development.

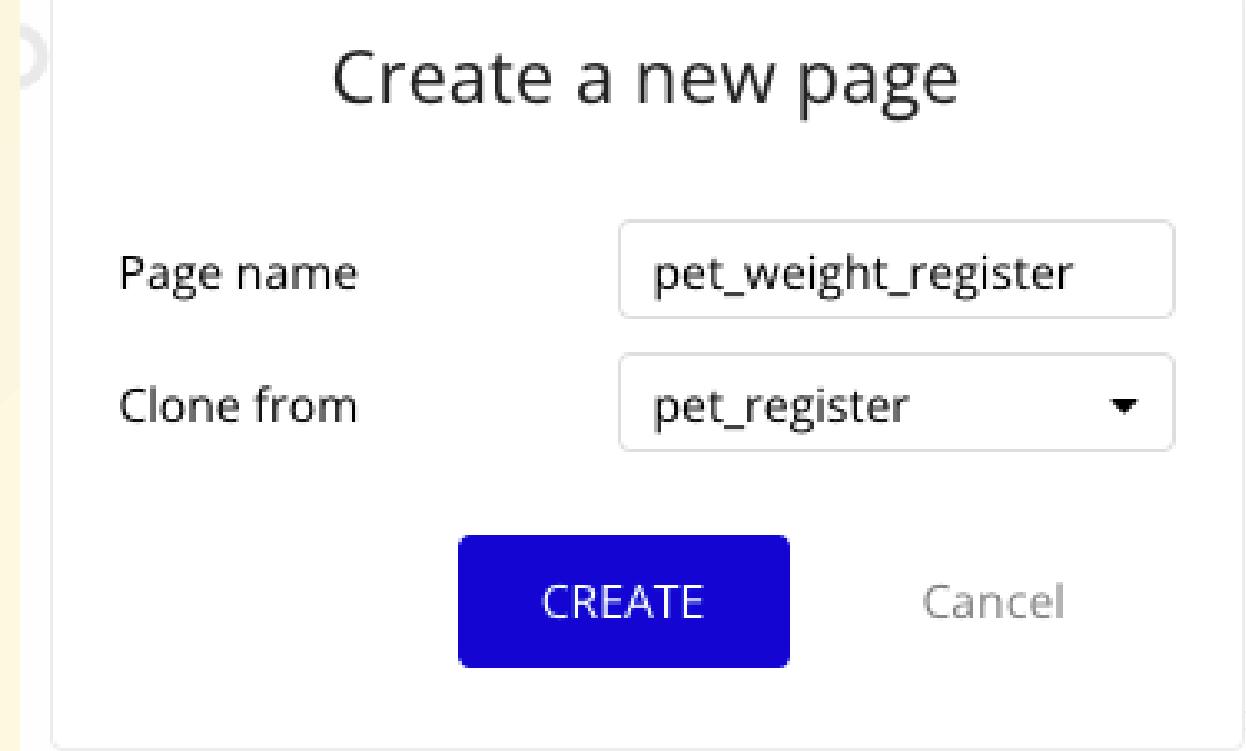
- Now that the database is ready, let's set up the screens.
- First, let's create a new page to manage the weight of pets.
- From Add a new page, select the page name `pet_weight_register` and the source is `pet_register`, then CREATE

### Create a new page

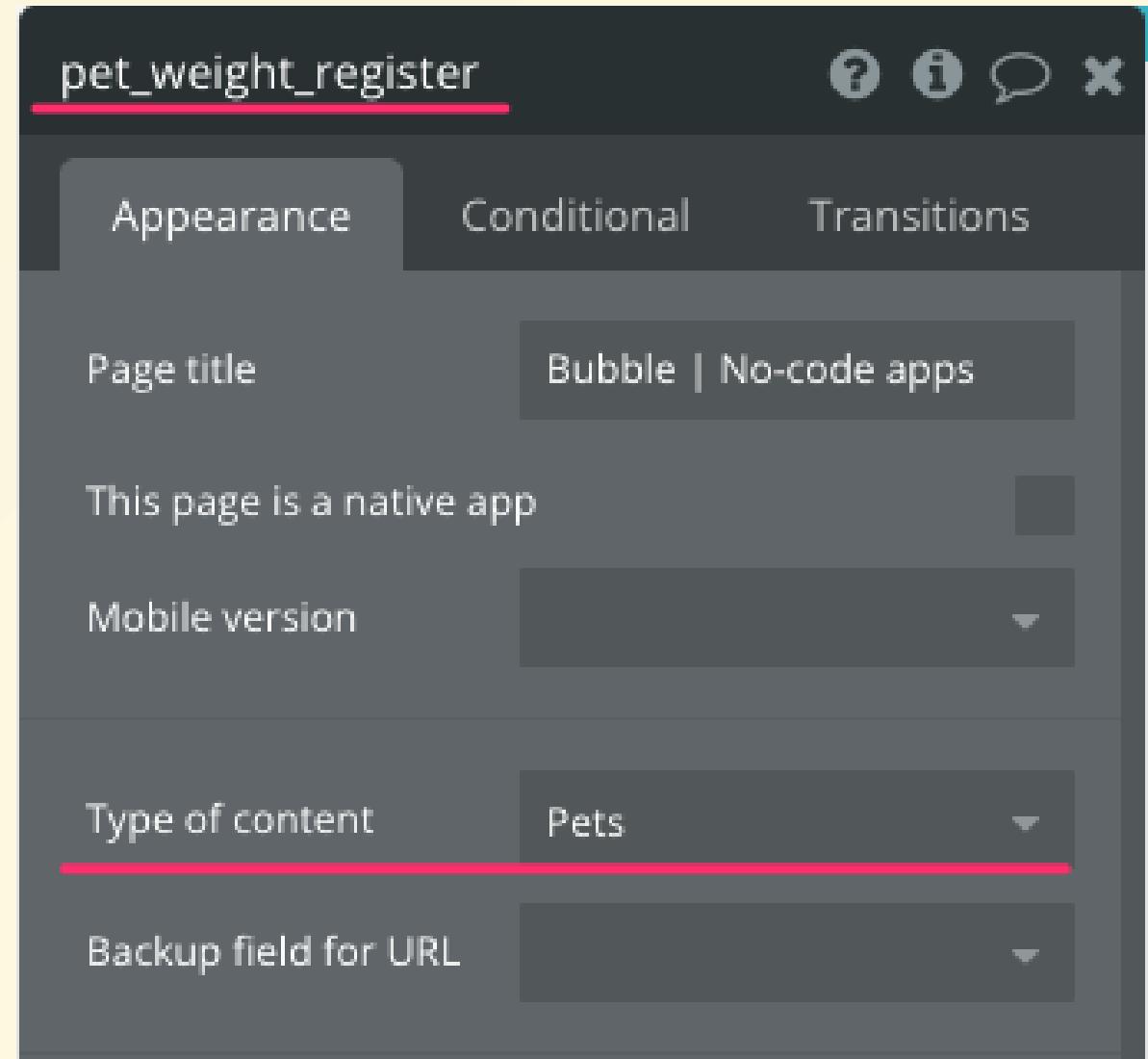
Page name

Clone from  ▾

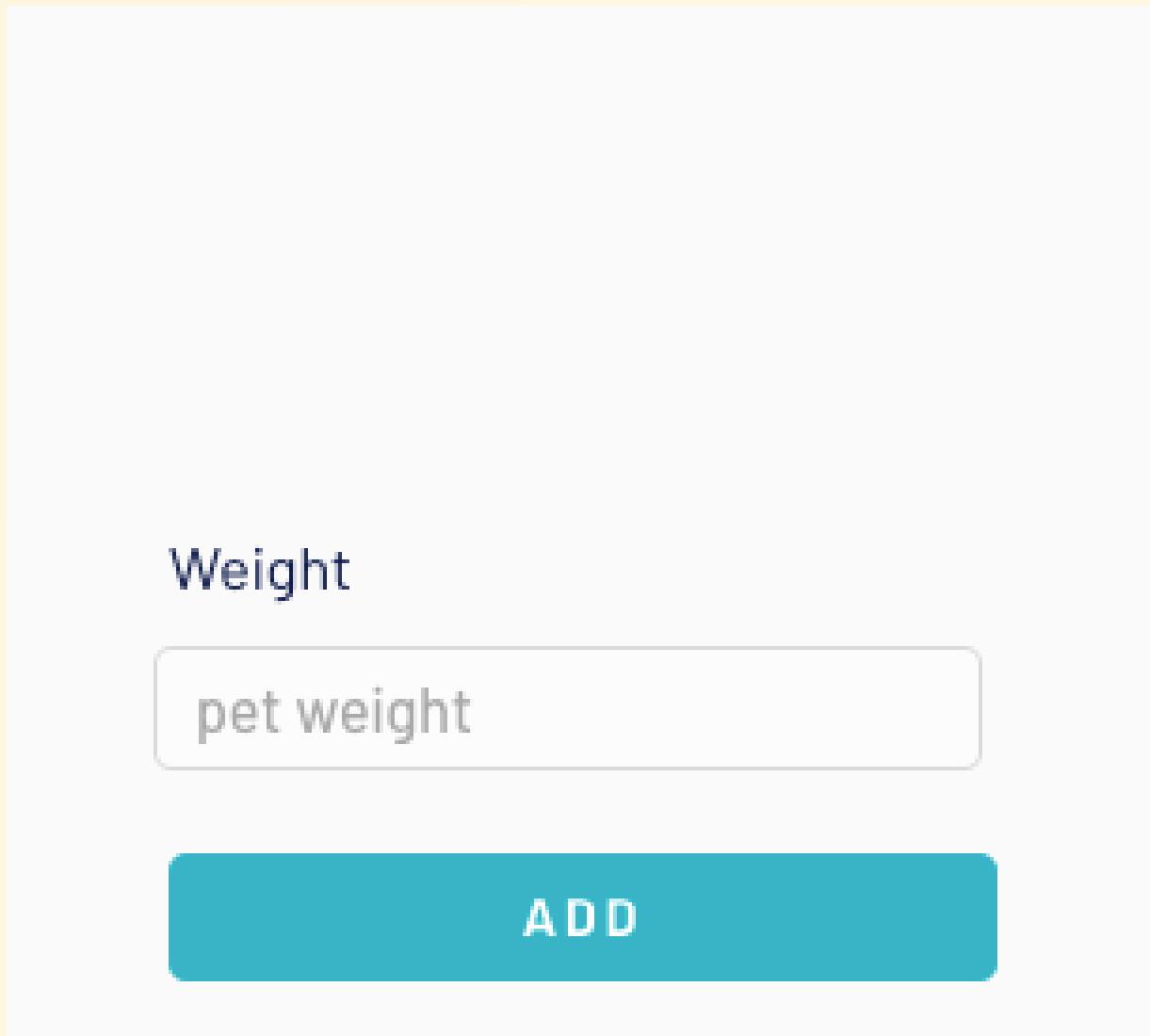
CREATE Cancel



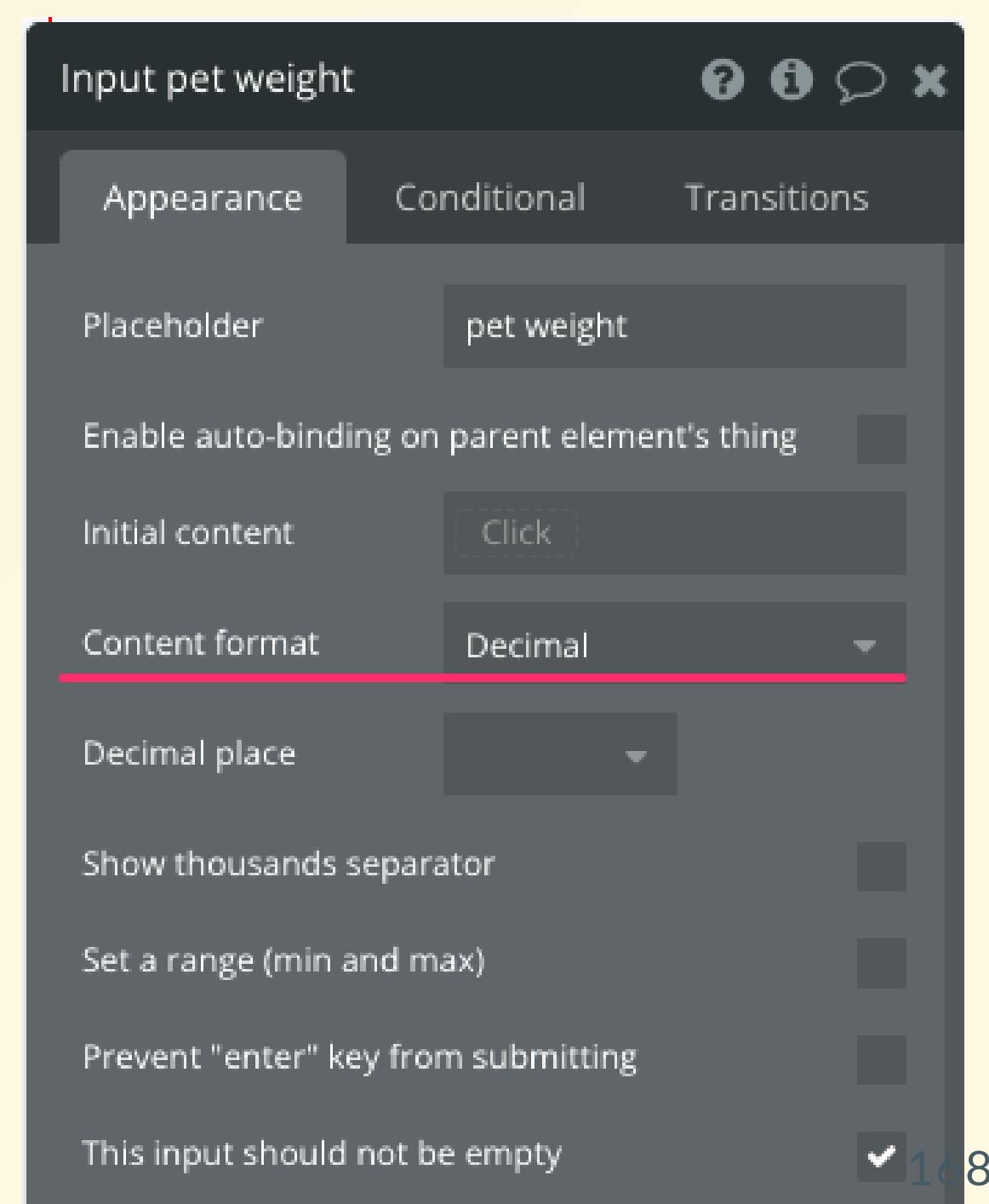
- Since this screen is also supposed to be transitioned from the pet details, specify Pets as the Type of content in the advanced settings dialog for the pet\_weight\_register screen



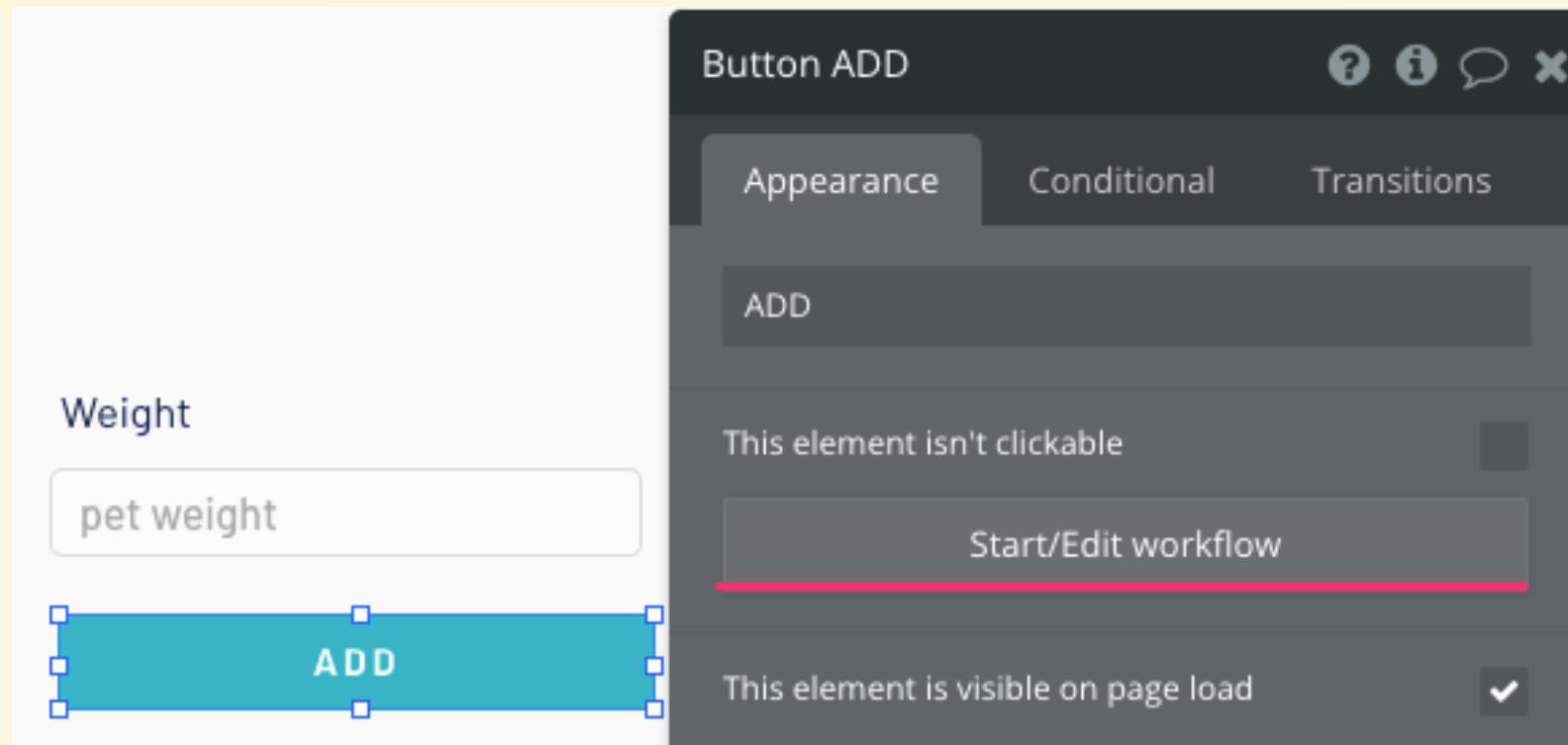
- Since the elements from the original pet registration screen are still there, delete the unnecessary ones and place the elements as shown in the reference image.
- However, the graph drawing area will not be placed at this time.



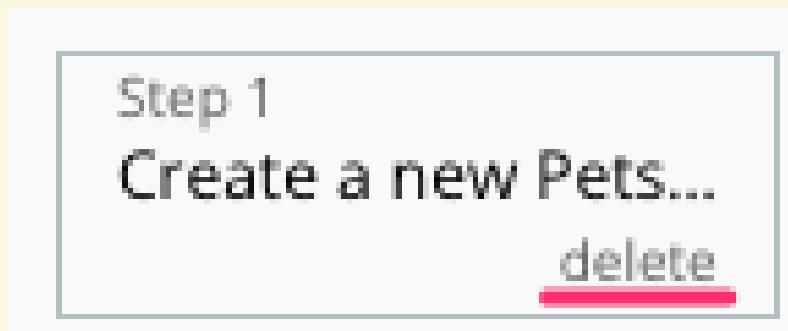
- At this time, we want to allow only numbers with a decimal point to be entered for the weight input element, so we will just set that
- Double-click on the pet weight input element and select **Decimal** for **Content format**.
  - Note that **Integer** is for integer input without decimal point.



- Next, we will set up the workflow for the Save Weight button click
- Double-click on the ADD button, and select Start/Edit workflow from the Advanced Settings dialog



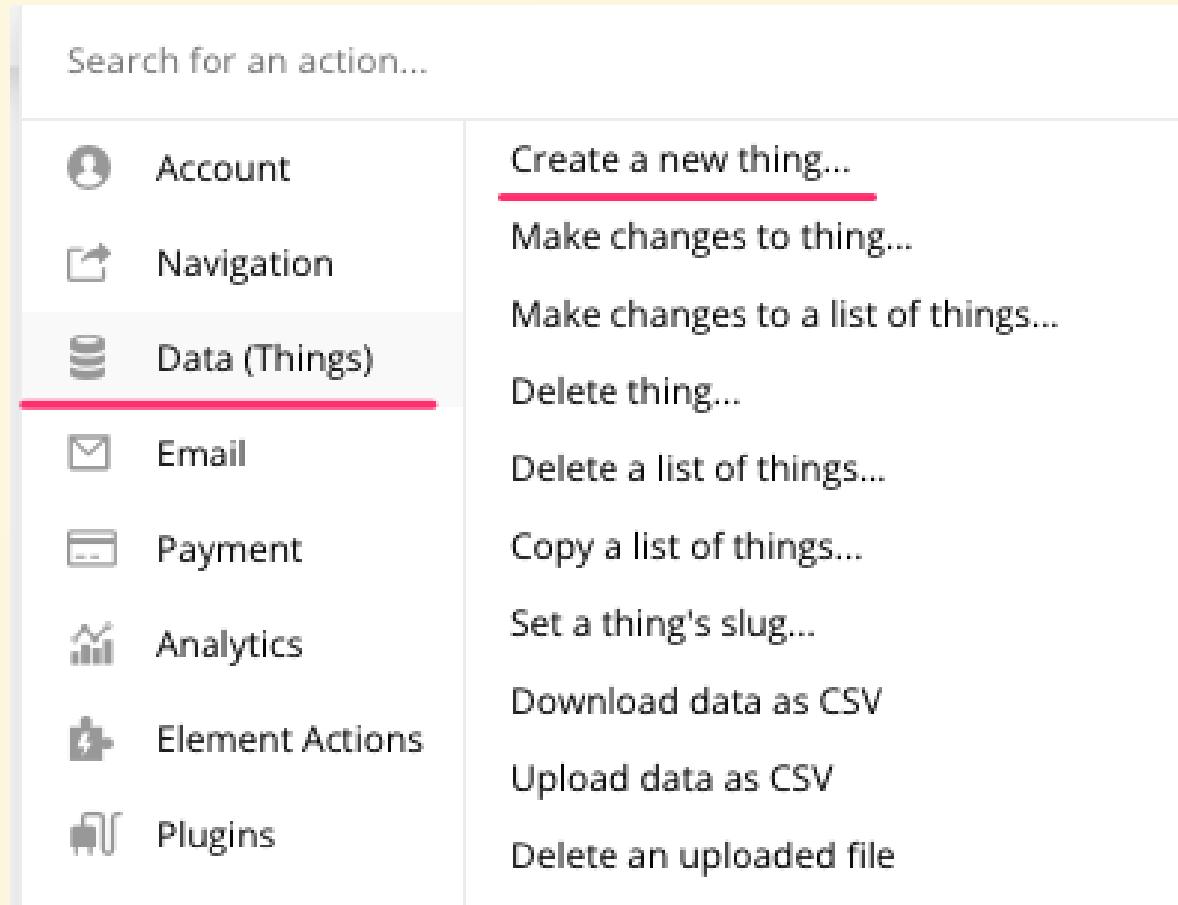
- Then you can see that some behaviors are already set for `When Button ADD is clicked
- This is because we copied `pet_register` when we created the page, so the workflow settings are inherited from that time.
- Since we will be defining new behaviors, we will delete the existing workflow.
- If you select the behavior of Step1 / Step2, you will see `delete` in the bottom right corner, so you can delete the behavior from there



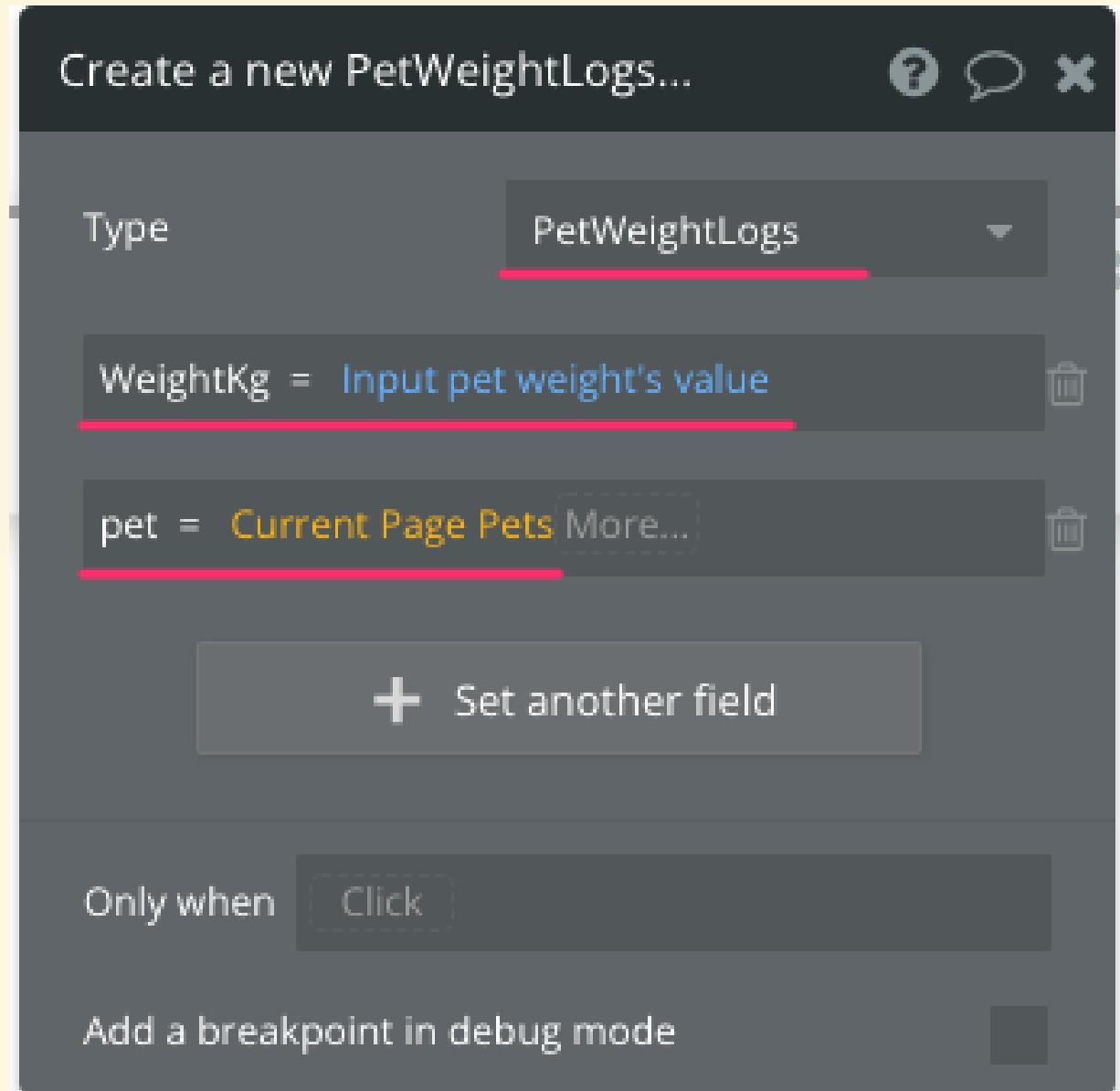
- Now let's set up the workflow.
- The first thing we want to do is to save the weight values entered on the screen to the PetWeightLogs that we prepared earlier.
- Please set this up based on what we have learned so far!
- I've included an explanation on the next page, so let's get started!



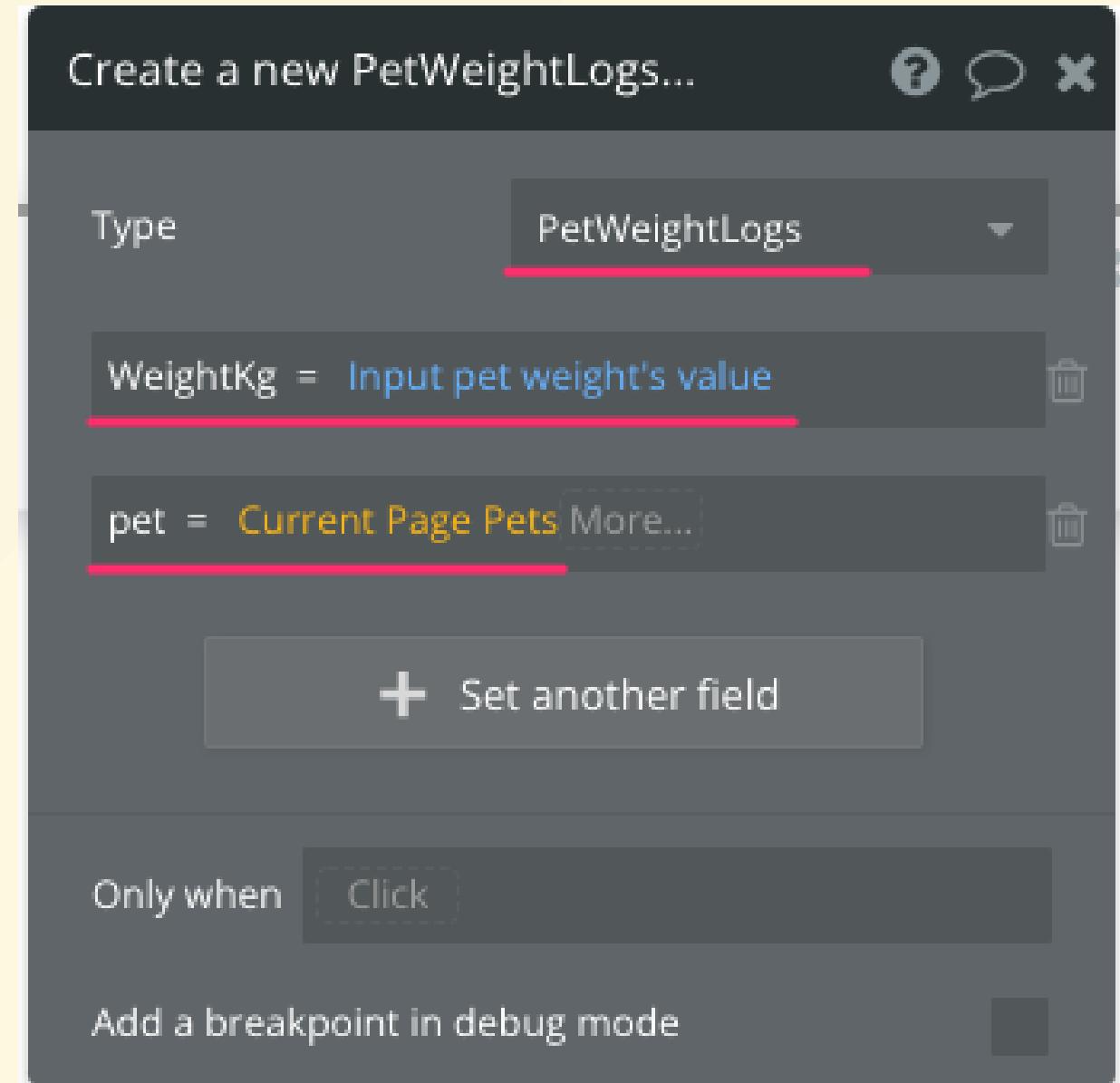
- The first action to set is Data(Thing) --> Create a new thing....



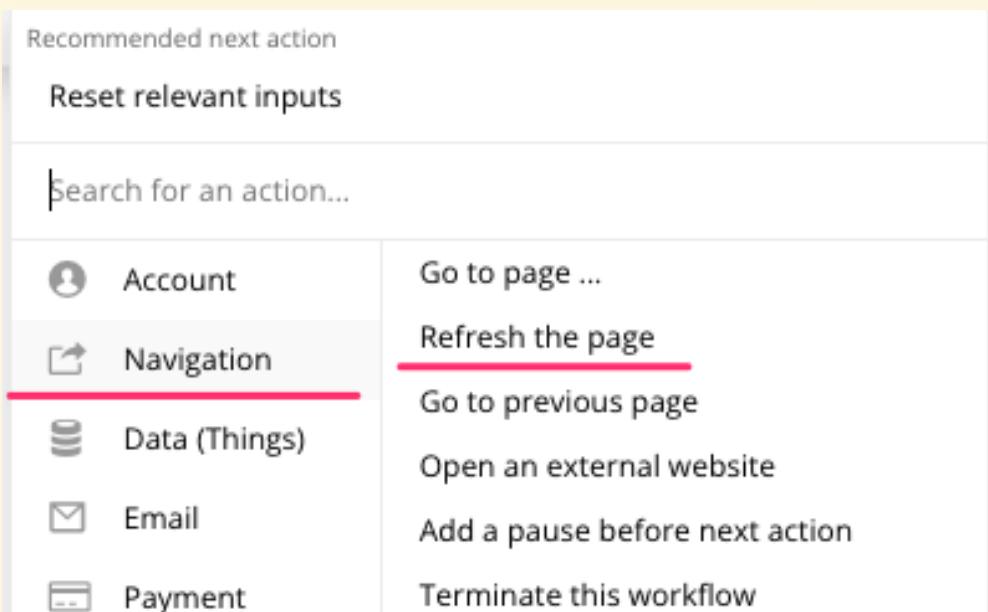
- Then, specify the type and field to be registered
- Since we are registering the weight of pets, the Type is PetWeightLogs .



- So the fields to set are `WeightKg` and `pet`.
- If you don't set the `pet` field, you won't be able to determine which pet's weight it is, so specify it!
- If you don't set the `pet` field, you won't be able to determine the weight of the pet.



- Since we want to redraw the screen after registration to update the graph to be displayed later, let's define the behavior to redraw the screen after registration
- The action for screen manipulation was **Navigation**.
- If you look at the sub-menu, you will see that **Refresh this page** is the right action, so let's select it



- Now that you've registered your pet's weight, it's time to prepare a link from the details screen.
- Let's open the detail page and create a link between Image and Birthday.
- Let's use **Link of Visual elements** for the link.

- There are two points. 1.
  1. set the destination to **Destination page** 2.
  2. give the pet information of the current page to the link destination
- The following page shows you how to set it up, so you can try it yourself.

- Like this!

The screenshot shows a 'Link Weight Logs' dialog box from a web editor. The dialog has tabs for Appearance, Conditional, and Transitions, with Appearance selected. The title 'Weight Logs' is displayed in a rich text editor. Below the editor are three settings: 'Link destination' set to 'Internal page', 'Destination page' set to 'pet\_weight\_register', and 'Data to send' set to 'Current Page Pets'. A red vertical line and the text '240px' are overlaid on the left side of the dialog, indicating the width of the 'Weight Logs' element.

Link Weight Logs

Appearance Conditional Transitions

Weight Logs

Rich text editor

Link destination Internal page

Destination page pet\_weight\_register

Data to send Current Page Pets

Weight Logs

Birthday

Current Page Pet's Birthday: f...

Gender

Current Page Pet's Gender is...

240px

- Once you've created the link from the detail screen to the weight management, you'll need to prepare the reverse link as well!
- The idea is the same as the link in the detail screen.

- After setting up, let's go back to the list screen and run the preview.
- After moving to the list, detail, and weight management screens, enter a value in the Weight field and press ADD to redraw the screen, right?

Weight

pet weight

← Back to detail

ADD

- Just to be sure, let's check that the data is registered in the database
- If you select **All PetWeightLogs** from the App data tab of Data, you should see the data registered on the left side, right?

The screenshot shows the Bubble app interface. On the left, there is a vertical sidebar with icons for Design, Workflow, Data (which is selected), Styles, Plugins, and Settings. The main area has a header with 'Page: pet\_weight\_register' and navigation links for Edit, Help, Saved, and a yellow gift icon. To the right of the gift icon are buttons for '8 issues', a refresh arrow, a circular arrow, a search icon, Development mode, and Preview.

The main content area has tabs for Data types, Privacy, App data (which is selected), Option sets, and File manager. Below these tabs is a section for Database views with buttons for New view, Primary fields, Search, and a search bar showing '2 entries (displaying 2)'. There are also buttons for New entry, Delete (0), Upload, Modify, Export, and Bulk operations. To the right of these buttons are links for 'Copy and restore database' and 'Switch to live database'.

The central part of the screen shows a table titled 'Application data - All PetWeightLogs - Development version'. The table has columns for pet (with a search icon), WeightKg, Created Date, Modified Date, Slug, and Created By. Two rows of data are listed:

pet	WeightKg	Created Date	Modified Date	Slug	Created By
1636457887522x519076068	13.1	Nov 11, 2021 9:17 am	Nov 11, 2021 9:17 am		kyogoku+bubble_test@g
1636457887522x519076068	12.3	Nov 11, 2021 9:13 am	Nov 11, 2021 9:13 am		kyogoku+bubble_test@g

Below the table, there are buttons for All Pets and All PetWeightLogs, each with edit and delete icons. A red box highlights the table containing the two entries.

# Let's display the pet's weight graphically.

- Let's display the weight of the registered pets graphically as we did in Adalo.



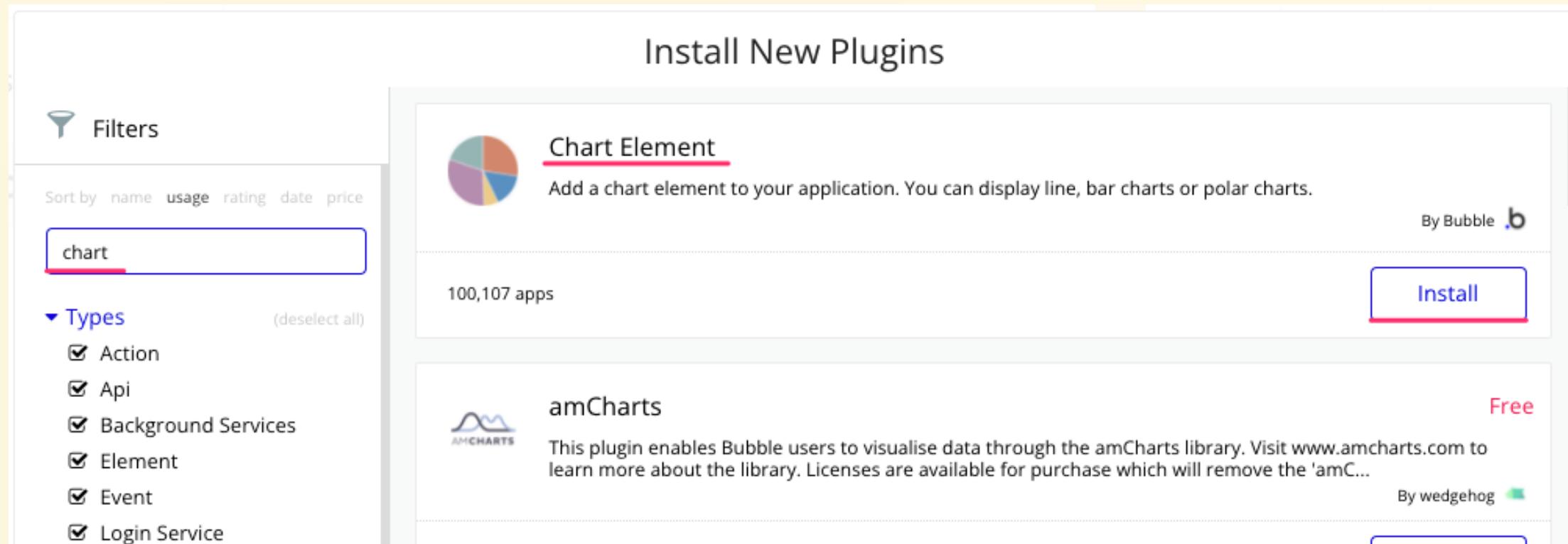
# First, we need to install a plugin.

- Bubble doesn't come with graphing functionality by default, so we'll need to add it in the form of a plugin.
- In Adalo, we used Component, but in Bubble, we'll use Plugins.

- Select Plugins from the left menu to display the Plugins screen.
- Then press **+ Add plugins** in the upper right corner to search for plugins to add

The screenshot shows the Bubble Plugins screen. On the left, there is a vertical navigation bar with icons for Design, Workflow, Data, Styles, Plugins (which is the selected tab), and Settings. The main area is titled "Installed Plugins" and shows a single item: "Bubble Plugin". Below it, there is a larger section titled "Bubble Plugins" with a description: "This plugin returns the list of all the plugins Bubble integrates with." At the bottom of this section, there are two buttons: "Plugin content" and "Data calls". In the top right corner of the main area, there is a blue button with a white plus sign and the text "+ Add plugins". The top navigation bar includes "Edit", "Help", "Saved", a yellow gift icon, a red triangle icon with "8 issues", and links for "Development", "Preview", and a user profile.

- When the **Install New Plugins** popup appears, type **chart** in the search window on the left panel.
- Then, the right panel will display the search results, and click the **Install** button for the **Chart Element** (probably the top one)



- When the installation is complete, click DONE to close the popup
- If Chart Element is included in Installed Plugins, you are good to go.

The screenshot shows a software interface with a navigation bar at the top. The page title is "pet\_weight\_register". On the right side of the title are buttons for "Edit", "Help", and "Saved". Below the title is a vertical sidebar with icons and labels: "Design" (gear), "Workflow" (stacked boxes), "Data" (database), "Styles" (paintbrush), "Plugins" (blue square with white "b"), "Settings" (gear), and "Logs" (document). The "Plugins" icon is highlighted with a blue border. To the right of the sidebar, the main content area has a header "Installed Plugins". Below it is a section titled "Bubble Plugins" with a "b" icon. Underneath is a section titled "Chart Element" with a pie chart icon. A red horizontal bar is positioned below the "Chart Element" section.

- There are two types of Bubble plug-ins: free and paid.
- Plugins marked **Free** or **By Bubble** in the list of plugins are free.
  - Especially the ones marked **By Bubble** are literally official Bubble plugins, so you can trust them for anything.
- On the other hand, plug-ins with a price are literally paid plug-ins.
- Since there are various plug-ins, when you install a plug-in, you should first look for a plug-in that is free and sufficient, and if there is no other plug-in but a paid plug-in, you may consider installing a paid plug-in.



## Simple LineChart from ApexCharts

Free

Easy to use beautiful line chart from the Apex JS library

By Avalan Labs

2,710 apps

All versions 4.0/5.0

Current version 4.0/5.0

Plugin page

**Install**



## Chart JS - Charts and Graphs

\$190 once or \$9/month

24 highly-customizable and beautiful charts! These charts are built on the most popular Javascript chart library (Chart.js) as well as hundreds of hours of development for custom features and perf...

By TechBlocks

2,192 apps

All versions 4.8/5.0

Current version 4.8/5.0

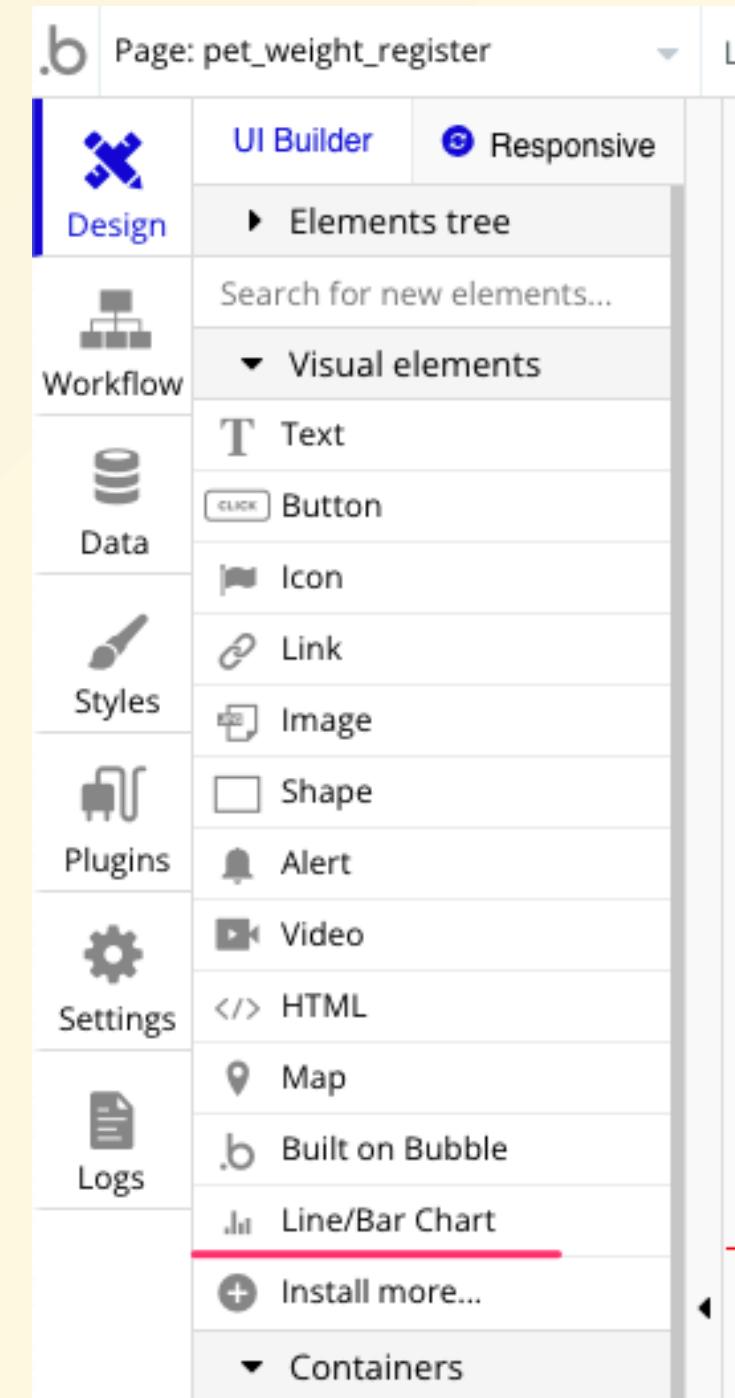
Plugin page

**Buy \$190**

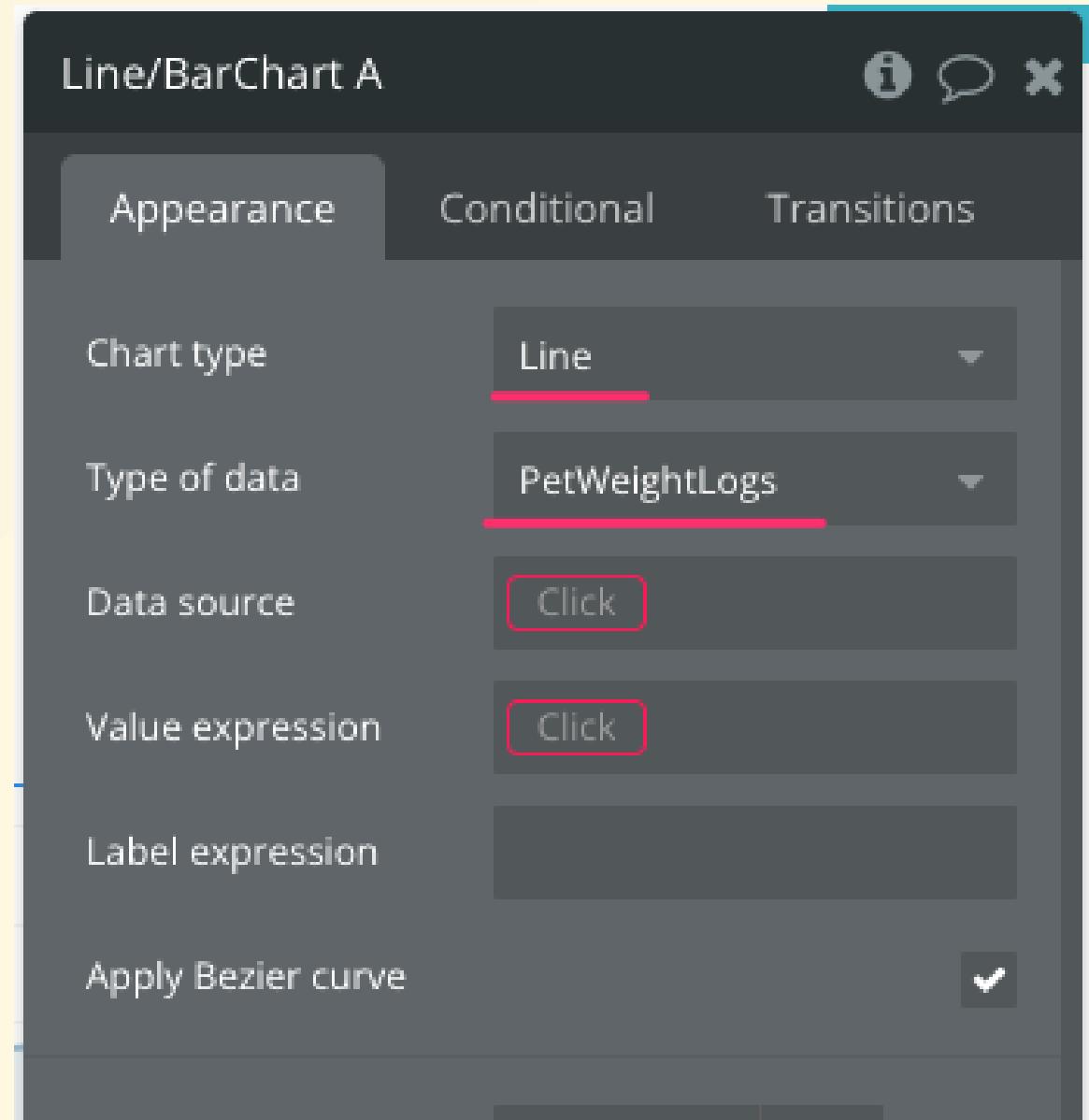
**Subscribe**

# Now let's do some actual graph drawing!

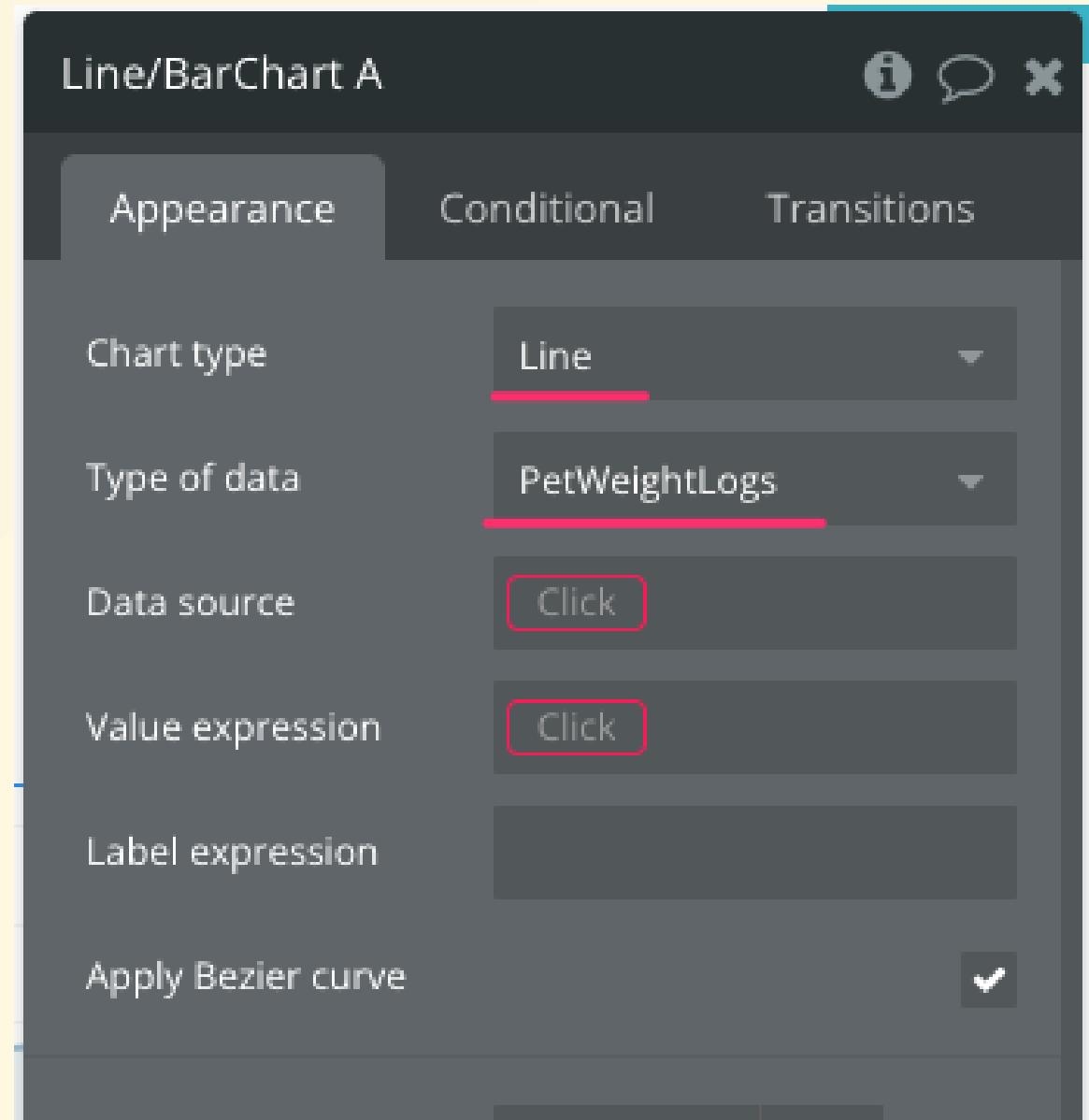
- Open the Design section of the `pet_weight_register` page.
- In the `Visual elements` section, you will see a new element called `Line/Bar Chart`, select it and drag it to the top of the weight input.



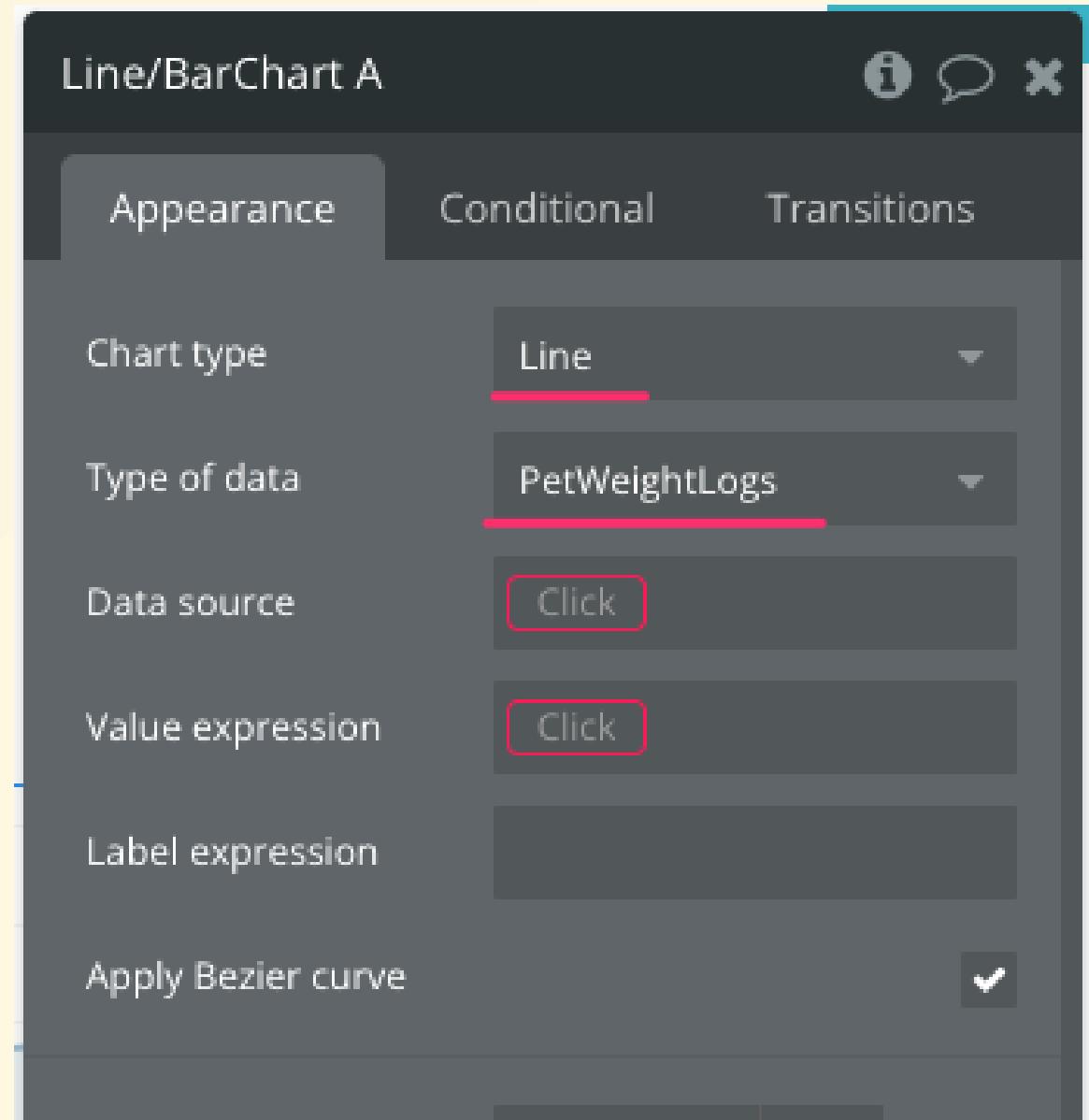
- Once drawn, a dialog box for detailed settings will appear as before, so let's set each item!
- The **Chart type** specifies the type of graph.
  - We'll use **Line** for the line graph.



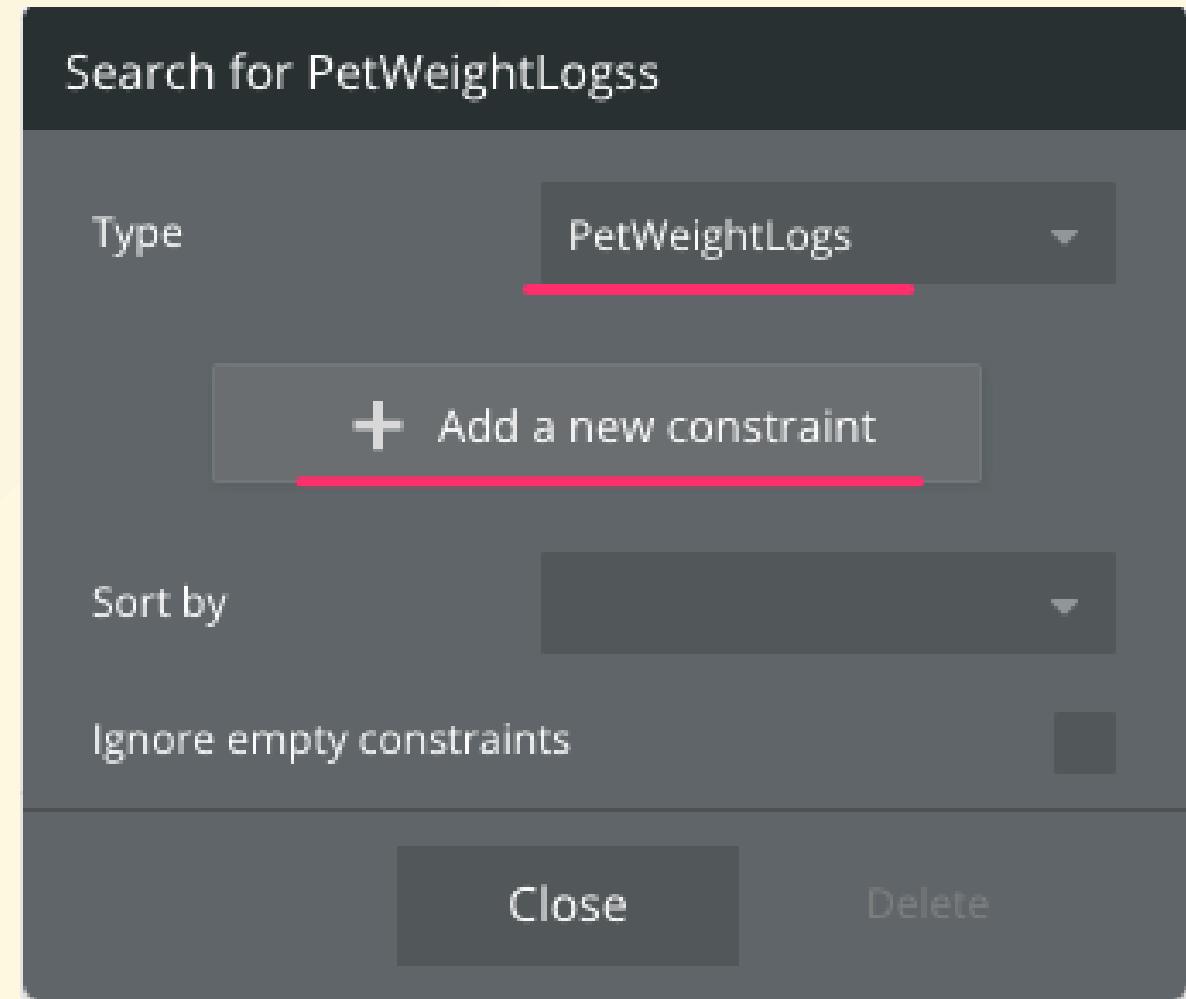
- The **Type of data** specifies the type of data to be graphed.
  - This time, we'll use **PetWeightLogs** to display the weight of pets.



- Then, specify the condition of the target data in **Data source**.
  - As in the list screen, first select **Do a search for**.

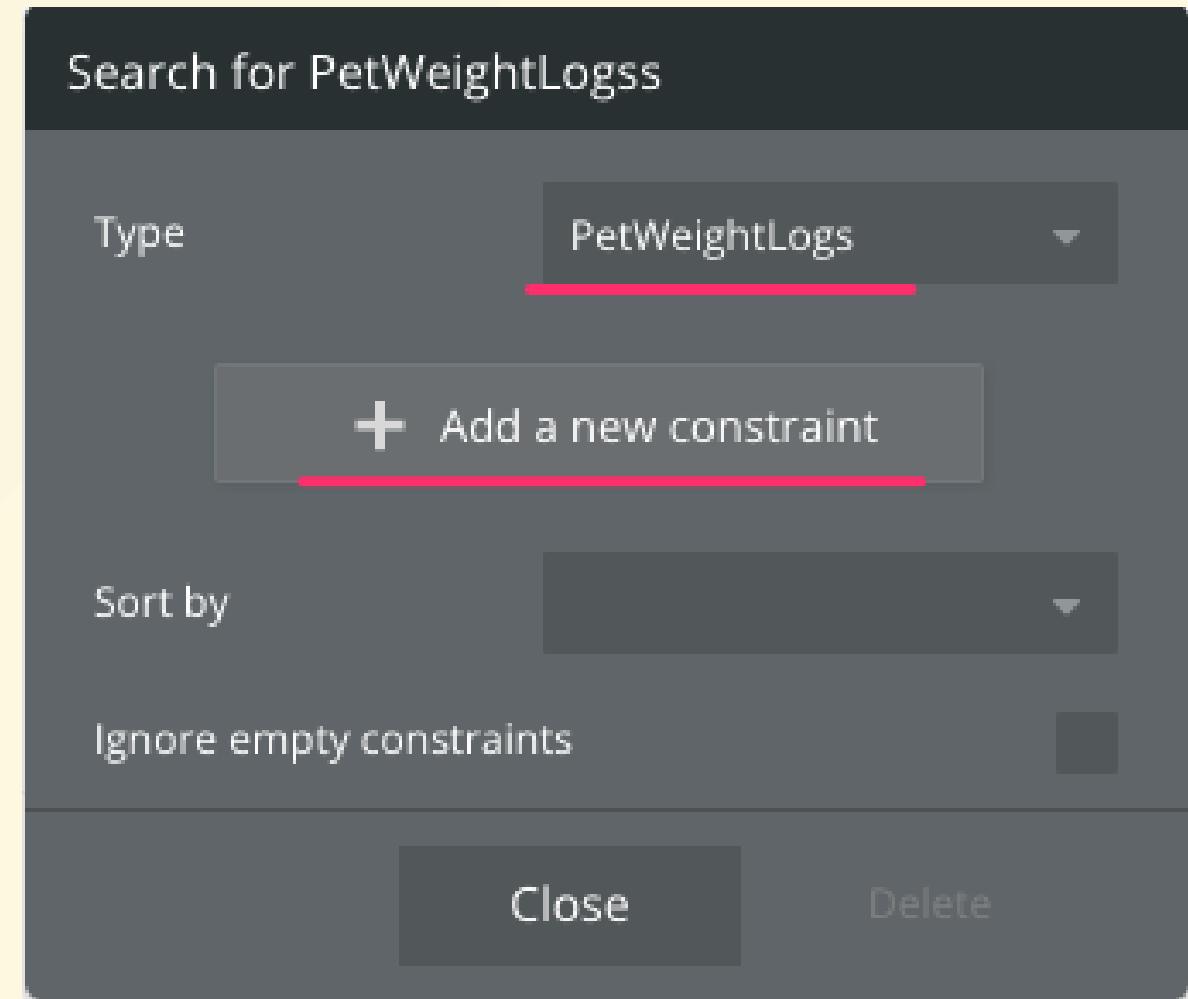


- The **Do a search for** dialog box will appear.
  - Select **PetWeightLogs** for **Type** to specify the type of the target data.

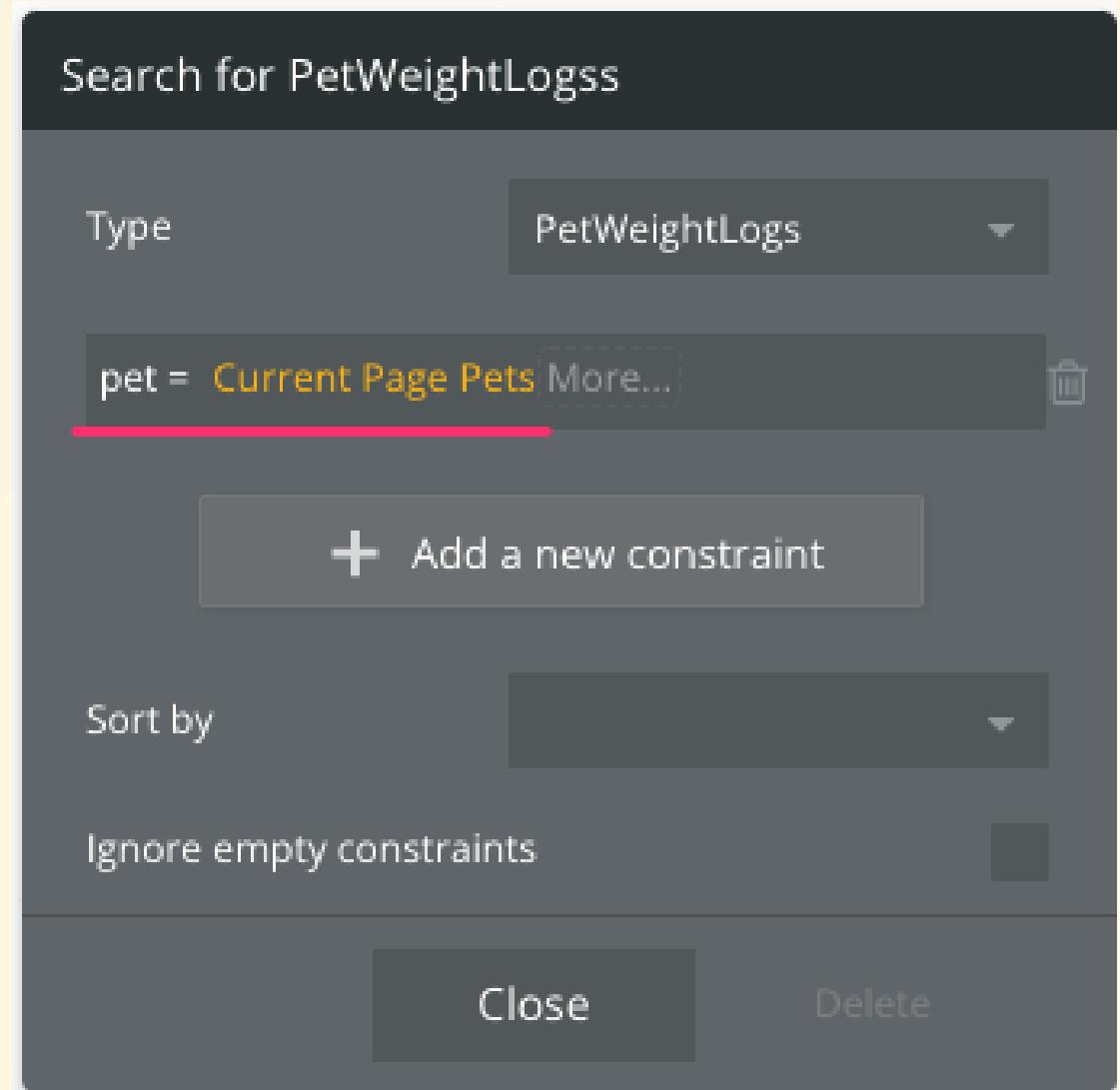


- This will display all weight data for all pets, so we need to specify the criteria for narrowing down the list.

- Click on  
Add a new constraint to  
specify the criteria

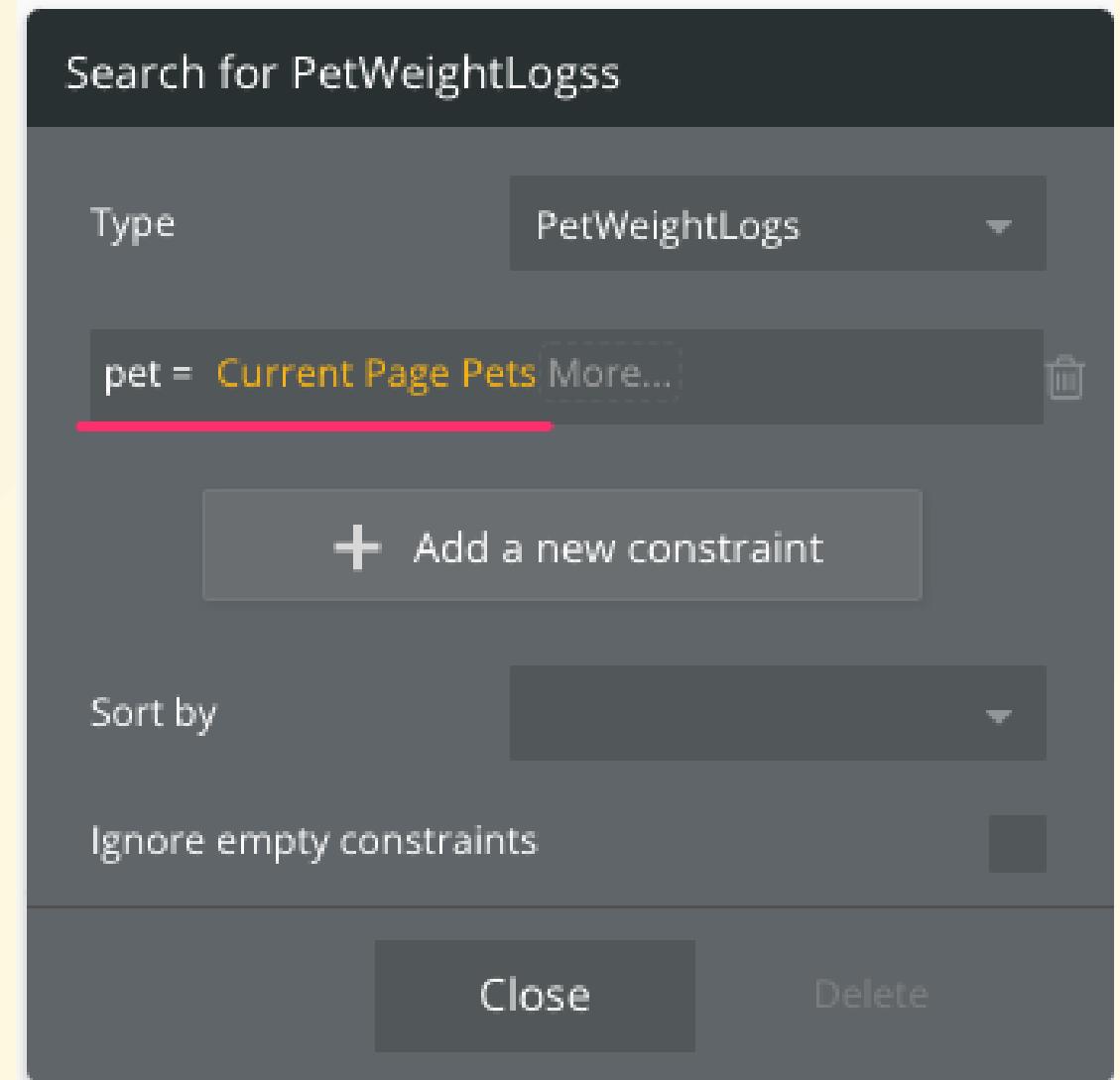


- First, click on Click to display the fields of PetWeightLogs specified in Type.
- This time, we will target the data of the Pet associated with PetWeightLogs, which is the same as the currently displayed pet.



- If we express this as a condition, it will look like this, so let's set it

```
pet = Current Page Pets  
``` pet = Current Page Pets
```



- Finally, specify the order of the data.
  - To show the graph, we want to put the registered weights in order of newest to newest



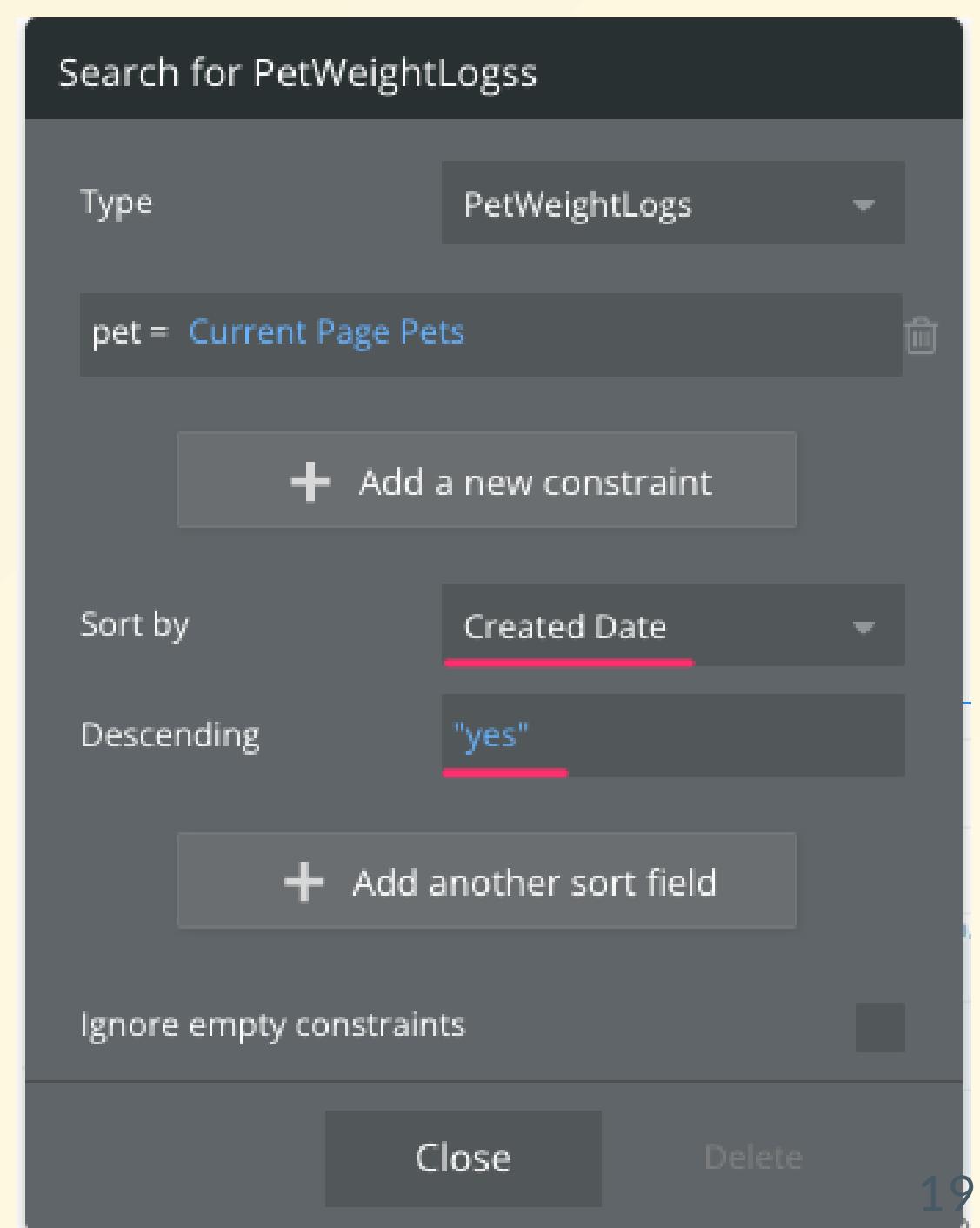
Weight

pet weight

← Back to detail

ADD

- Specify `Created Date` for `Sort by`.
  - Now we've specified the sort order key.
- Set `Descending` to "yes".
  - Now we specify the order of the keys in descending order



- If you want to order by oldest first, specify "no" for **Descending** to make it oldest first (ascending order).
- When you are done, click **Close** to close this sub-dialog.

- Finally, specify the X-axis and Y-axis to be drawn on the graph.
- In this case, the X-axis is the registration date and time, and the Y-axis is the weight.



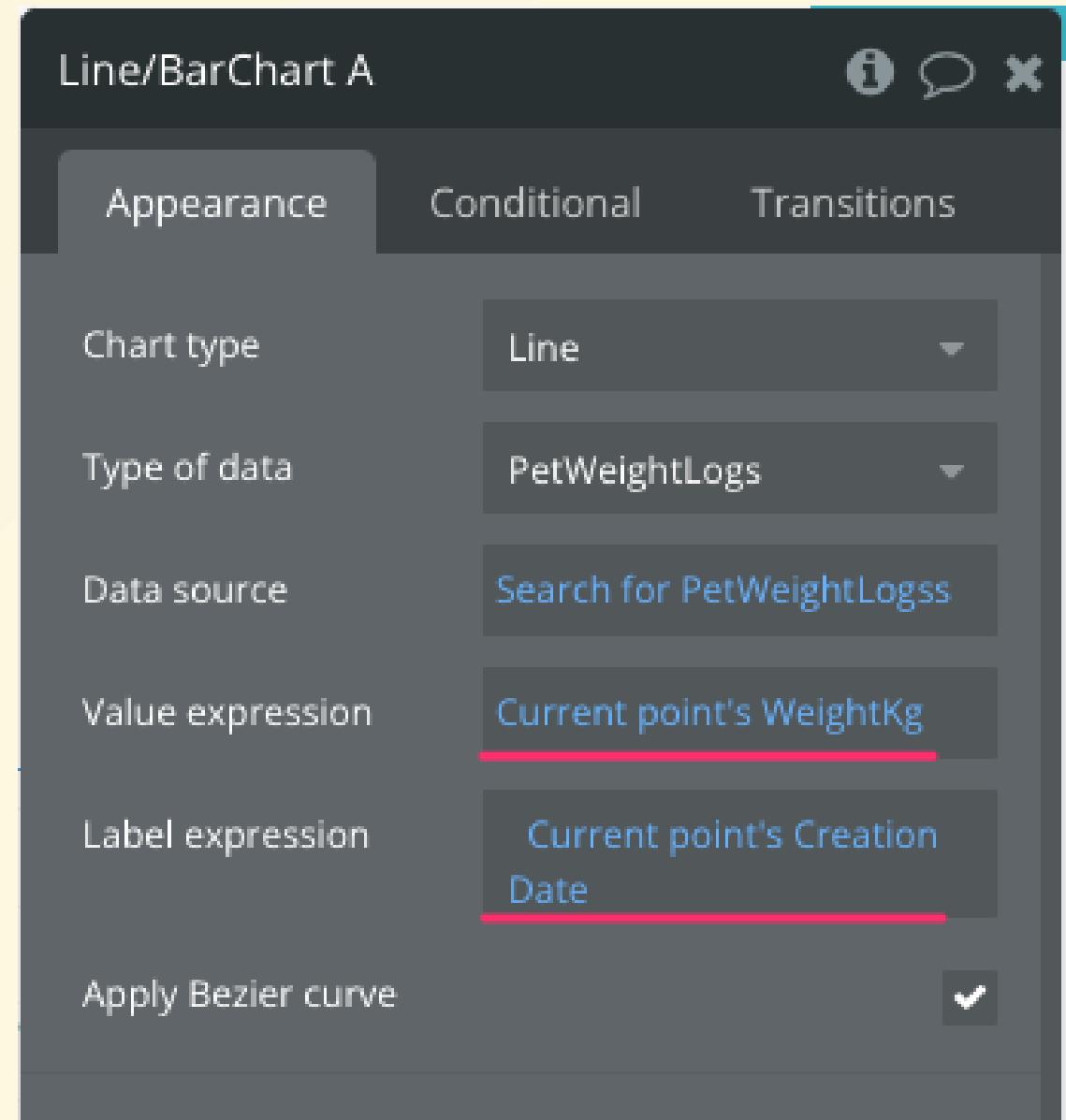
Weight

pet weight

← Back to detail

ADD

- To set the X axis, use **Label expression**.
  - In this case, we'll use Current point's Creation Date
- The Y axis is set to **Value expression**.
  - The Y axis is set to Current point's WeightKg this time



- The `Current point` refers to the individual data plotted for each of the `PetWeightLogs` specified in the target data type !

- Now let's go back to the list screen and preview it!
- Is the weight registered for the selected pet in the list displayed as a graph?
- Does the graph change as you add more and more weights in the same screen?



## **Appendix4: Let's change the labels of the X-axis of the graph.**

- At the moment, the labels on the X-axis are in a long date/time format, so you can change them to any format you like.
- For example



## Appendix5: Let's send a message when there is no weight data.

- In the current development, if there is no weight data for a pet, the graph drawing area will be blank and the page will not look good.

Weight

pet weight

← Back to detail

ADD

- So, let's try to improve usability by displaying a special message when none of the pet's weights have been registered!

Please ADD Your Pet Weight!!

Weight

pet weight

← Back to detail

ADD

- In the corresponding image, a Shape with the same width and height as the graph element is prepared at the same position.
  - Place the message you want to display on top of it as Text, and group the Shape and Text together.
  - By grouping them together, you can treat them as one element.
- Set the following for the Shape
  - The element itself will always be hidden
  - If the number of PetWeightLogs in Conditional is 0, the element will be shown.
- In this way, the Shape can be displayed when no weight is registered, and the graph can be displayed when no weight is registered.

- Here's what the setup image looks like

The image displays two side-by-side configuration interfaces.

**Left Screenshot: Search for PetWeightLogss**

- Type: PetWeightLogs
- pet = Current Page Pets
- + Add a new constraint
- Sort by
- Ignore empty constraints
- Close
- Delete

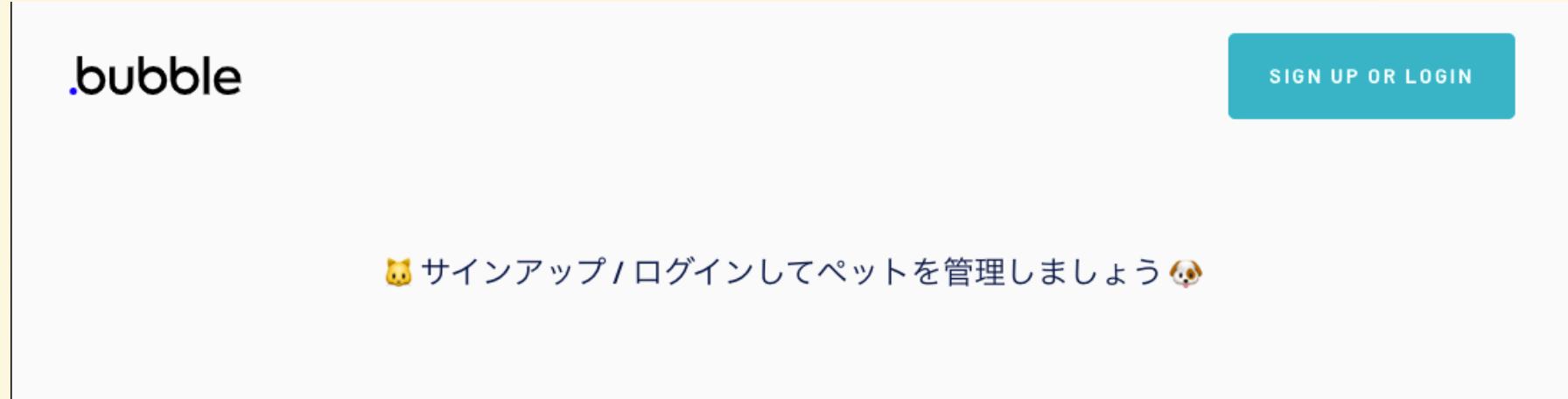
**Right Screenshot: Group A**

- Appearance
- Conditional (1)
- Transitions
- OFF remove condition
- When Search for PetWeightLogss:count is 0 More...
- This element is visible  This element is visible
- Select a property to change when true
- + Define another condition

## Appendix6: Let's display different messages in the pet list.

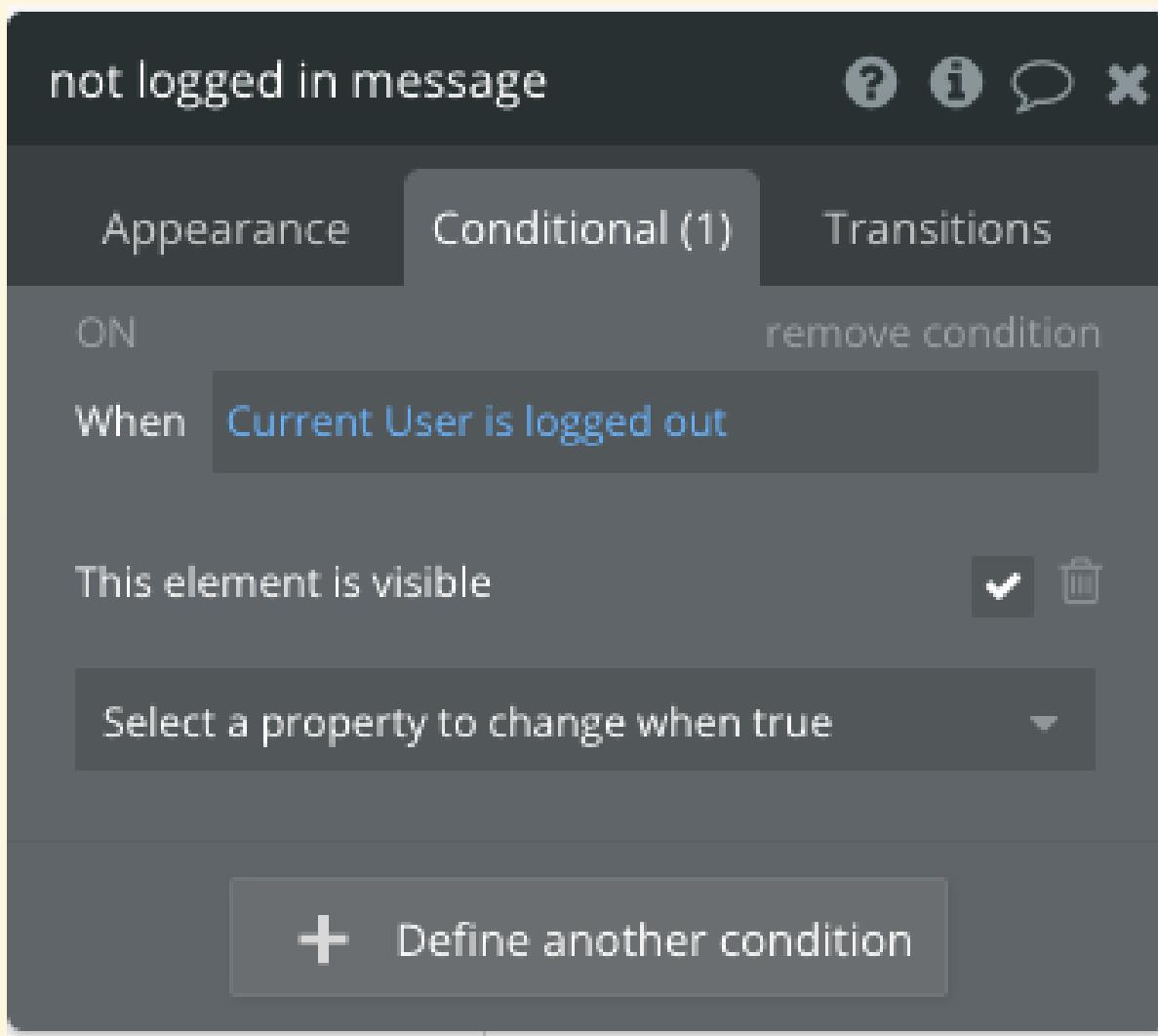
- You need to be logged in to see the Pet List.
- Moreover, even if you are logged in, nothing will be displayed if no pets have been registered.
- However, it is difficult for users to understand why the Pet List is not displayed.

- So, let's send out different messages according to each status



- The concept of the setting is the same as the one described above when the weight of the pet is 0.
- Prepare a message for each of these conditions, but hide them by default.
- Then, when each condition is met, the message will be displayed.
  - Whether the current user is logged in or not?
  - Is the current user logged in and does the user have at least one registered pet?
- The answers (settings) are written on the next page!

- Is the current user logged in or not?



- Whether the current user is already logged in and whether there is at least one pet registered by the user?

The screenshot shows a configuration dialog for a UI element named "pet empty message". The top bar includes standard icons for help, information, message, and close. Below the title, there are three tabs: "Appearance" (selected), "Conditional (1)", and "Transitions". The "Conditional (1)" tab is active, showing a condition setup. The condition is set to "OFF" and can be removed. The "When" condition is defined as "Current User is logged in and Search for Petss:count is 0". Below this, under "This element is visible", there is a checked checkbox and a trash bin icon. A dropdown menu allows selecting a property to change when the condition is true. At the bottom, a button labeled "+ Define another condition" is available.

## Appendix7: Let's create an update function by setting up a line from the detail screen to the update screen.

- As for the points, they are as shown in the following sheet! Please try to put them into practice!

Name

Image



Birthday

Gender

[← Back to detail](#)

UPDATE

- Since the update screen has almost the same screen elements as the registration screen, let's clone `pet_register` and create `pet_update` !
- In the registration screen, the content of each input element (name, birthday, etc.) was empty, but in the update screen, we need to set the value of the data that has already been registered at the time the screen is displayed.
- From the dialog of each input element, there is an item that looks like this, and we can specify the initial value there.
  - Hint  Insert dynamic data
  - Hint  Initial content

- As for the behavior when the update button is clicked, it was **Create a new thing...** in **Data(Thing)** when it was a new registration. But this time it's an update, right?
  - Speaking of updating, change... Oh, there's an action that looks like that 😊
- Finally, when we move from the detail screen to the update screen, we need to pass the information of the pet displayed in the detail screen. This is the part I learned in today's lecture!
  - Hint 💡 Data to send

# Summary

- In this lecture, we started with how to use Bubble, and then went on to the actual design of the screen and integration with the database.
- The basic flow is similar to that of Adalo, but the workflow is a little more in line with why it is called a visual programming tool.
- In the next article, I hope to give a more in-depth lecture on Bubble, so stay tuned!

**That's all!**

**Thank you for your time!**